GeoNode Installation and Configuration

Salvador Bayarri
sbyarri@gmail.com
World Bank Consultant
Contents

• GeoNode components
• Preparation of environment: virtualization
• GeoNode installation process: from packages and from tarball
• Additional tools installation
• Basic GeoNode configuration
• URL configuration and change
• User administration and registration
• Printing configuration
• Management commands
• Web customization
• Debugging GeoNode
• References and support
GeoNode stack

Application layer

Service layer

Data layer
GeoNode stack
PostGIS

• PostgreSQL & PostGIS: Spatial Database

The database component manages *data* and *configuration* information for Geonode/Django, GeoNetwork and GeoServer. All of these tables and data are stored within the *geonode* database on PostgreSQL.

In the workshop, we will use the *pgAdmin* application to manage the PostGIS database directly. This application is not in the GeoNode default installation.
GeoNode stack

GeoServer

• **The GeoSpatial Data Manager: GeoServer**

GeoServer is a Java-based server with an OGC-compatible data store that can provide **WMS**, **WFS**, **WCS** and other services in common formats like **GML**, **GeoJSON**, **KML** and **GeoTiff**.

It can be connected to different spatial databases including **PostGIS**, **Oracle Spatial**, **ArcSDE** and others.

ArcSDE and Oracle data can be added to GeoNode using console commands.

GeoServer has a **Web front-end** to handle data storage, styling and Web service configuration directly, but this should be avoided in the context of GeoNode.
GeoNode stack

GeoNetwork

- The Geo-MetaData Manager: GeoNetwork

GeoNetwork is a Java-based server managing metadata storage, search and access. It provides a standard catalog interface based on OGC standards. It is used via the CSW Web service to create and update metadata records when they are accessed in GeoNode.

GeoNetwork has a Web front-end to search, edit and import/export metadata in ISO 19139 and other formats, and CSW service configuration.
GeoNode stack

Django

• The GeoNode front-end Manager: Django

Django is a high-level *Python Web framework* that encourages rapid development and clean, pragmatic design.

It allows the user to easily tweak the content and look and feel and to extend GeoNode to build customized GeoPortals.

It includes tools to handle user registration and accounts, avatars, and helper libraries to interact with GeoServer and GeoNetwork.

There is a wide range of third party apps that can be plugged into a GeoNode-based site including tools to connect to different social networks, to build content management systems and more.
GeoNode stack
GeoExt and OpenLayers

• The Mapping Components

The main map interface for GeoNode is the Map Composer / Editor. It is built on top of GeoExt and uses OpenLayers, GXP. It talks to the other components via HTTP and JSON as well as standard OGC services.
Virtualization

• GeoNode is typically installed on Ubuntu Linux
• To run GeoNode in a Windows environment, the best solution is to create a *Virtual Machine*
• Virtual Machines are also a good idea to *distribute* samples, *learn* SDI administration with minimal risk and use them in *training*, because they provide a reproducible environment
• Virtual Machines can also be used to *scale* up your GeoNode-based SDI to more powerful hardware, even run it on a cloud-based infrastructure
Virtualization

- How does it work?
Virtualization example: Ubuntu ‘guest’ running on Windows ‘host’

- **Download the guest Operating System** as an .iso file (installation CD-DVD)
  For the workshop, we use *Ubuntu 10.04 (Lucid) i386* release: [http://releases.ubuntu.com/lucid/](http://releases.ubuntu.com/lucid/)

- **Download and install the virtualization environment** for the host OS
  For the workshop, we use Oracle’s Virtual Box (free): [https://www.virtualbox.org/wiki/Downloads](https://www.virtualbox.org/wiki/Downloads)
Virtualization example:
Ubuntu ‘guest’ running on Windows ‘host’

Create a Virtual Machine with VirtualBox, using the Ubuntu .iso file

See details in http://www.psychocats.net/ubuntu/virtualbox
Virtualization example: Ubuntu ‘guest’ running on Windows ‘host’

Assign memory and disk space
Virtualization example:
Ubuntu ‘guest’ running on Windows ‘host’

Assign disk space
Virtualization example: Ubuntu ‘guest’ running on Windows ‘host’

Assign as boot device the Ubuntu .iso file
Virtualization example: Ubuntu ‘guest’ running on Windows ‘host’

Start the VM.
The autorun of the Ubuntu .iso disk starts up
Select the ‘Install Ubuntu’ option
Virtualization example: Ubuntu ‘guest’ running on Windows ‘host’

Accept the default disk configuration to use the space you created earlier.
Virtualization example:
Ubuntu ‘guest’ running on Windows ‘host’

Create the super-user account and give a network name to the virtual machine
Virtualization example: Ubuntu ‘guest’ running on Windows ‘host’

Once installation is done, **click** on “Restart Now” and then **close** the VM window with the “Power off” choice.
Virtualization example: Ubuntu ‘guest’ running on Windows ‘host’

Start the VM again from VirtualBox

Log into Ubuntu and run the “Update Manager” to get latest updates (make sure you are connected to the Internet)
Virtualization example: Install Guest Additions

To allow larger screen modes, run the “Devices/Install Guest Additions” menu command...

... and accept the Autorun action once the virtual CD is mounted
Authenticate with your user’s password when prompted
Wait until all modules are built
Virtualization example: Install Guest Additions

After restarting the virtual machine, you can set the Auto-resize mode to change the VM display size like a normal window...

... or use the “System/Preferences/Monitors” to set one of the suggested sizes

Then **eject the Guest Additions CD** in the desktop
GeoNode installation

• Make sure you have *superuser* privileges and password (if you installed the virtual machine, just use your user and password)

• Open a command terminal

  “Applications/Accessories/Terminal” in Ubuntu 10.04)
GeoNode installation
(simple way, using deb packages*)

• Make sure previous GeoNode installations are removed, by executing the following commands:

```
sudo apt-get remove --purge geonode
sudo apt-get autoremove --purge
sudo rm -r /etc/geonode
```

• Run the following commands to download and install the GeoNode packages from the official repository, then **restart**:

```
sudo apt-get install python-software-properties
sudo add-apt-repository ppa:geonode/release
sudo apt-get update
sudo apt-get install geonode
```

* Geonode packages may not available yet for most recent Ubuntu releases
  This has been tested with Ubuntu 10.4 - Lucid
Testing the GeoNode installation

- To test that the installation is correct, open a Web browser and open the URL http://localhost
- You should see the GeoNode welcome page
Testing the GeoNode installation

- In the [http://localhost/geoserver/](http://localhost/geoserver/) you should see the GeoServer welcome page
Testing the GeoNode installation

• In the [http://localhost/geonetwork/](http://localhost/geonetwork/) you should see the GeoNetwork welcome page
GeoNode installation
(another way, using tarball distribution)

• Use this option for Linux distributions for which no installation packages are available
• Install updates for your Linux distribution
• Install dependencies (python, Java, tomcat, postgres, gdal...):

```bash
sudo apt-get install python python-support python-dev python-virtualenv
openjdk-6-jre tomcat6 postgresql-8.4 gcc patch zip python-imaging
gdal-bin libgeos-dev python-urlgrabber
python-pastescript gettext postgresql-contrib
postgresql-8.4-postgis libpq-dev unzip libjpeg-dev libpng-dev
python-gdal libproj-dev python-psycopg2 apache2 libapache2-mod-wsgi
libxml2-dev libxslt1-dev
```
GeoNode installation
(another way, using tarball distribution)

• Download desired tarball package from http://dev.geonode.org/release/, for instance “GeoNode-1.2.tar.gz”

• Extract it to a folder:
GeoNode installation
(another way, using tarball distribution)

• From within the extracted folder, run the install script with the configuration for your Linux platform
  (if the config file for your OS version is not in the ‘support’ folder, you may need to create one manually)

  `sudo ./install.sh support/config-ubuntu.sh`

• If you are using other Linux platforms, you may need to change the installation script (.sh), even install GeoNode from source code, which will also enable you to inspect and modify the GeoNode code (you don’t need to do this for normal customization)

Additional installations for workshop: Quantum GIS (qGIS)

- *qGIS* is an open-source GIS which can connect to SDI services and PostGIS database from GeoNode
- To install, add these lines in your /etc/apt/sources.list file for you Ubuntu distribution (‘lucid’ in our case)
  
  ```
  deb     http://qgis.org/debian lucid main
  deb-src http://qgis.org/debian lucid main
  ```

- Then run the installs (python and grass plugins included here)
  ```
  sudo apt-get update
  sudo apt-get install qgis
  sudo apt-get install python-qgis
  sudo apt-get install qgis-plugin-grass
  ```
Additional installations for workshop: Quantum GIS (qGIS) - Plugins

• After installing qGIS, run the Quantum GIS Desktop from the Applications/Science menu

• Run the ‘Plugins/Fetch python plugins’ command

Install these plugins if they are not already installed:
- OpenLayers
- CSW Client
- OpenStreetMap
- GDAL
- fTools

NOTE: need to change owner of the /home/user/.qgis folder to allow plugins installs
Additional installations for workshop:
pgAdmin

- `pgAdmin` is an open-source tool to manage, query and perform operations on PostgreSQL-PostGIS databases
- To install, just run:

  ```
sudo apt-get install pgadmin3
  ```
GeoNode basic configuration

• To start using GeoNode as administrator, just run:

  ```
sudo geonode createsuperuser
  ```

• The geonode command interpreter will request:
  – Your VM superuser password (VM logging password)
  – The name and password you want for the GeoNode superuser (to administer GeoNode & GeoServer)
  – The email address you want system notifications to be sent to
GeoNode basic configuration

• Let’s review some basic configuration written to the /etc/geonode/local_settings.py file

  sudo gedit /etc/geonode/local_settings.py

• Notice the connection information to access the Posgres-PostGIS DB, and the administration password (we don’t need to change it)

```python
DATABASE_ENGINE = 'postgresql_psycopg2'
DATABASE_NAME = 'geonode'
DATABASE_USER = 'geonode'
DATABASE_PASSWORD = 'cuckwCF1k'
DATABASE_HOST = 'localhost'
DATABASE_PORT = '5432'
```
GeoNode basic configuration

• We can also find the default user and password for GeoNode to access Geonetwork (the metadata manager)

```python
# The username and password for a user with write access to GeoNetwork
GEONETWORK_CREDENTIALS = 'admin', 'admin'
```

• We need to change the default ‘admin’ password in this file, but also by logging into Geonetwork!

• Do not forget to **restart the server** after saving the settings file
Geonetwork basic configuration

• To change the default Geonetwork password, log in with the default user and password into http://geonodeURL/geonetwork

• In the Administration menu, use the ‘Change password’ link to set the same password written to the local_settings.py file
GeoNode basic configuration

• The default map layers that appear in the GeoNode map viewer are defined in the MAP_LAYERS section

```json
MAP_BASELAYER = [{
    "source": {
        "ptype": "gxp.wmscssource",
        "url": GEOSERVER_BASE_URL + "wms",
        "restUrl": "/gs/rest"
    }
}, {
    "source": {
        "ptype": "gx_olsource",
        "type": "OpenLayers.Layer",
        "args": ["No background"],
        "visibility": False,
        "fixed": True,
        "group": "background"
    },
    "source": {
        "ptype": "gx_olsource",
        "type": "OpenLayers.Layer",
        "args": ["No background"],
        "visibility": False,
        "fixed": True,
        "group": "background"
    }
},
```

• We can remove these entries, or add new ones
  – To add a default WMS layer in the GeoNode map viewer, add:
    
    "source": {"url": "http://my-server/wms"},
GeoNode basic configuration

• We can uncomment some lines in the configuration file to receive error notifications. This is very useful!

```python
# Uncomment the following to receive emails whenever there are errors in GeoNode
#ADMINS = (
#     ('John', 'john@example.com'),
#     )
```
GeoNode basic configuration

• Before going on, let’s upload a test shapefile in GeoServer to check everything works
• In this way we will see the problems when we change the node URL afterwards
GeoNode URL configuration

• To make the GeoNode accessible across the network, it must have a URL or IP number instead of ‘localhost’
• If we use a virtual machine, first we need to change the network configuration to use a bridge adapter:
GeoNode URL configuration

- The virtual machine must now pick up whatever IP is assigned by your network configuration.
GeoNode URL configuration

• To make GeoNode work with a specific URL, just change the ‘SITEURL’ property in the settings file:

```
sudo gedit /etc/geonode/local_settings.py
```

```
import os
import geonode

DEBUG = TEMPLATE_DEBUG = False

SITENAME = 'GeoNode'
SITEURL = 'http://192.168.2.99/'
```

DATABASE ENGINE = 'postgresql_psycopg2'
DATABASE NAME = 'geonode'
DATABASE USER = 'geonode'
DATABASE PASSWORD = 'u4kwCF1k'
DATABASE HOST = 'localhost'
DATABASE PORT = '5432'

```
GeoNode URL configuration

- Restart the server: `sudo service apache2 restart`
- Try the new IP in both the server and another computer across the network

![GeoNode URL configuration](image)
GeoNode URL configuration

- The site URL must also be used in configuring the GeoServer proxy
- Log in GeoServer at http://geonodeURL/geoserver
GeoNode URL for user registration

- To allow user automated registration from the GeoNode front-end, we must also change the site address or domain name in the Admin menu, so the registration link sent in the emails is correct
- Log in GeoNode and follow the ‘Admin’ link to the ‘Site’ config
GeoNode URL configuration

• A problem with changing the URL of your GeoNode: links in existing metadata to the layer pages and services will be broken!

• As of now, there is no management tool to update the URLs automatically, so you need to do it by hand
GeoNode URL configuration

• First, we need to change a couple of metadata tables in the configuration stored in the PostgreSQL DB

• To do this, we can use the pgAdmin tool (See details in the GeoNode_DM2 presentation)

• In the maps_layer table, change the ‘distribution_url’ field
GeoNode URL configuration

- Also, in the `maps_maplayer` table, change the ‘ows_url’ field for those layers that have already a string there.
GeoNode URL configuration

• Secondly, we also need to change the metadata stored by Geonetwork
  See more details about Geonetwork in the GeoNode_DM2 presentation
• Log in as administrator in http://geonodeURL/geonetwork
• Find the metadata by using the Search tools
• Click on the ‘Edit’ button for the metadata that needs updating
GeoNode URL configuration

• Go to the ‘Transfer options’ section of the metadata
• Change the URL for all OnLine resource entries (there will be a number of them)
GeoNode user administration

• Users can be created in two ways in GeoNode, by using the Django administration platform
  – Administrators can directly create and configure users in GeoNode Web
  – Administrators can enable user to register automatically

• There is no specific GeoNode Django administration documentation, but you can check:

GeoNode user administration

• How to configure groups and users in GeoNode?
• Go to the ‘Admin’ link in the GeoNode home page
GeoNode user administration

• Create *groups* to define common types of permissions for users (e.g. data uploader, map editor, content reviewer, etc.)
GeoNode user administration

Django administration
Welcome, salvador. Change password / Log out

Home › Auth › Groups

The group "Community manager" was added successfully.

Select group to change

Action:  

0 of 2 selected

Group
Community manager
Editor

2 groups
GeoNode user administration

- Users inherit permissions from groups, but can be overwritten
- We can deactivate users without deleting them
GeoNode user administration

- Staff users can log into the Administration site
- Superusers have all privileges, even if not explicitly set
GeoNode role mapping

- Another way to define permissions is by using *role mappings*, which are combinations of the basic permissions.
- The combinations are defined in the ‘object role’ section.
GeoNode role mapping

- The defined object role mappings can then be applied per layer and map, to distinguish privileges for *registered versus anonymous users*
- This is a way to separate more ‘public’ content (anybody can access for reading) from more ‘institutional’ content (only named users can access)
GeoNode role mapping

• The role mappings per layer can also be applied differently for specific users, giving more control to the administrator on privileges
• This is done through the ‘user role mappings’
Community management

• Staff users can control elements contributed through the social features of GeoNode, like comment, tags, etc.
• With the right permissions, comments and tags can be modified or deleted
Community management

- An important information field in the metadata specifies the author and contact person for each layer.
- This information can also be changed in the administration tools for each layer.
Community management

• The list of contact people can also be managed
• The GeoNode users become contacts by default, but contacts can be non-users, too
User registration

- To allow user automated registration from the GeoNode front-end, the following steps must be followed
  - Set up the email backend for Django if not set already (https://docs.djangoproject.com/en/dev/topics/email/#e-mail-backends)
  - Set appropriate email backend local settings and enable registration

```python
# Uncomment the following to use a Gmail account as the email backend
EMAIL_BACKEND = 'django.core.mail.backends.smtp.EmailBackend'
EMAIL_HOST = 'smtp.gmail.com'
EMAIL_HOST_USER = 'sbayarri@gmail.com'
EMAIL_HOST_PASSWORD = 'password'
EMAIL_PORT = 587
EMAIL_USE_TLS = True

# User Registration
ACCOUNT_ACTIVATION_DAYS = 7
REGISTRATION_OPEN=True
```
User registration

• After registration is enabled, users will see a Register link in the GeoNode Web site.
User registration

- The administrators will be able to see registered users in the User list, with their registration status, and change their privileges as with other users.
User management in GeoServer

• For other users to be able to style data, upload external data or cascade external services using GeoServer, appropriate privileges must be given

• GeoServer provides also ways to secure access to the data and services, but this we should leave to GeoNode permissions management

See more about users and security in GeoServer in:
User management in Geonetwork

• For other users to be able to create or modify metadata using Geonetwork, appropriate privileges must be given (by default, only the admin account exists)

See more about user management in Geonetwork in:
Performance enhancements

• The key performance factor for GeoNode mapping is the GeoServer image management

• Some tricks:
  – Install the *native Java image libraries* instead of the default ones:
    
    n/java.html#install-native-jai-and-jai-image-i-o-
    extensions

  – Set the JAI settings in GeoServer as shown, to have acceleration enabled

  – Do not set resource consumption limits for WMS in GeoServer (they are already 0 by default)
Printing configuration

- The overall printing configuration is handled by the Mapfish print service managed by GeoServer
- The configuration can be found here:
  
  /var/lib/geoserver/geonode-data/printing/config.yaml

- It contains allowed DPIs, scales, and a description of the layout options
- When printing maps that contain images (WMS, tiles) generated outside our GeoNode, we will need to add these map servers to the config:

```yaml
hosts:
  - !localMatch
dummy: true
  - !ipMatch
    ip: www.camptocamp.org
  - !dnsMatch
    host: labs.metacarta.com
    port: 80
  - !dnsMatch
    host: terraservice.net
    port: 80
```
Printing configuration

- In the YAML file we also find the description of the *layout options* for printing the maps.
- The default layout is very simple. To customize the layout options, you can use the documentation at:

http://www.mapfish.org/doc/print/configuration.html
Management commands

• To create the GeoNode superuser we used a geonode command line instruction

• There are other management commands implemented on the Django framework. We can get the full list by doing

  geonode help

• For instance, we can use the ‘importlayers’ command to upload shapefiles and GeoTIFF files into GeoNode, instead of using the Web front-end

• The command you may have to use is **updatelayers**, which will synch GeoNode when external data sources are added via GeoServer (see the GeoNode_DM2 presentation)

• We will also use the **collectstatic** command later to update the Web pages when we change their content or style

• For more information:

Web front-end customization

- GeoNode uses Django *templates* to create the Web site appearance and content

- Two important folders that specify *content* are:

  **The default distribution templates.** According to the documentation is in
  /var/lib/geonode/geonode/templates/  
  *But in our case they are in:*

  /var/lib/geonode/src/GeoNodePy/geonode/templates/

  **The overriding templates** for customization. Empty by default

  /etc/geonode/templates/  
  *Copy here the default files*  
  *and edit them to customize the content*
Web front-end customization

• For instance, to customize the content of the GeoNode home page, we can copy the *index.html* file from the default template to the customization folder, and edit it:

```bash
sudo cp /var/lib/geonode/src/GeoNodePy/geonode/templates /index.html
/etc/geonode/templates

sudo gedit /etc/geonode/templates/index.html
```

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>GeoNode</title>
</head>
<body>
    <div class="twocol">
        <div>
            <h2>{% trans "Welcome to our customized version of GeoNode" %}</h2>
            <p>{% blocktrans %}This is our community to share geographic information about Natural Resources in the Caribbean{% endblocktrans %}</p>
        </div>
    </div>
</body>
</html>
```
Web front-end customization

- If we refresh the GeoNode home page, the changes will be seen

- We can follow the same process to customize the data, maps, data search, etc. which can find in the same distribution templates folder
- For more information on the template syntax, see ‘The Template Layer’ in https://docs.djangoproject.com/en/1.4/
Web front-end customization

- A similar process can be followed to modify the style of the Web site

  The default distribution styles. According to the documentation is in
  /var/lib/geonode/geonode/media/ But in our case they are in:
  /var/lib/geonode/src/GeoNodePy/geonode/static/

  The overriding styles for customization. Empty by default

  /etc/geonode/media/ Copy here the default files
  and edit them to customize the style
Web front-end customization

• Exercise:
For instance, to customize the logo on the GeoNode home page, we can copy the logo_bn.png file from the default theme to the customization folder, and edit it:

```
sudo mkdir /etc/geonode/media/theme
sudo mkdir /etc/geonode/media/theme/img

sudo cp /var/lib/geonode/src/GeoNodePy/geonode/static/theme/img/logo-bg.png /etc/geonode/media/theme/img
```
Web front-end customization

• In our example, we can use the Gimp image editor
• If it is not installed, you can use a package manager to install this software
Web front-end customization

```
sudo gimp /etc/geonode/media/theme/img/logo-bg.png
```

Use the eraser tool to remove the default logo and type or draw a new one

- After editing and saving the logo image file, we have to run the `collectstatic` management command to propagate style changes to the Apache Web server

```
sudo geonode collectstatic -v0
```
Web front-end customization

• Exercise:

Most color, borders and font styles are defined in *site.css*, in the theme folder
Copy the original to the customization folder and edit this file

For instance, change the footer color
(colors are in RGB format, with each component from 00 to ff)

```css
#footer {
    background: #f10000;
    ...
}

#footer .onecol {
    ...
    background: #f10000;
}

#footer .twocol {
    ...
    background: #f10000;
}

#footer .threecol {
    ...
    background: #f10000;
}
```
Debugging GeoNode

• If you have problems making GeoNode work correctly, you should check the logs to see if there are any error messages

```java
2013-01-22 12:44:11,743 ERROR [geoserver.ows] -
org.geoserver.platform.ServiceException: Could not find layer gn:world
at org.geoserver.wms.map.GetMapKvpRequestReader.parseLayers
(GetMapKvpRequestReader.java:1180)
```
Debugging GeoNode

• Typically, we will get errors in the Web front-end like “500: Internal Error”

• If you need to get more meaningful error messages, sacrificing temporarily some performance, enable the DEBUG mode in the local settings file

```
import os
import geonode

DEBUG = TEMPLATE_DEBUG = True
```

• You need to restart the Apache server after this change
GeoNode installation reference

• The general GeoNode documentation site is in: http://docs.geonode.org/en/latest/
• The quick installation guide from package repository: http://docs.geonode.org/en/latest/intro/install.html
• Detailed installation from tarball distribution: http://docs.geonode.org/en/latest/deploy/install.html
• Detailed configuration for production environment: http://docs.geonode.org/en/latest/deploy/production.html
GeoNode administration support

• The general GeoNode documentation site is in: http://docs.geonode.org/en/latest/

• Go to http://geonode.org/communication/ to subscribe to user and developer mailing lists (we recommend that you join both). You will need a Google account to do this