AMsoil

The glue for Aggregate Manager developers

(radio edit)

OFELIA

OpenFlow in Europe:
Linking Infrastructure and Applications

OFELIA FP7 project in numbers

3 Years, ending Sept. ’13

6.5 € Budget

4.45 € Funding

10 Partners

12 after the 1st OpenCall

17 after the 2nd OpenCall

7 OpenFlow islands

phases

2010

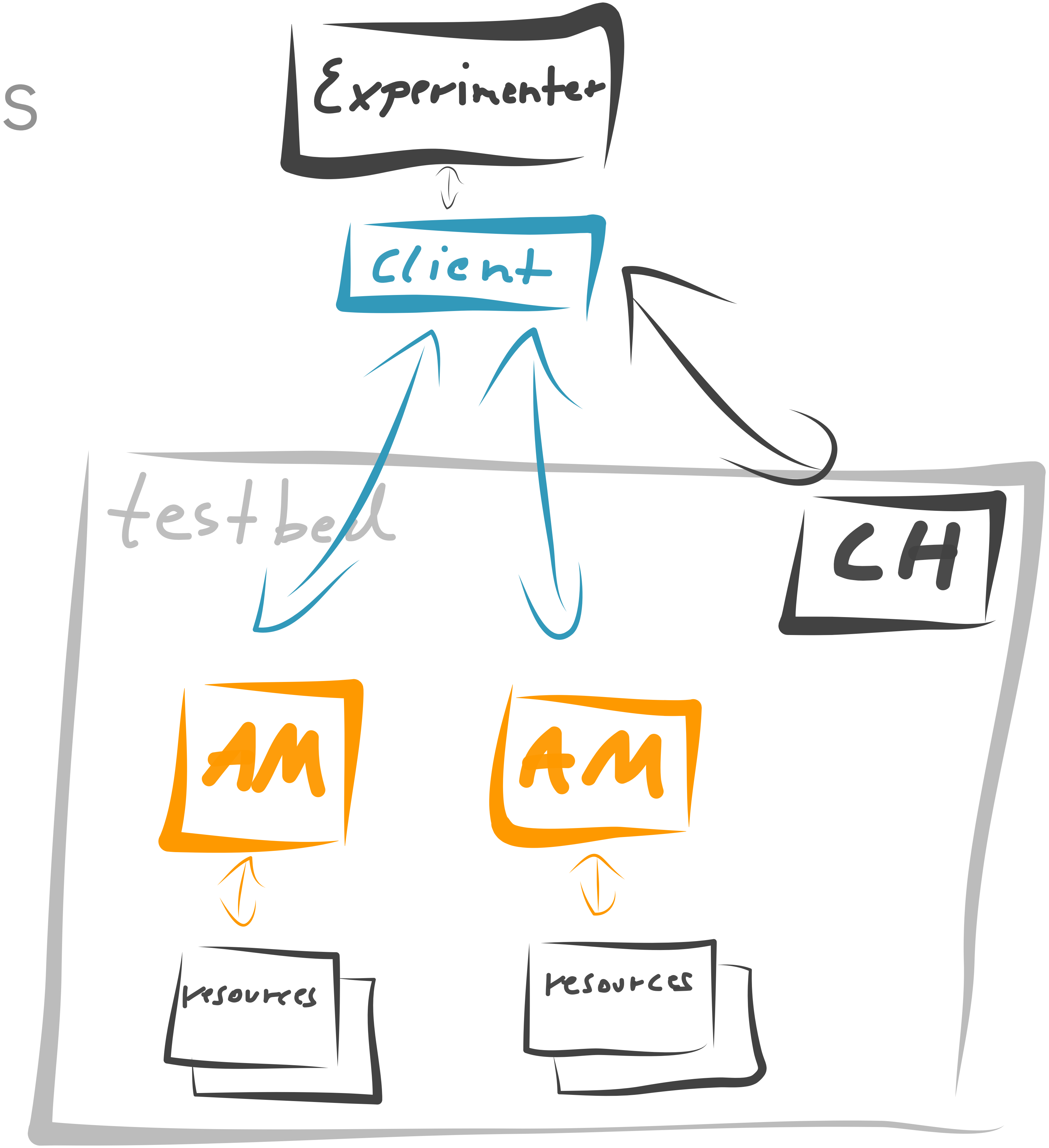
2013

AMsoil

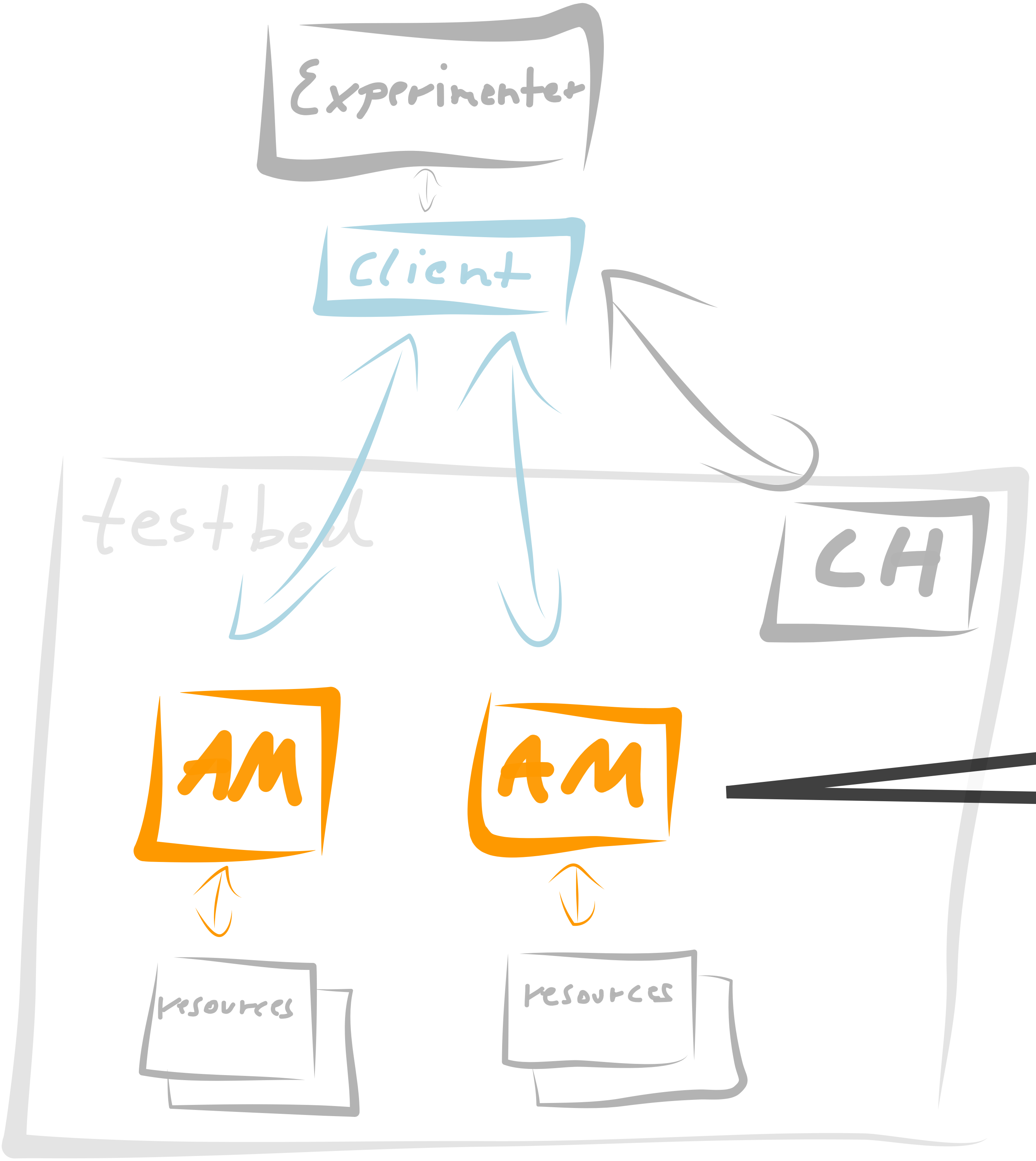
The glue for Aggregate Manager developers

(radio edit)

components



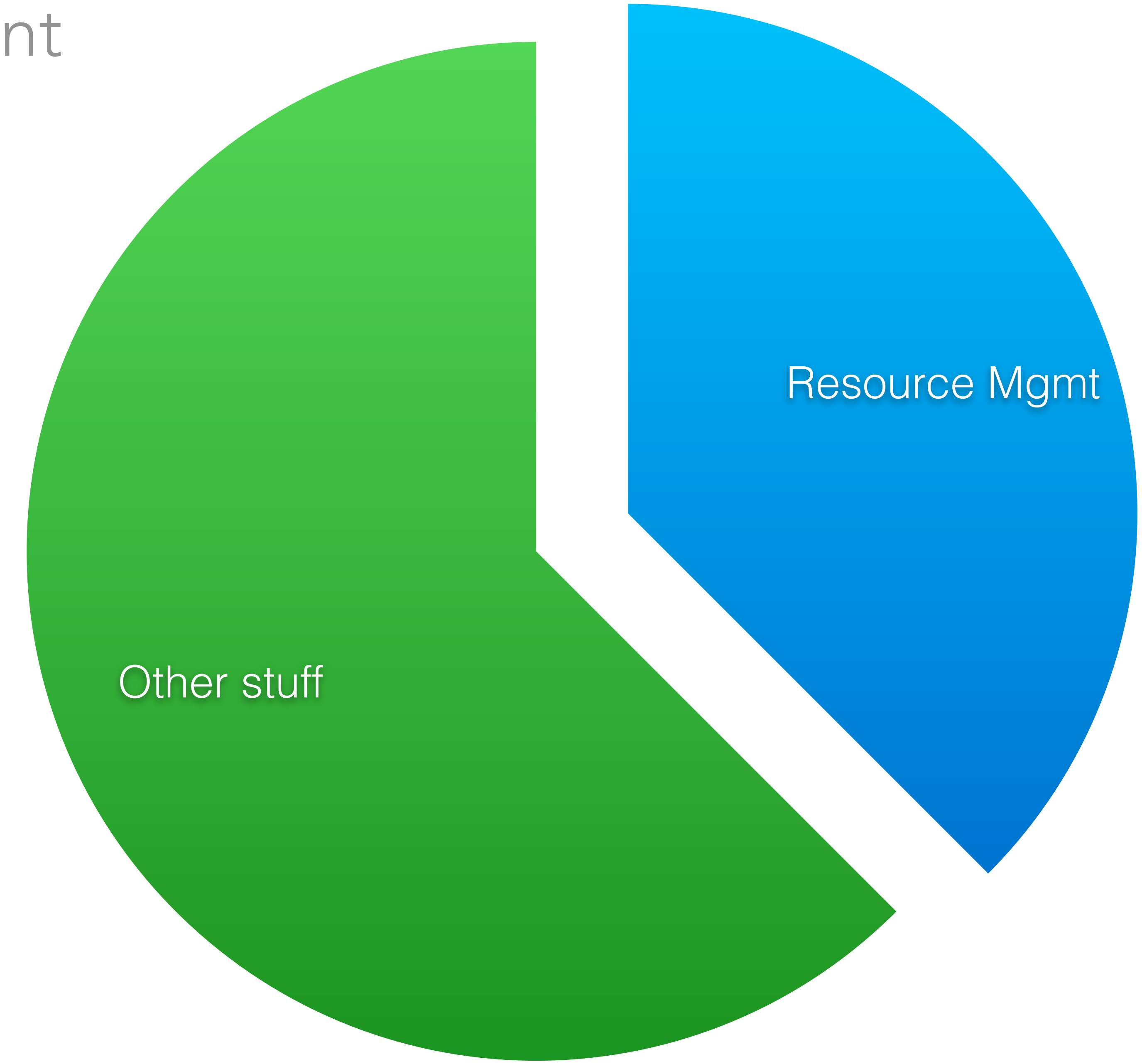
AMsoil?



Someone needs to implement this!

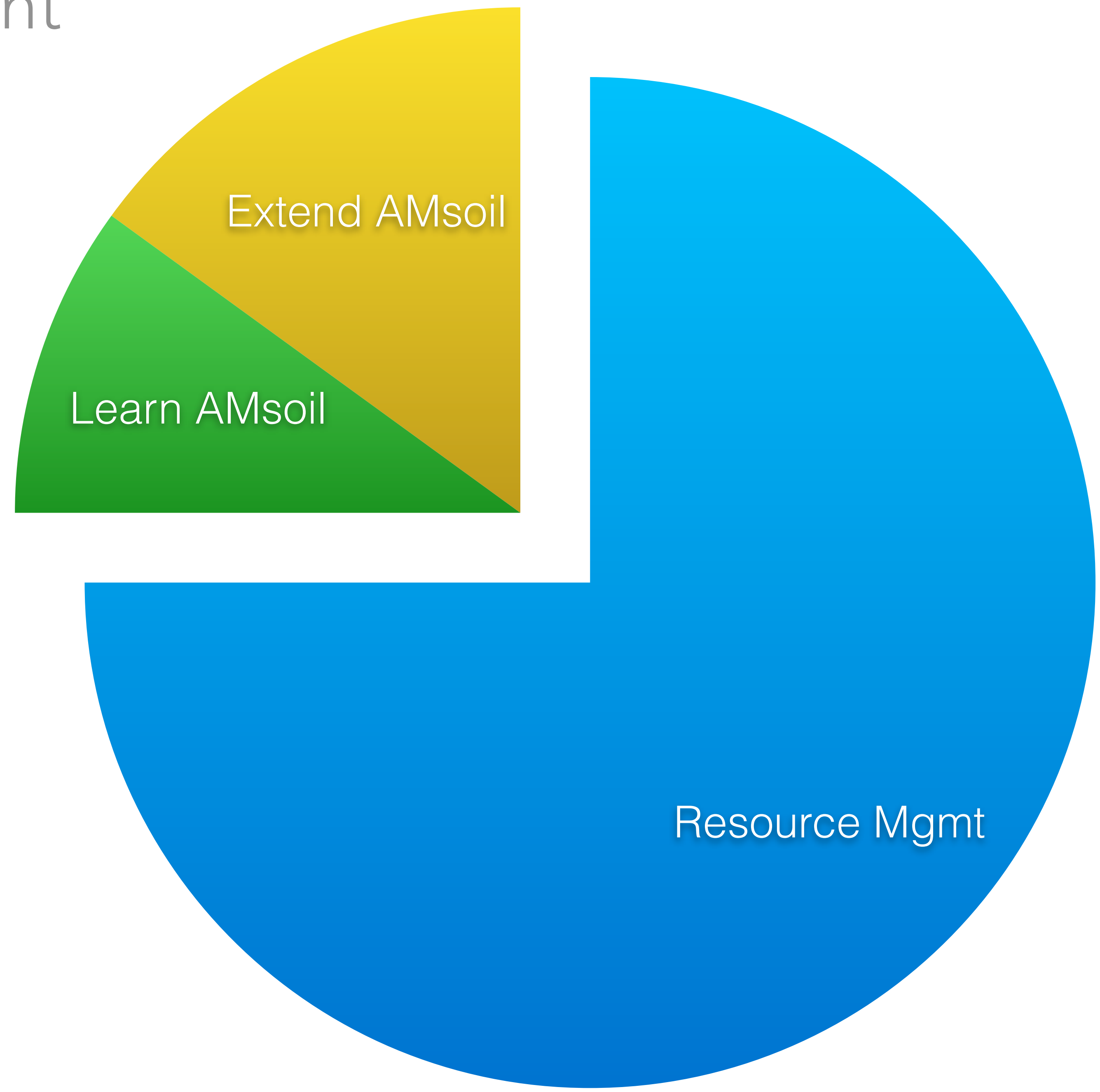
motivation

AM development without AMsoil



motivation

AM development
with AMsoil



AMsoil?

> AMsoil is a light-weight framework for creating Aggregate Managers in test beds.
>
> AMsoil is a pluggable system and provides the necessary glue between RPC-Handlers and Resource Managers . Also it provides helpers for common tasks in AM development.

what we need to know

- how plugins work
- what plugins you need to develop
- what else AMsoil supports

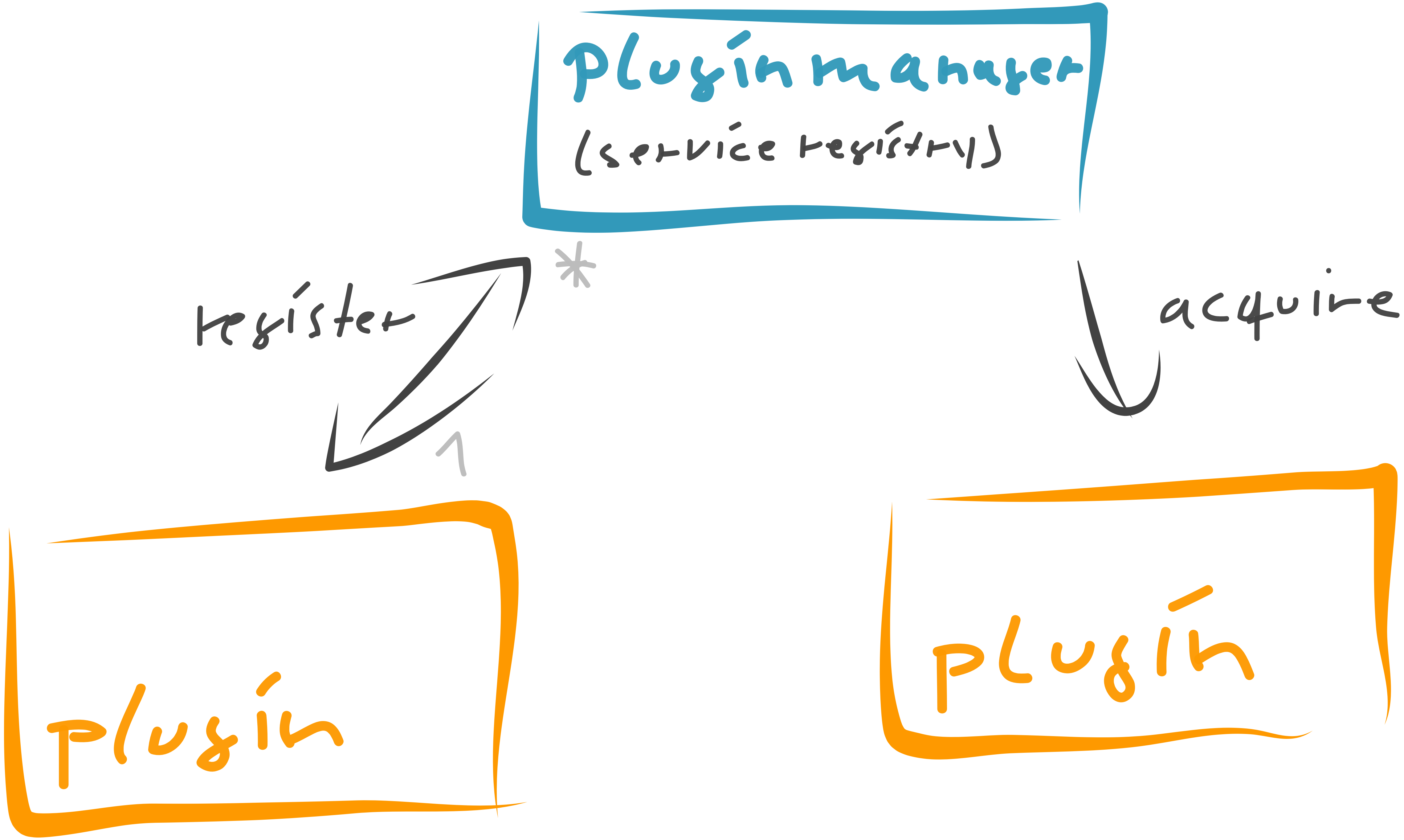
- how a GENI testbed works

why plugins?

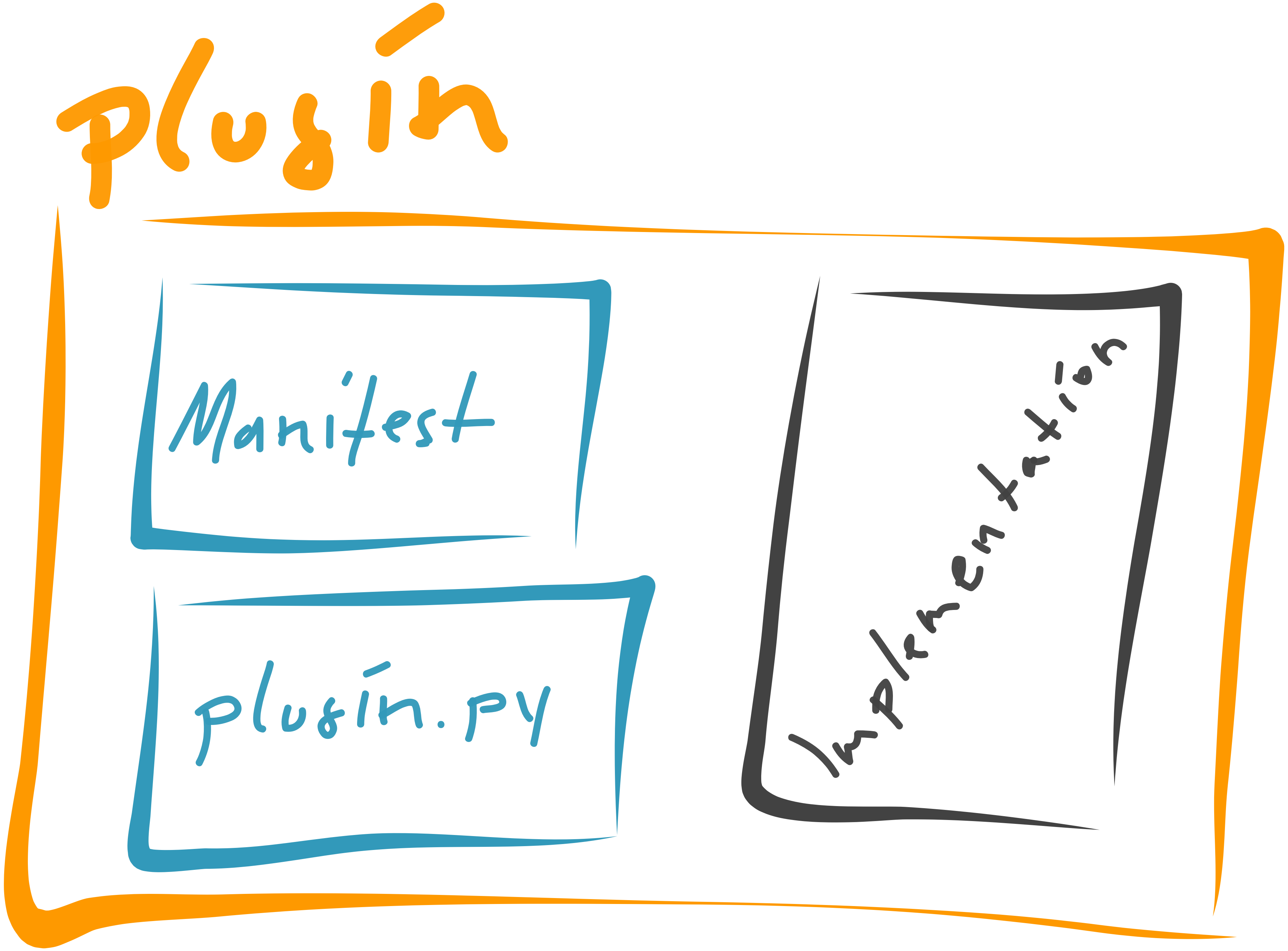
- **Selection**
  An administrator can add/remove plugins/functionality.
- **Exchangeability**
  The interface remains, but the implementation be changed.
- **Clarity**
  Provide a set of services and hide the details behind.
- **Encapsulation**
  Protect implementations from other developers.

plugin

register and use plugins



under the hood



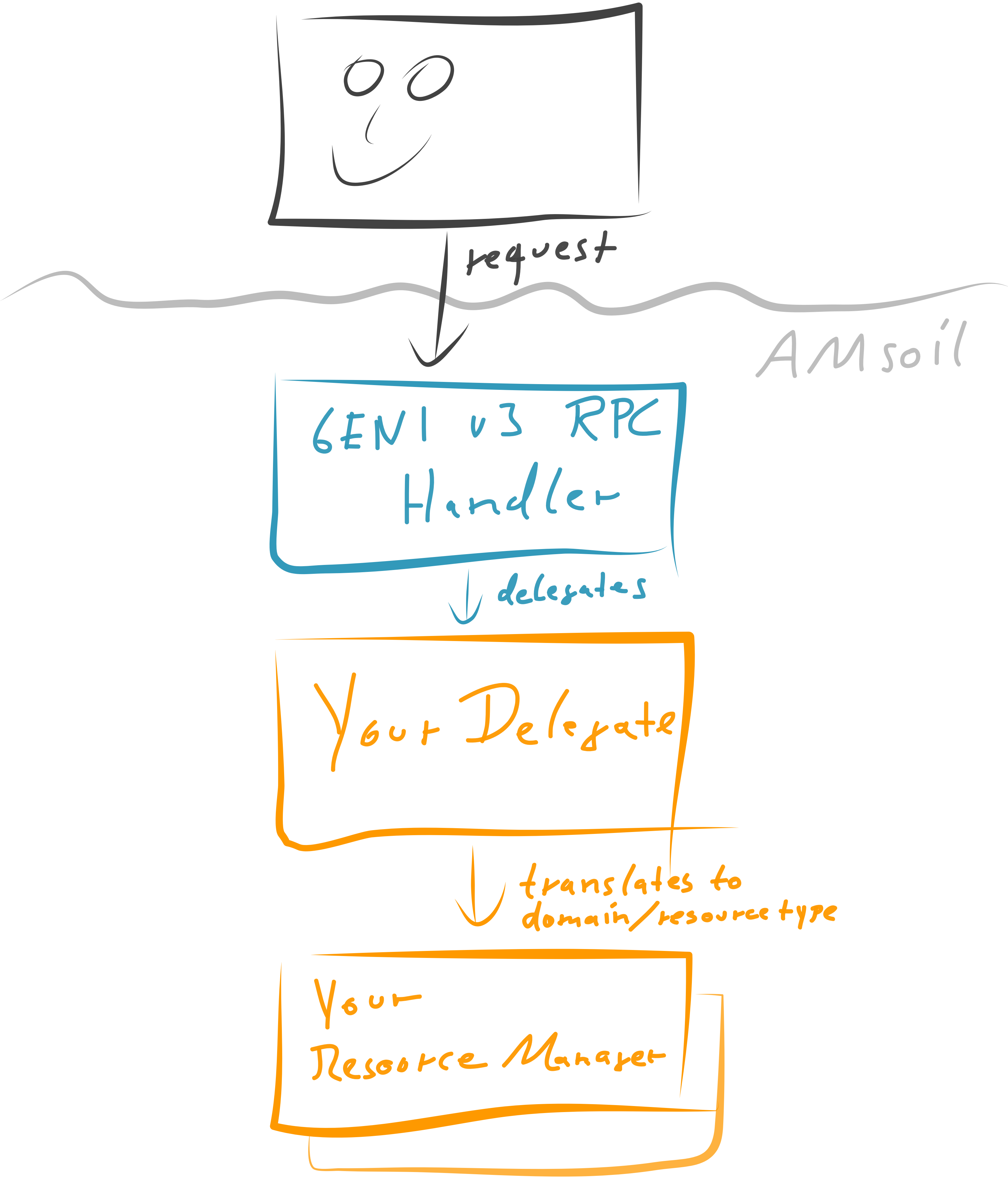
implement a plugin

- create a new folder in plugins
- create the manifest.json
- create the plugin.py
    - write a setup() method
- register your services

handling the AM API

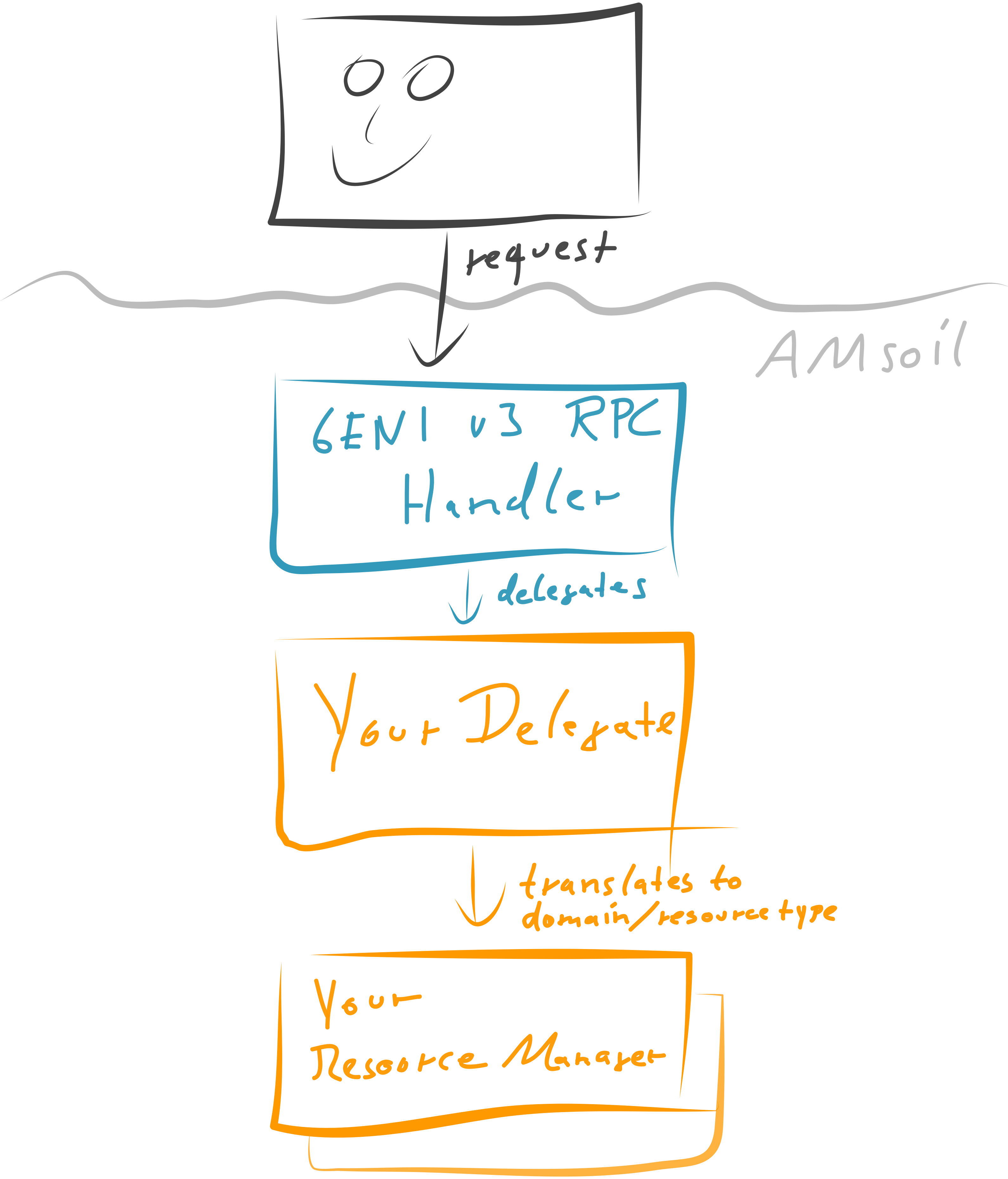
or how to develop a GENI AM API v3 Aggregate

getting the requests



GENI

getting the requests



why so complicated?

interfaces

- **Delegate**
  Should derive from DelegateBase and overwrite the methods prescribed (e.g. list_resources, allocate, ...).
- **Resource Manager**
  You make up the interface!
  The methods, attributes, parameters are domain-specific and depend on the resource type being handled.

interfaces

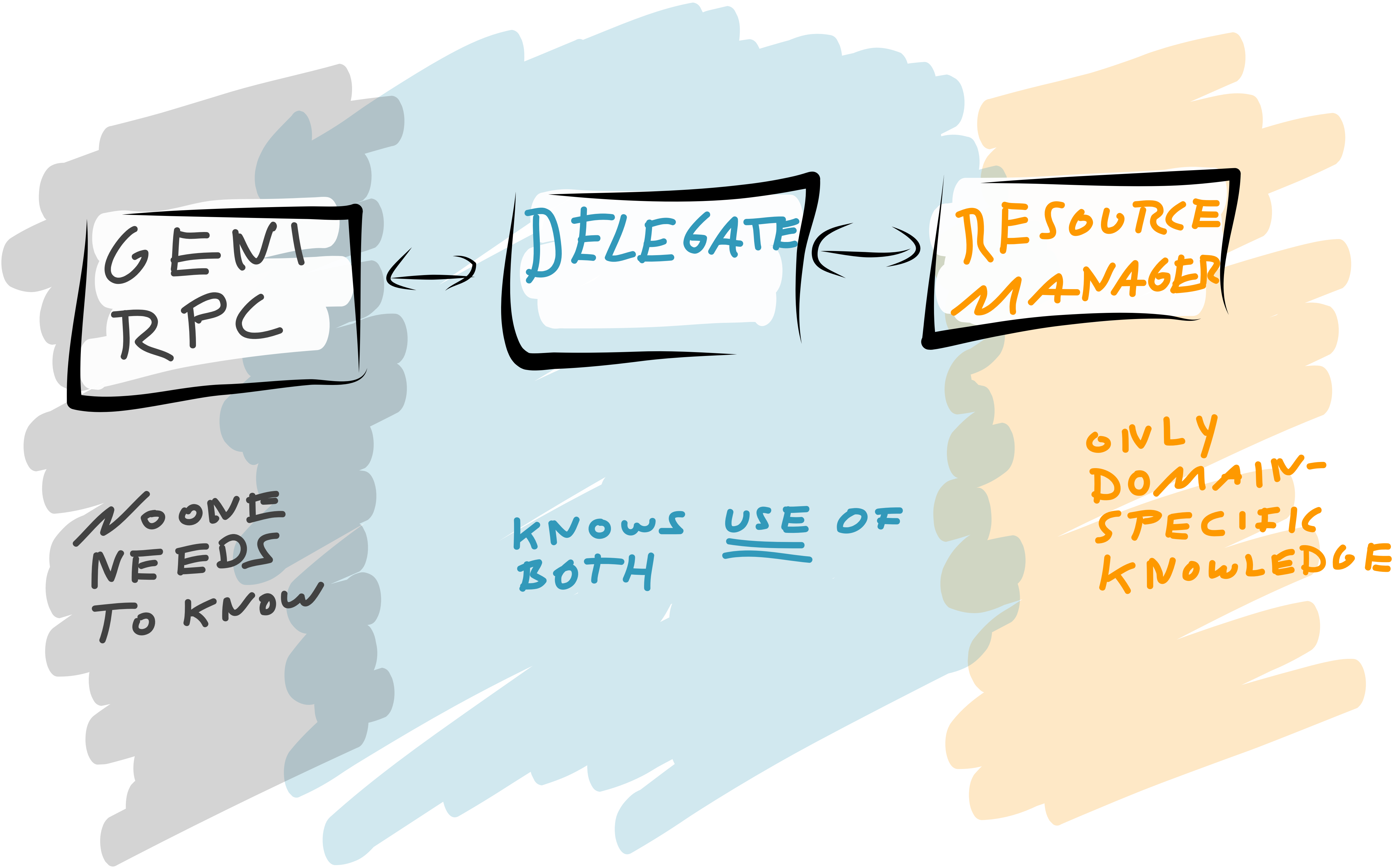
- **Delegate**
  Should derive from DelegateBase and overwrite the methods prescribed (e.g. list_resources, allocate, ...).

FIXED

- **Resource Manager**
  You make up the interface!
  The methods, attributes, parameters are domain-specific and depend on the resource type being handled.

FLEXIBLE

needed knowledge



hands on tips

Let’s see how we can make our life even easier.

configuration

anywhere.py

```python
import amsoil.core.pluginmanager as pm
config = pm.getService("config")      # get the service
myvalue = config.get("mygroup.mykey") # retrieve a value
config.set("mygroup.mykey", myvalue)  # set a value
```

persistence

SQLAlchemy tutorial

7900 words

vs.

Need to know

926 words

Persistence

Value objects

Bootstrap

Create a table

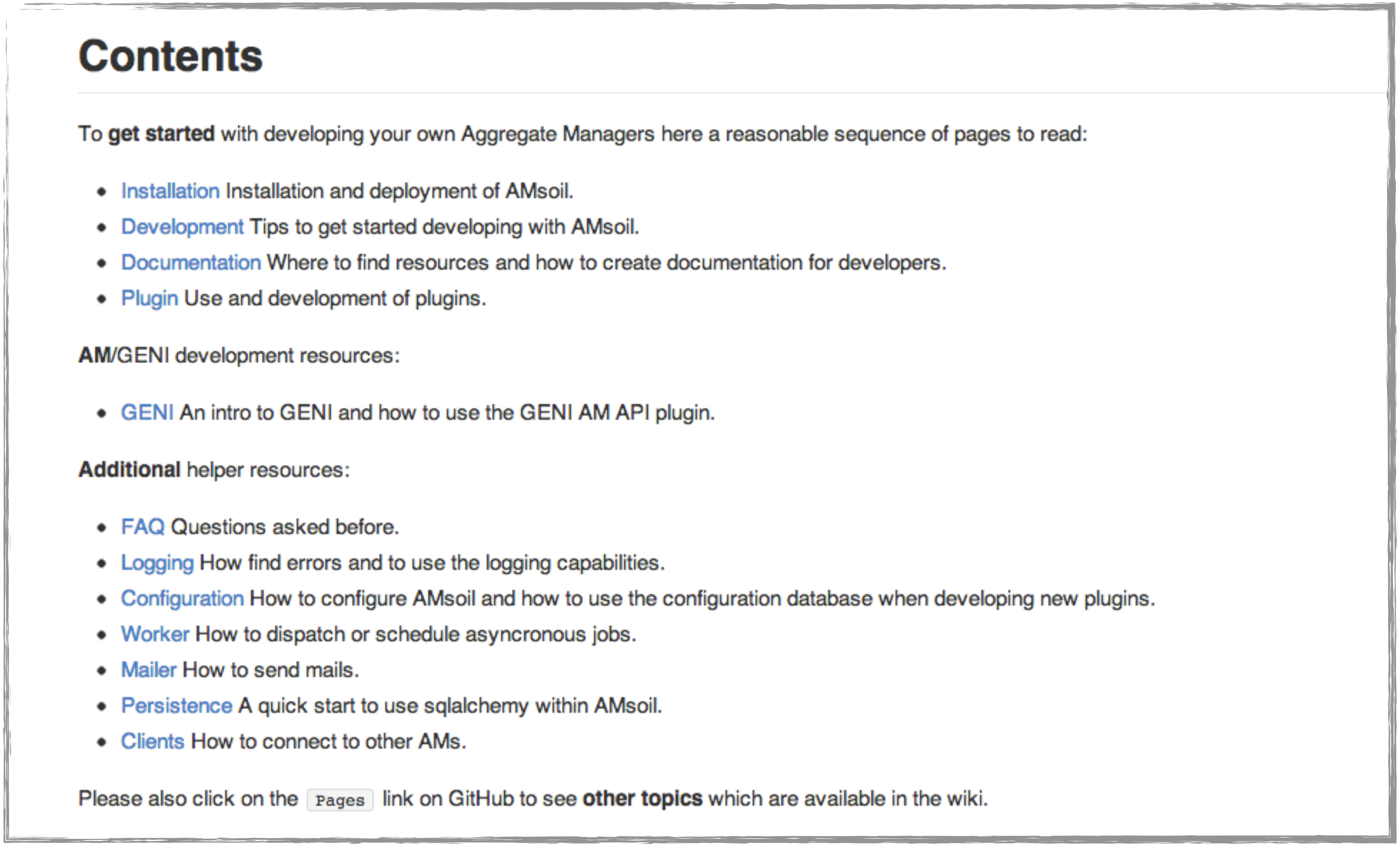
Query and changes

Objects in sessions

Expunge

More info

persistence



Contents

To **get started** with developing your own Aggregate Managers here a reasonable sequence of pages to read:

- Installation Installation and deployment of AMsoil.
- Development Tips to get started developing with AMsoil.
- Documentation Where to find resources and how to create documentation for developers.
- Plugin Use and development of plugins.

AM/GENI development resources:

- GENI An intro to GENI and how to use the GENI AM API plugin.

Additional helper resources:

- FAQ Questions asked before.
- Logging How find errors and to use the logging capabilities.
- Configuration How to configure AMsoil and how to use the configuration database when developing new plugins.
- Worker How to dispatch or schedule asyncronous jobs.
- Mailer How to send mails.
- Persistence A quick start to use sqlalchemy within AMsoil.
- Clients How to connect to other AMs.

Please also click on the `Pages` link on GitHub to see **other topics** which are available in the wiki.

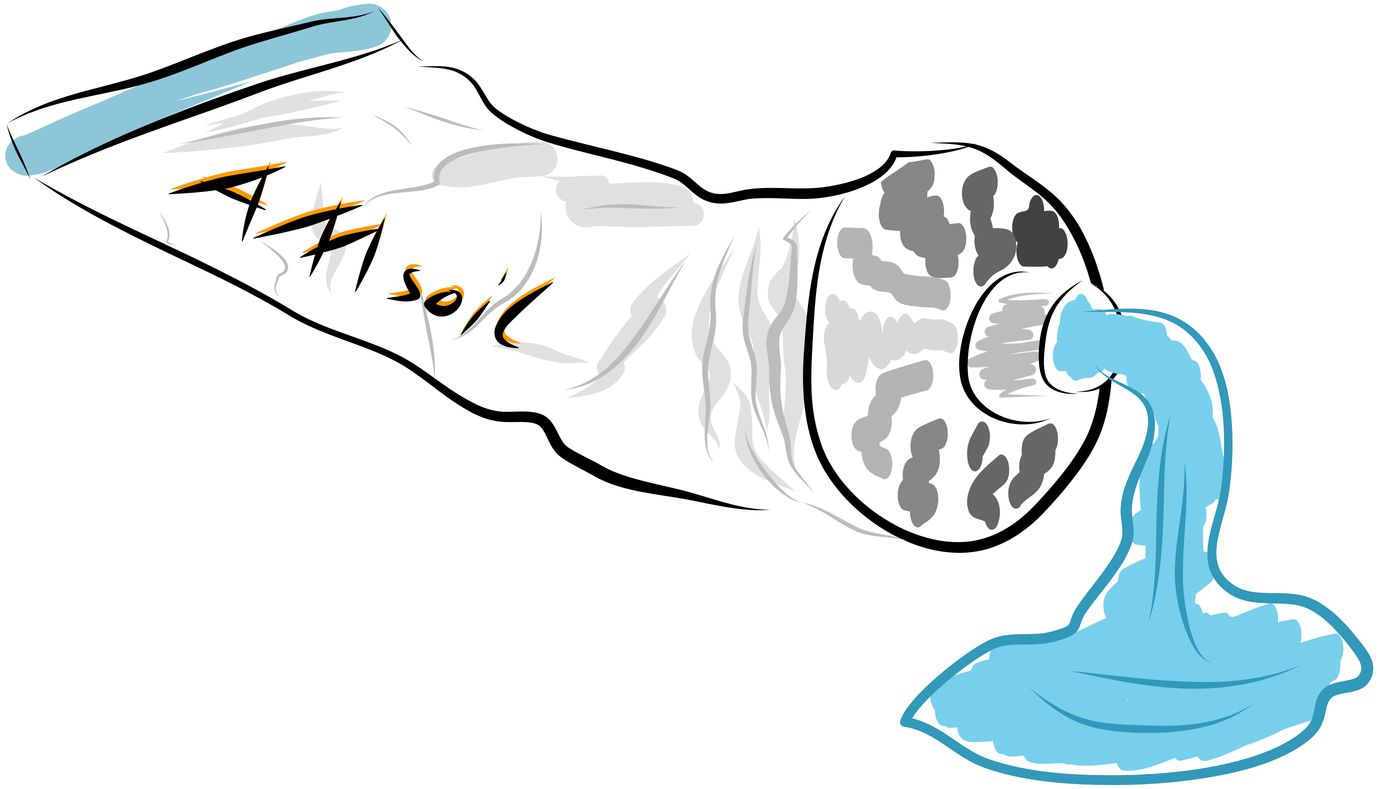
coming up

- GENI AM API v2
- Calendarization
- Policy Engine integration

- Clearinghouse

- lot’s more developers

check it out



clone the repository

https://github.com/fp7-ofelia/AMsoil.git

then **read** the wiki and **start** with the quick-start

https://github.com/fp7-ofelia/AMsoil/wiki