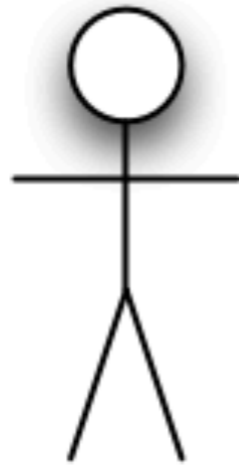




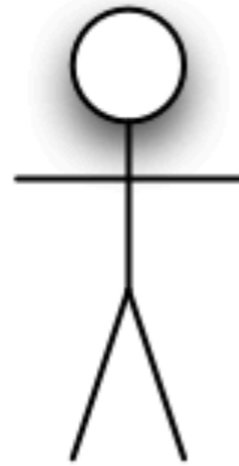
[open-emv.com](http://open-emv.com)



# The Team



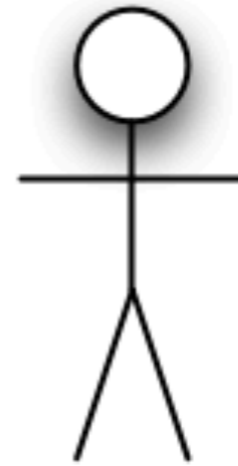
Steve  
EE



Eric  
EE



Jennifer  
SQL



Cris  
Python

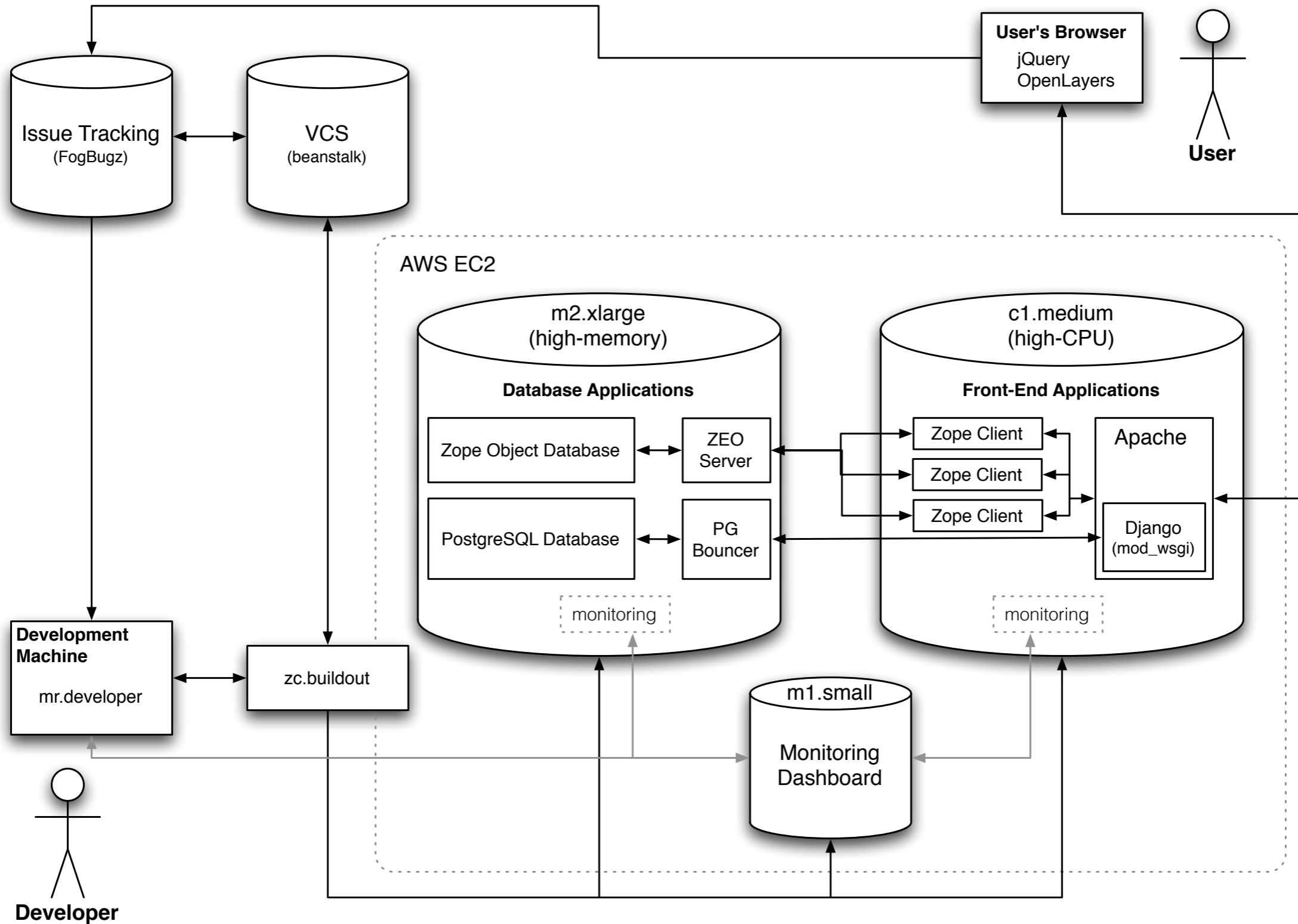


# Roadmap

- Introduction
  - Solution Overview
  - Energy Efficiency 101
  - Site Tour
- Under the Hood
  - Basic Machine Configuration (AMIs)
  - Software Configuration
  - Database Structure
  - The Web Application



# The Solution





# Why CA Energy Efficiency?



# World Leader

- 1.5 Million Jobs
- \$45 Billion in Payroll



How does that  
compare to the rest of  
the U.S?



# California...

- ... spends billions less on Electricity
- ... has a ratio of GDP to Energy Consumed 68% higher
- ... residents, per capita, pay lower utility bills





*"Imagine where the country could be if it were as efficient as California."*  
- F. Noel Perry, venture capitalist, founder of Next 10



# Energy Efficiency and EMV



# How Does CA do Energy Efficiency?

- CPUC established EE program cycle length and funding level
- CPUC provides \$ to utilities to implement EE Programs
- Utilities report program accomplishments to CPUC



# How Does CA do Energy Efficiency?

- CPUC hires independent EMV contractors to independently evaluate utility savings claims
- CPUC awards \$ to the utilities based on savings accomplishments
- CPUC releases data publicly via website



Rinse and Repeat



enabling data-driven energy efficiency

Search Site Search only in current section

You are here: Home

## What is OpenEMV?

The open source for measured performance

Energy efficiency decisions require data on the performance of energy efficiency measures. Measured energy savings data from program evaluation, measurement and verification (EM&V) studies exist in many contexts. OpenEMV's purpose is to make available all public and open EM&V data to support energy efficiency policy and business decision making.

[Tell me more about OpenEMV](#)

## What data do we have?

More than 3.7 million records of energy efficiency data

The California public dataset contains over 3.7 million records, representing efficiency projects and measure installations across the state. OpenEMV presents these data in an interactive format. You can use our tools to gain a better understanding of the magnitude of energy and cost savings, the geographic distribution of the savings, and the kinds of technologies and buildings involved.

[Show me the data](#)

## Why Open?

Public data and engaged community

Our goal is to serve both as a community space and as a source of technical data. By adopting the open source model, we assure that any party can contribute to and benefit from the data.

[Tell me more about open data](#)

## Why EM&V?

Solid data for solid decisions

Across the globe, enormous amounts of money are being spent collecting energy efficiency data. If all these data were openly available, they could be useful to efficiency program designers, policy makers, and other interested parties. A single, shared repository and analysis tools will improve the availability of measure performance data, lower program costs and provide support for better decisions.

[Tell me more about EM&V](#)

## Why California?

The largest efficiency program evaluation effort in history

In 2006-2008, California state government and utility companies worked together to generate an unprecedented dataset on energy efficiency and conservation measure performance. The data were made public as a matter of policy. OpenEMV is using this dataset as an example of the types of analyses that can be performed by anyone, anywhere, using open source tools.

[Tell me more about California](#)

### Upcoming

[SCALE 9x](#)



enabling data-driven energy efficiency

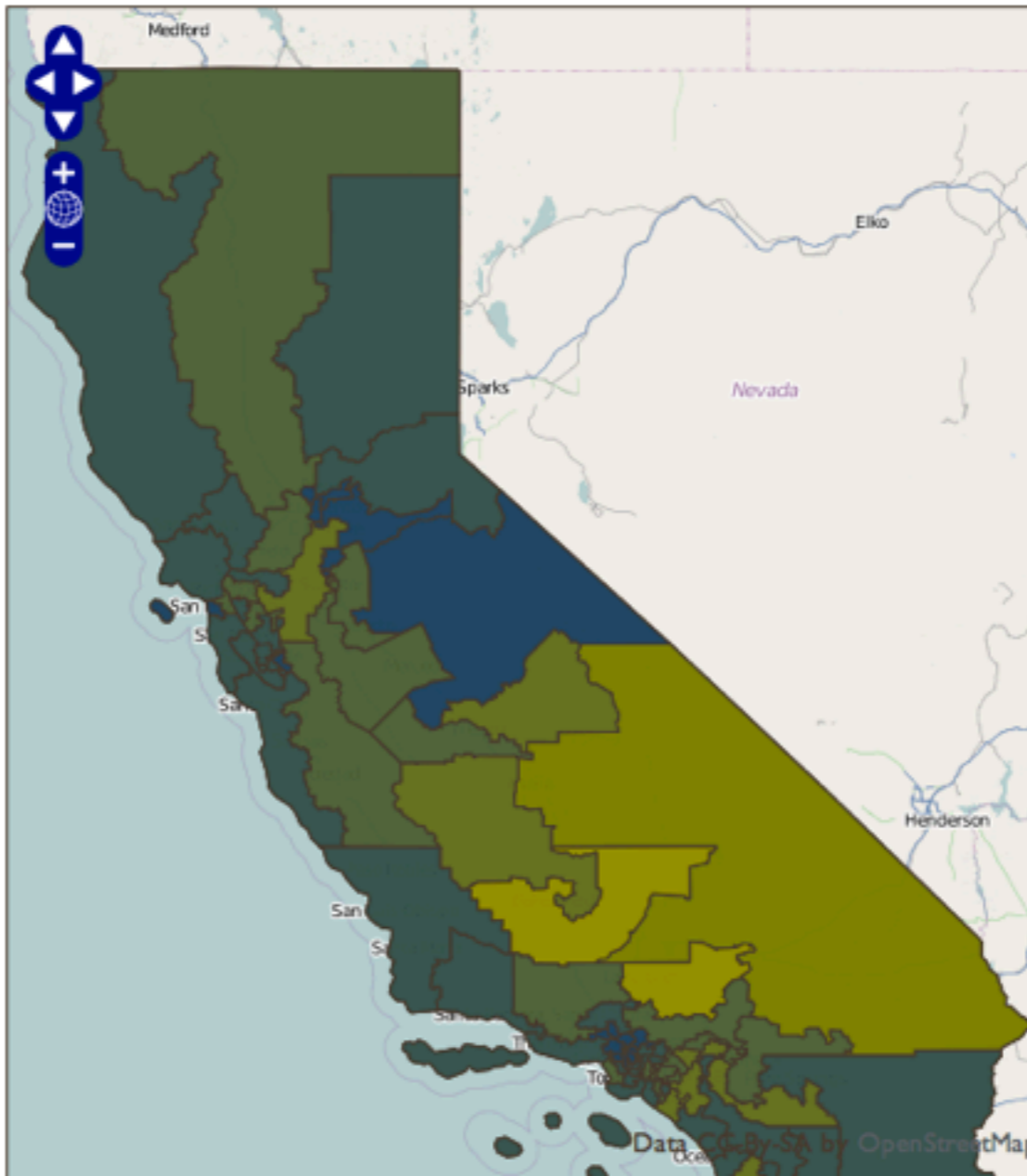
## California Energy Efficiency Program Data

Show me a map of firstyear evaluated kW savings by State House District

### Legend

- 0 - 15,100 kW
- 15,100 - 30,100 kW
- 30,100 - 45,100 kW
- 45,100 - 60,100 kW
- 60,100 - 75,100 kW
- 75,100+ kW

Download data: CSV, KML







enabling data-driven energy efficiency

## California Energy Efficiency Program Data

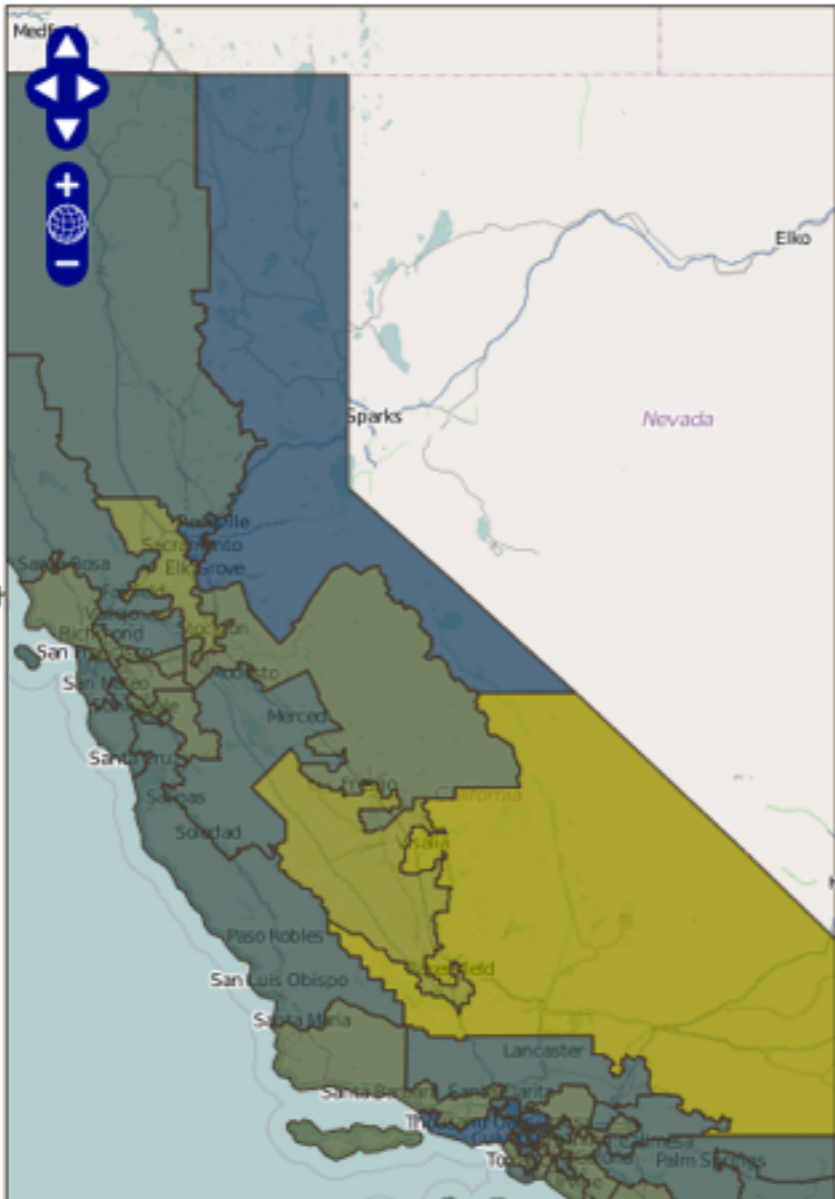
Show me a map of lifecycle evaluated kWh savings by State Senate District

compared to lifecycle total investment cost by State Senate District

### Legend

- 0 – 613,600,000 kWh
- 613,600,000 – 1,200,000,000 kWh
- 1,200,000,000 – 1,800,000,000 kWh
- 1,800,000,000 – 2,400,000,000 kWh
- 2,400,000,000 – 3,000,000,000 kWh
- 3,000,000,000+ kWh

Download data: CSV, KML



### Legend

- \$0 – \$32,600,000
- \$32,600,000 – \$62,500,000
- \$62,500,000 – \$92,400,000
- \$92,400,000 – \$122,300,000
- \$122,300,000 – \$152,100,000
- \$152,100,000+

Download data: CSV, KML







# Roadmap

- Introduction
  - Solution Overview
  - Energy Efficiency 101
  - Site Tour
- Under the Hood
  - Basic Machine Configuration (AMIs)
  - Software Configuration
  - Database Structure
  - The Web Application



# Roadmap

~~● Introduction~~

~~● Solution Overview~~

~~● Energy Efficiency 101~~

~~● Site Tour~~

● Under the Hood

● Basic Machine Configuration (AMIs)

● Software Configuration

● Database Structure

● The Web Application

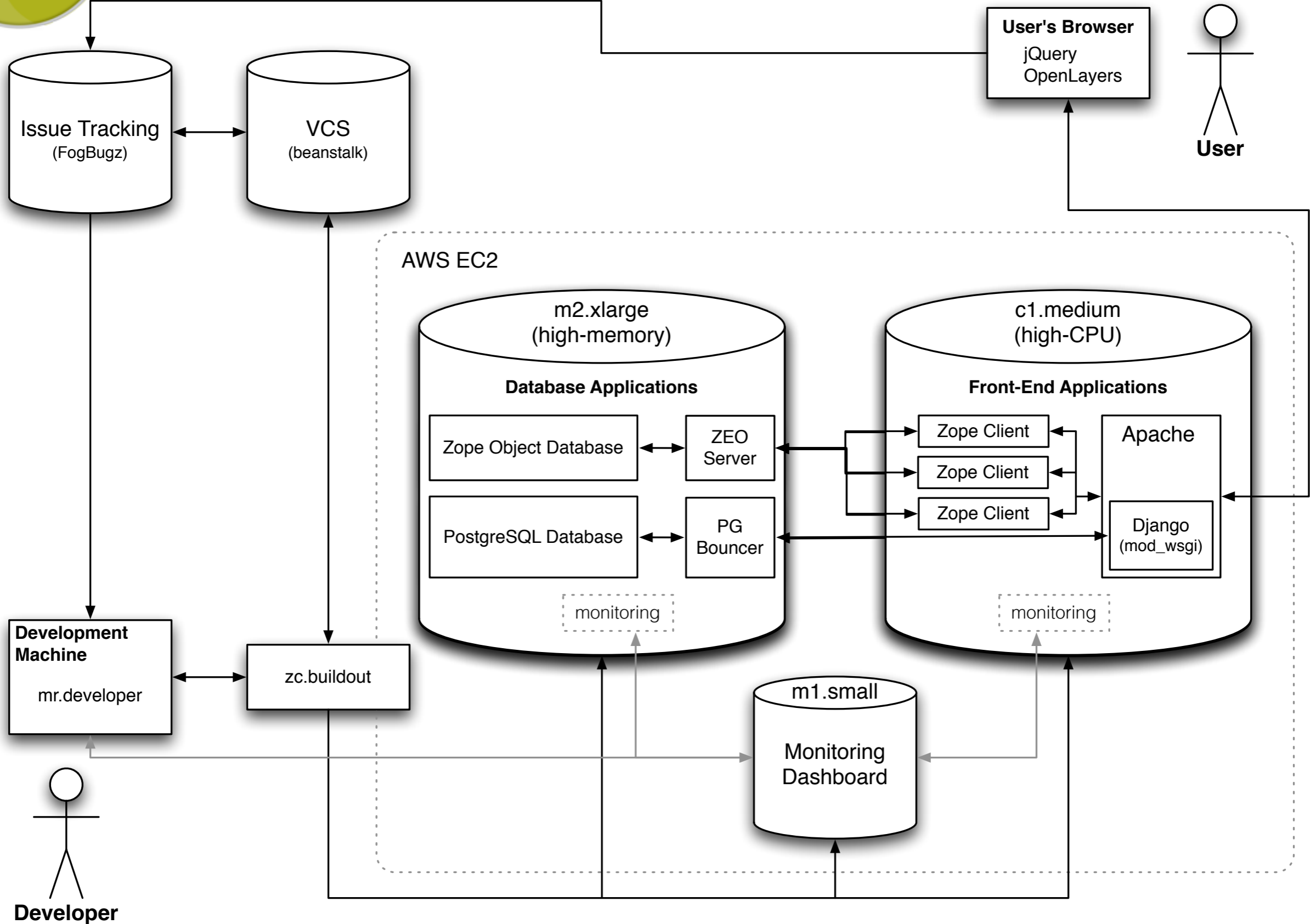


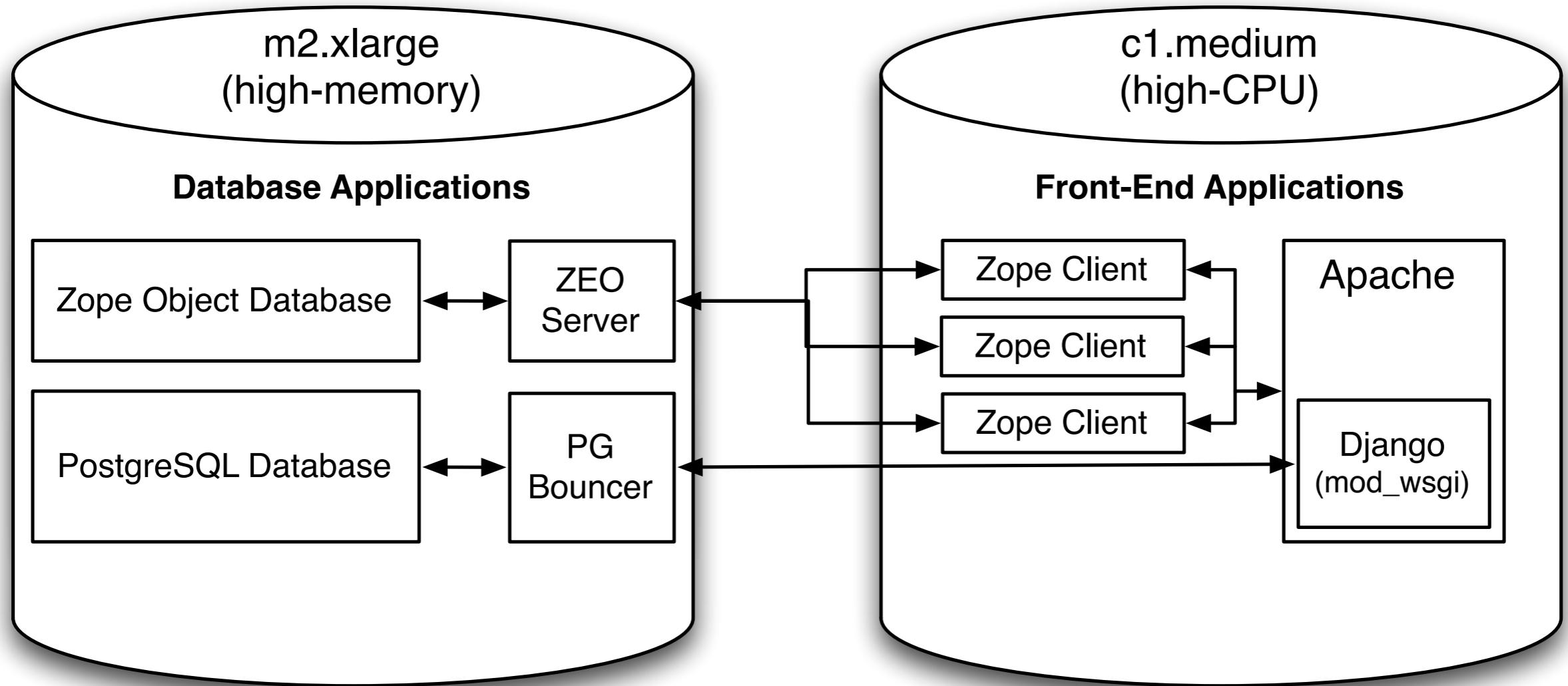
AMI



# AMI

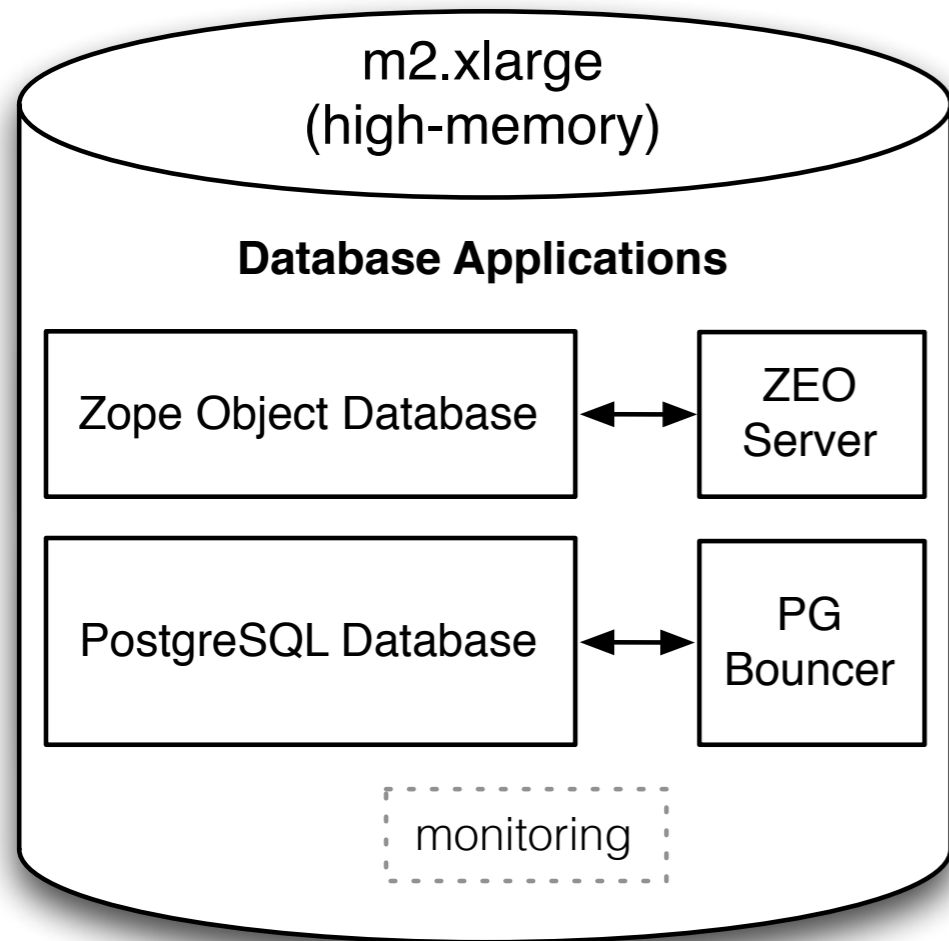
(**A** **m**azon **M**achine **I**mage)





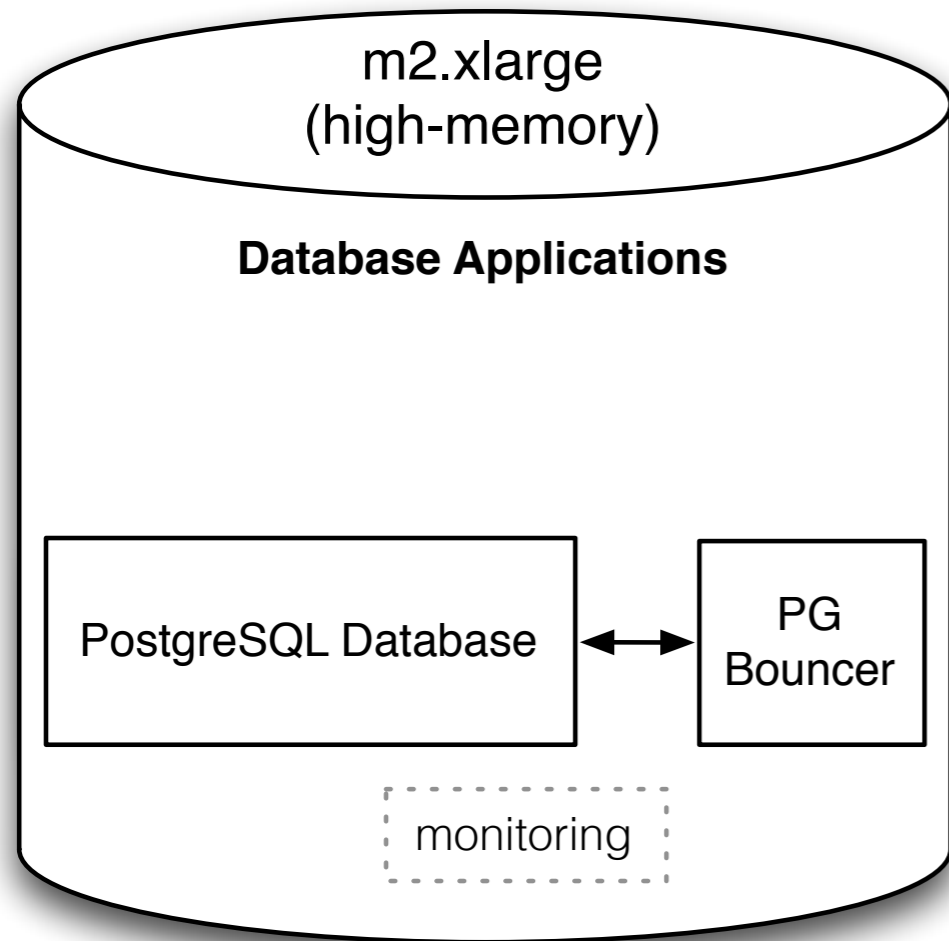


# Custom DB AMI





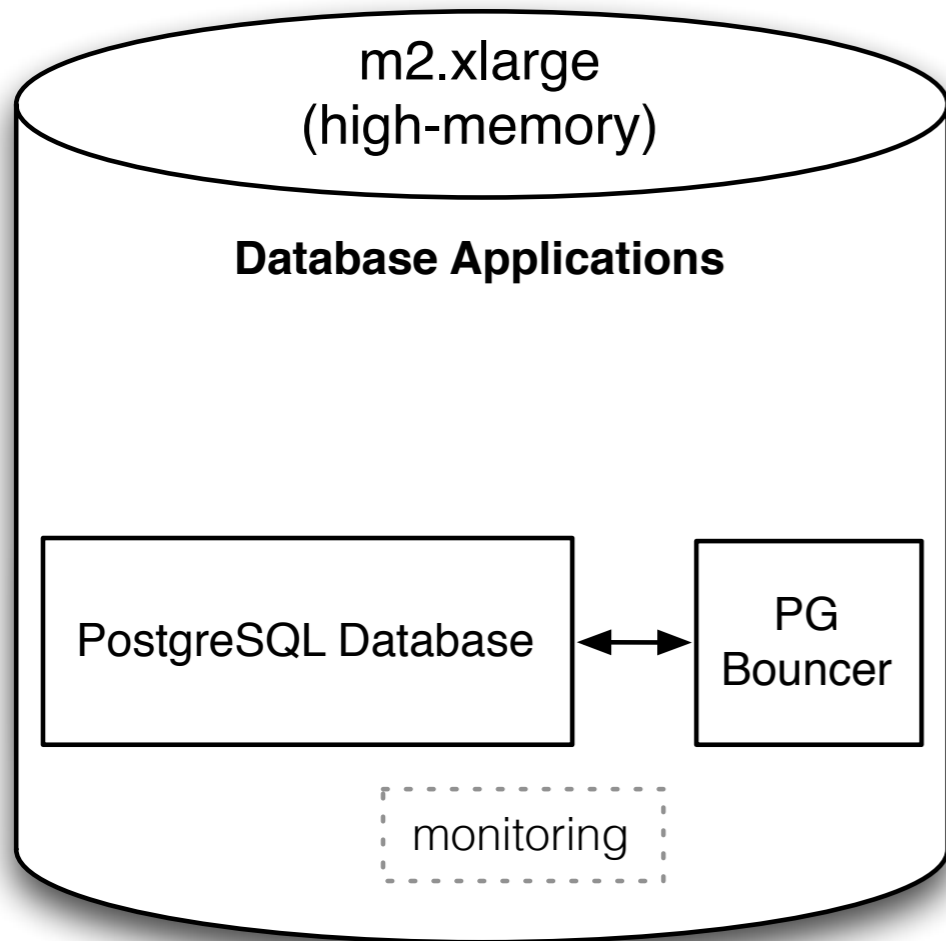
# Custom DB AMI







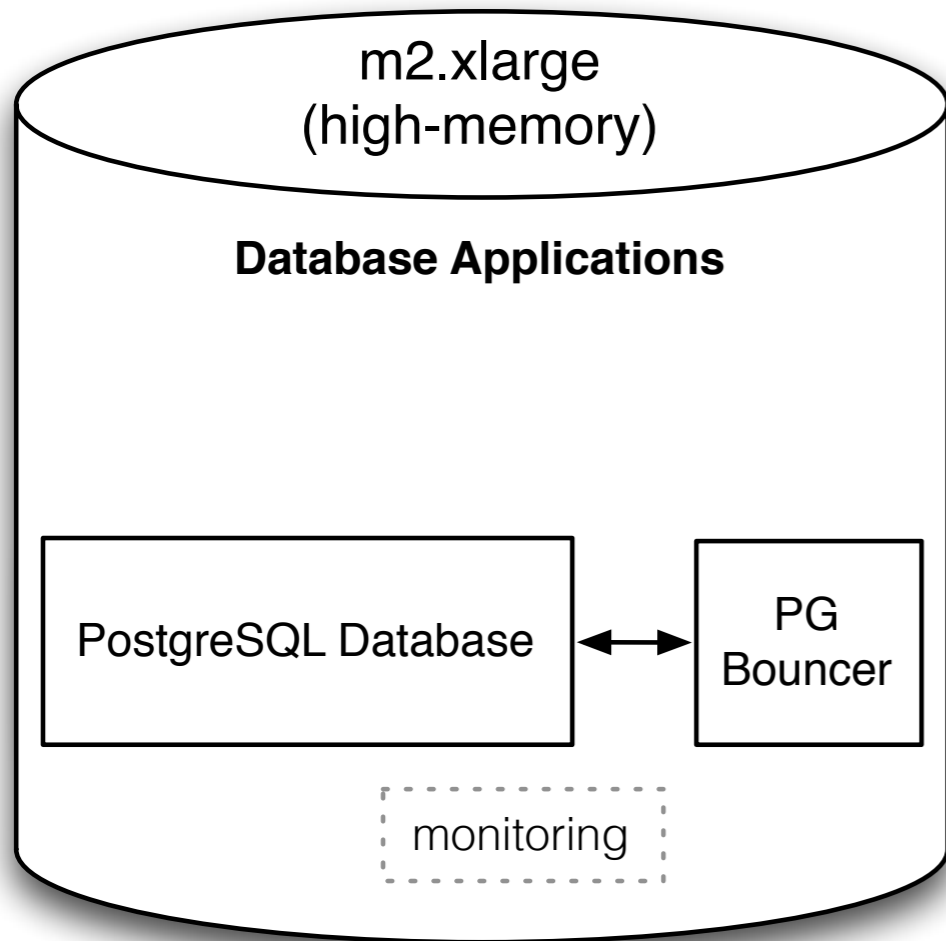
# Custom DB AMI



- Ubuntu 10.10 (Maverick)



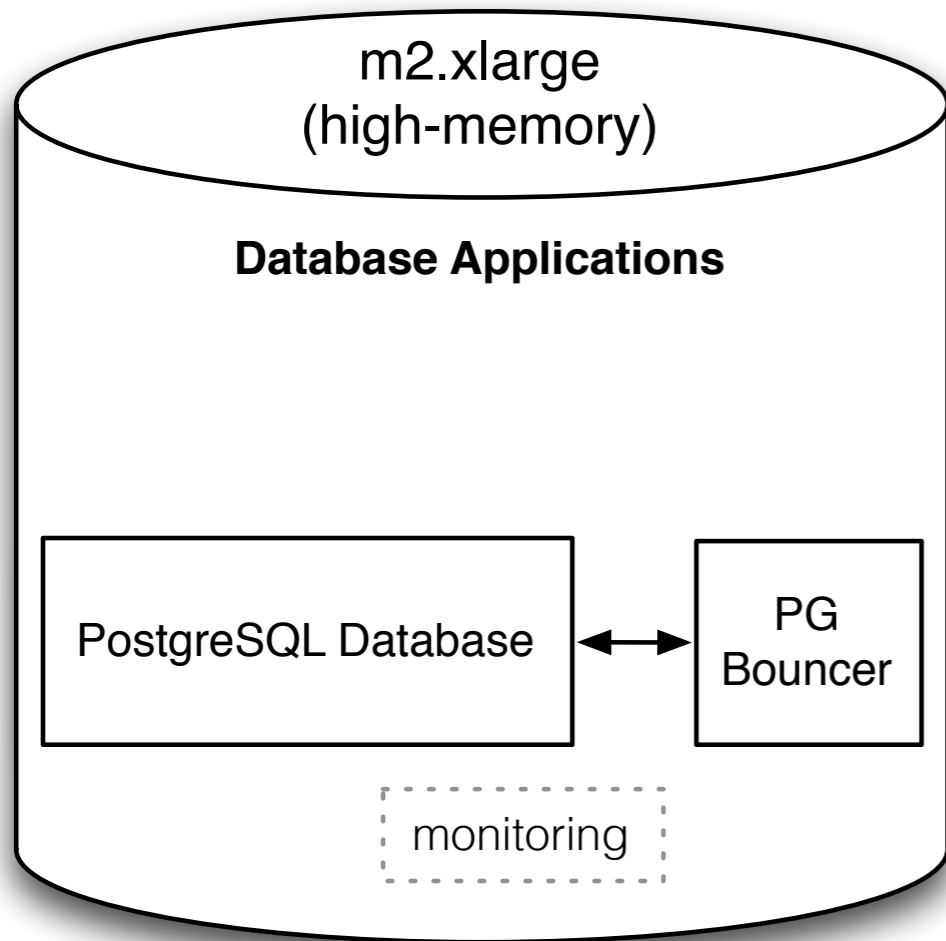
# Custom DB AMI



- Ubuntu 10.10 (Maverick)
- 64-bit



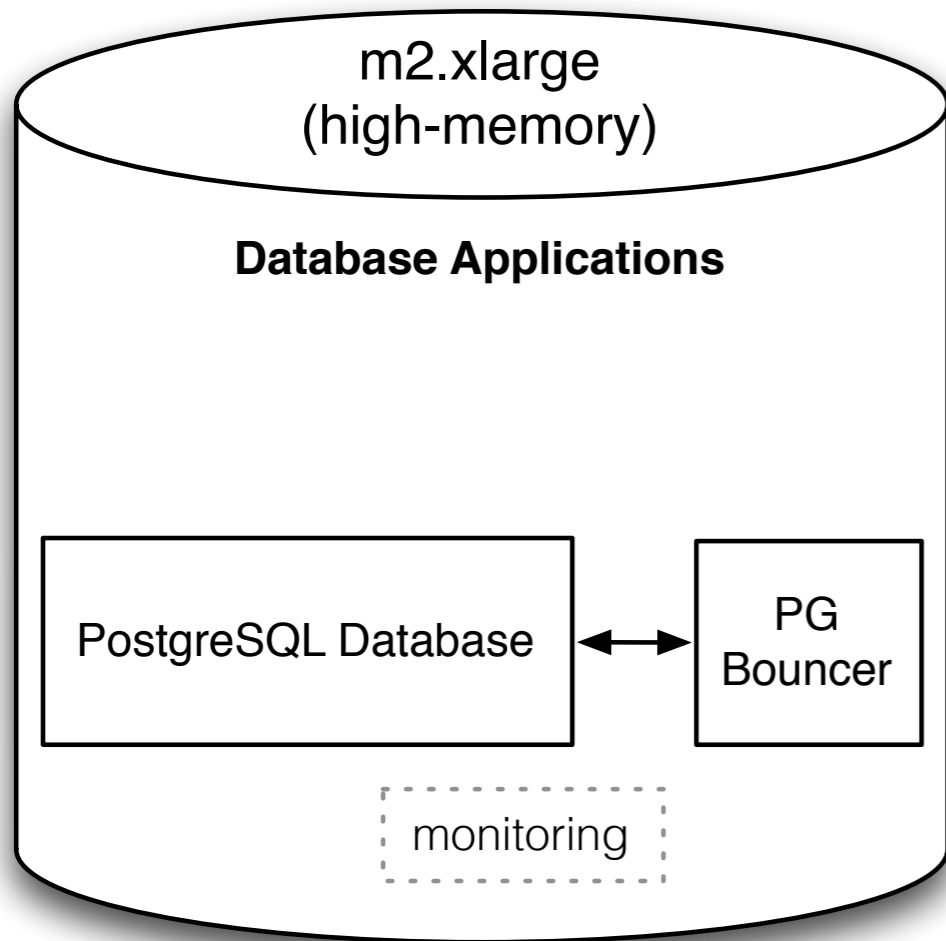
# Custom DB AMI



- Ubuntu 10.10 (Maverick)
- 64-bit
- Kernel Settings



# Custom DB AMI



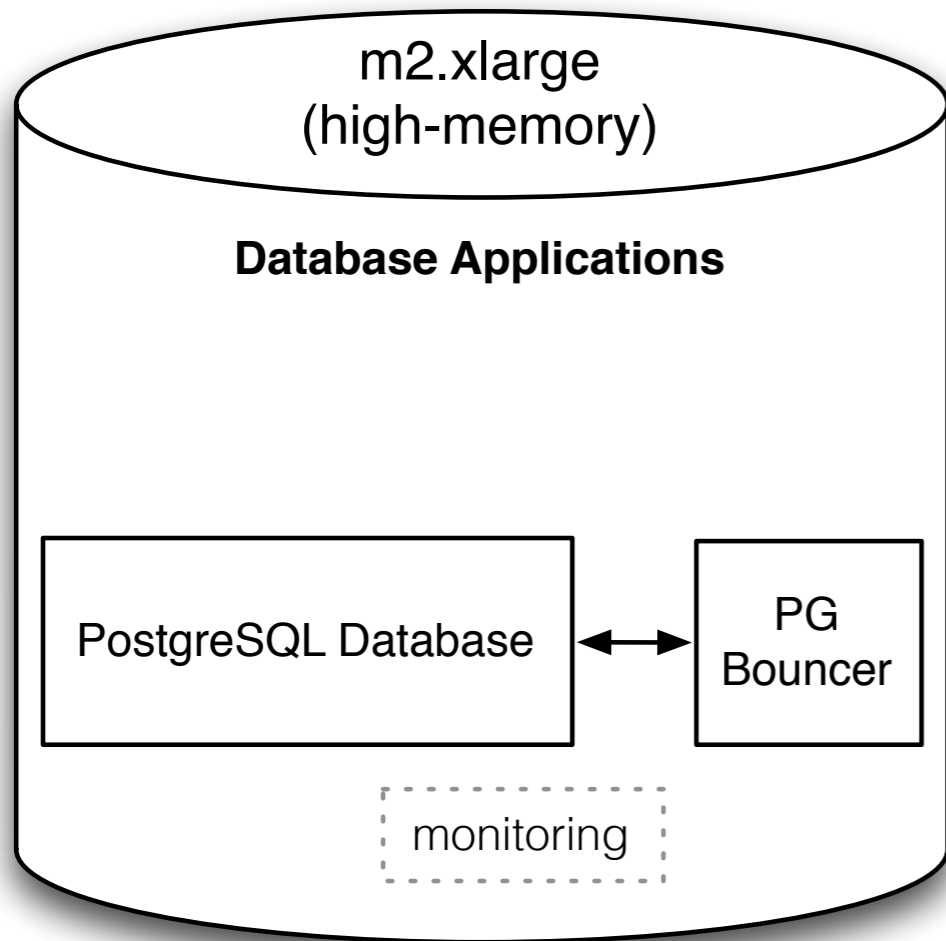
- Ubuntu 10.10 (Maverick)
- 64-bit
- Kernel Settings

```
kernel.shmmax  
= 8GB  
kernel.shmall  
= 8GB/4096
```

```
/etc/sysctl.conf
```



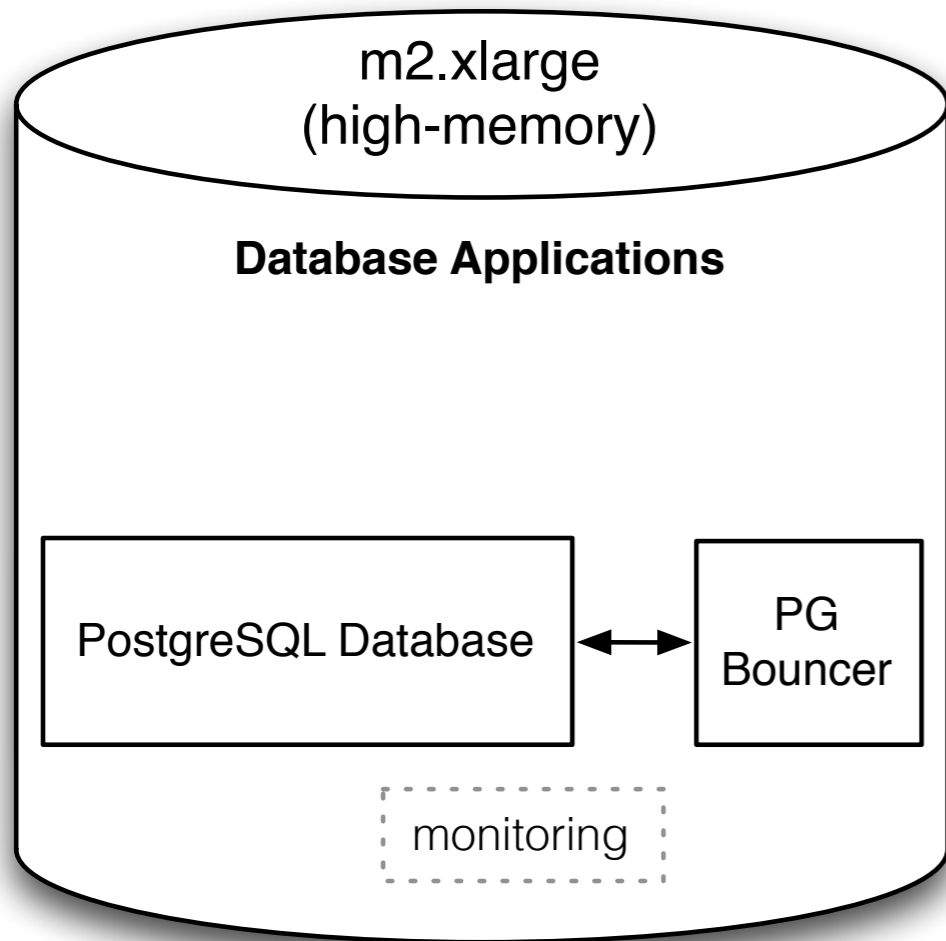
# Custom DB AMI



- Ubuntu 10.10 (Maverick)
- 64-bit
- Kernel Settings
- Spatial Database



# Custom DB AMI

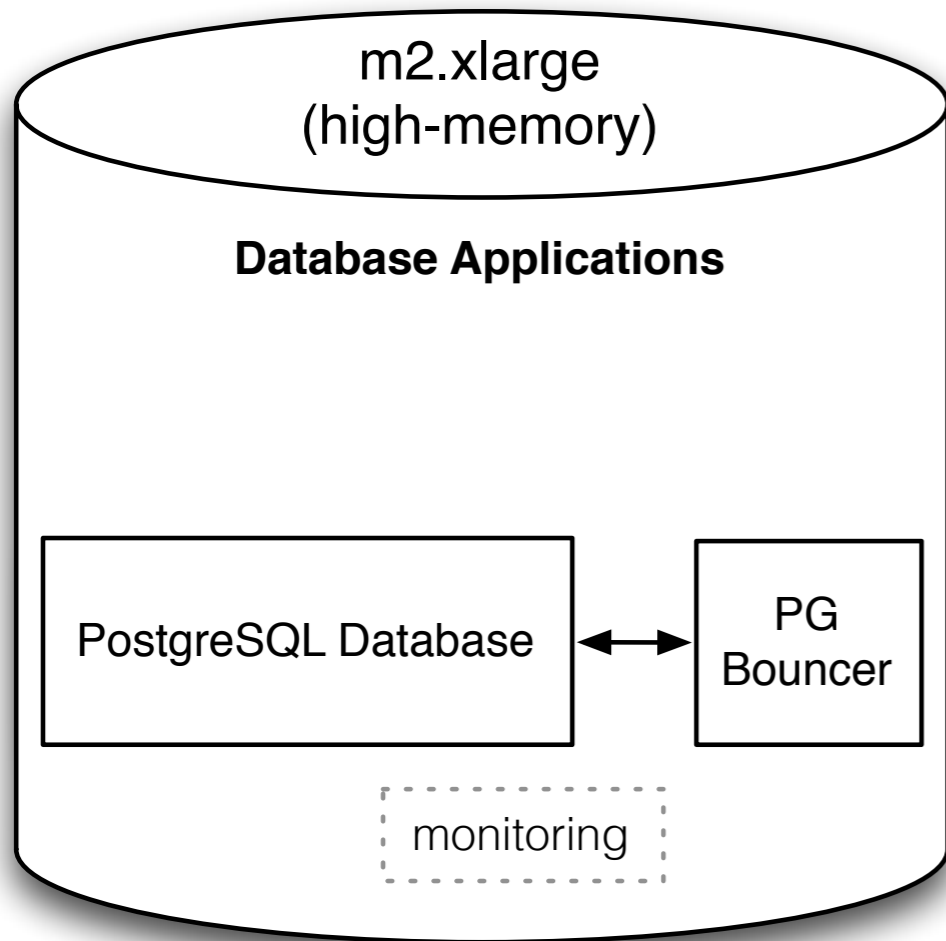


- Ubuntu 10.10 (Maverick)
- 64-bit
- Kernel Settings
- Spatial Database

```
apt-get install...  
PostgreSQL 9.0.3  
Geos 3.2.2  
GDAL 1.7.0  
manual install  
postgis 1.5.2
```



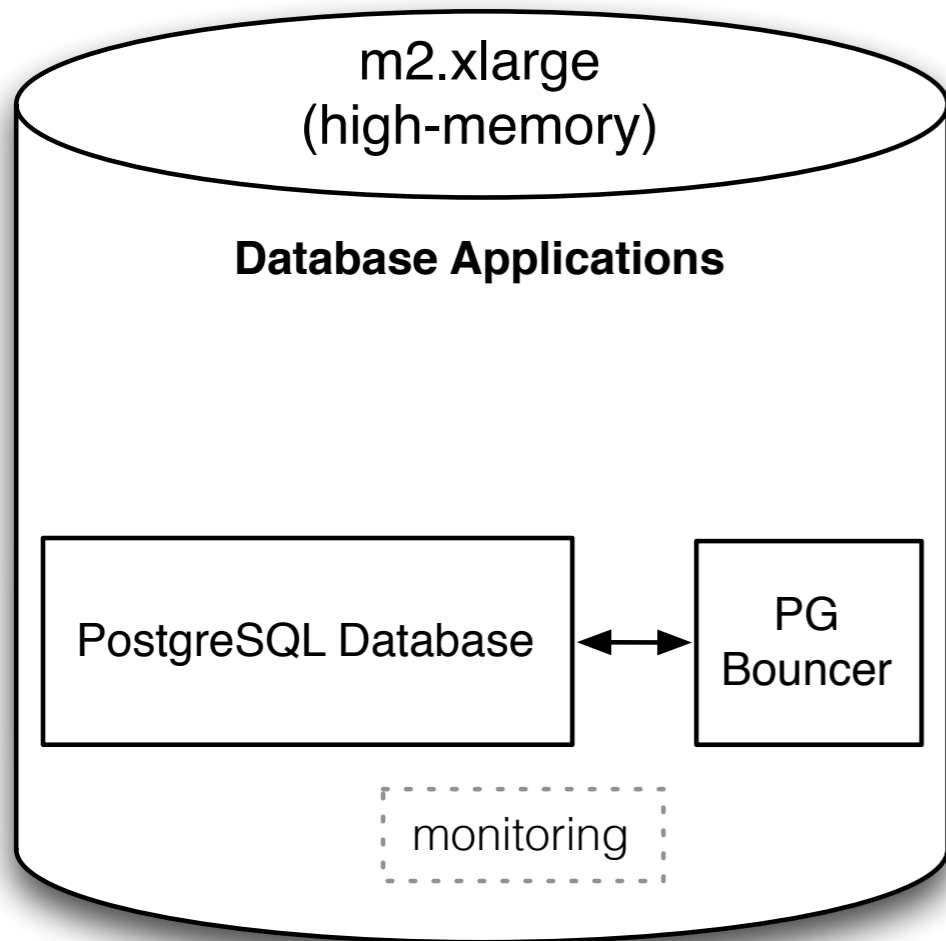
# Custom DB AMI



- Ubuntu 10.10 (Maverick)
- 64-bit
- Kernel Settings
- Spatial Database
- Connection Pooling



# Custom DB AMI



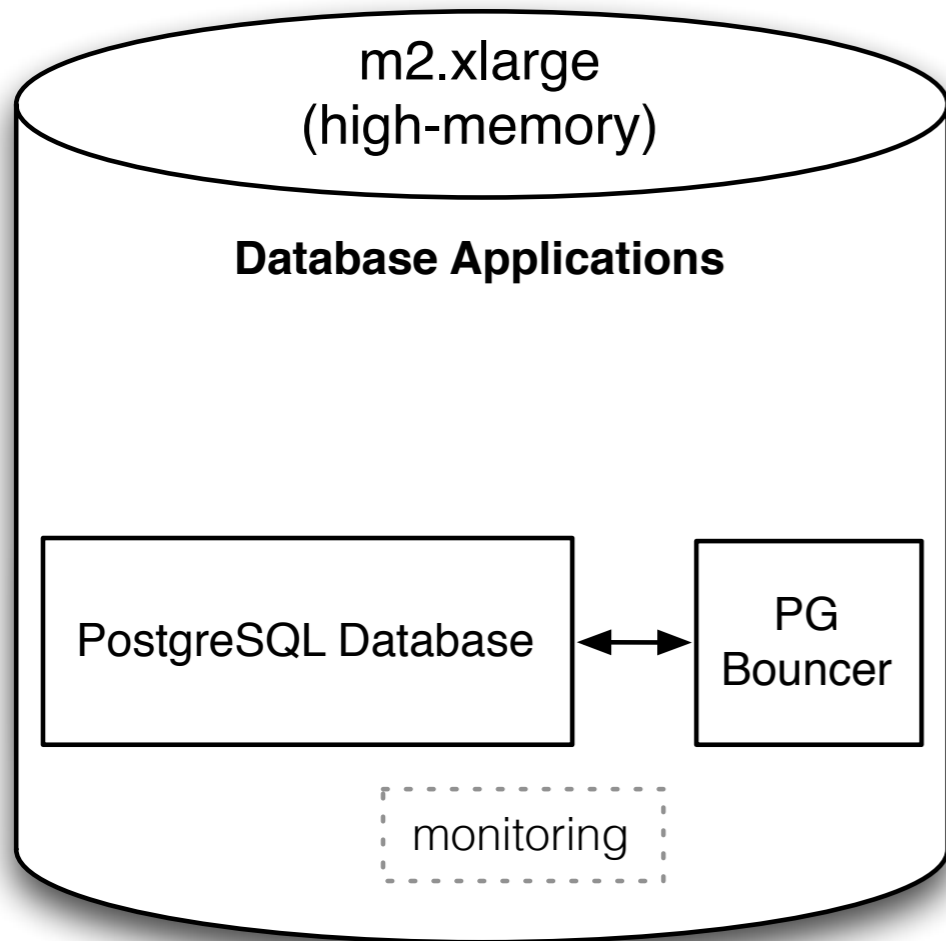
- Ubuntu 10.10 (Maverick)
- 64-bit
- Kernel Settings
- Spatial Database
- Connection Pooling

```
apt-get install...  
pgbouncer 1.3.3
```





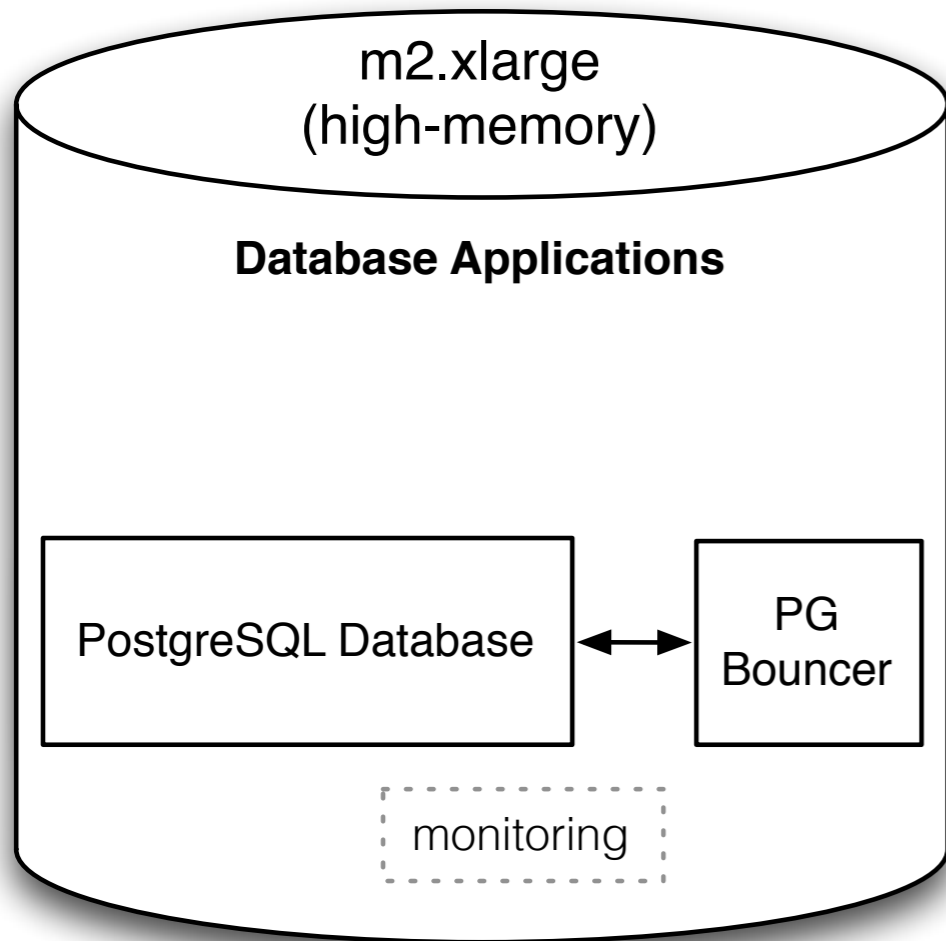
# Custom DB AMI



- Ubuntu 10.10 (Maverick)
- 64-bit
- Kernel Settings
- Spatial Database
- Connection Pooling
- Monitoring



# Custom DB AMI

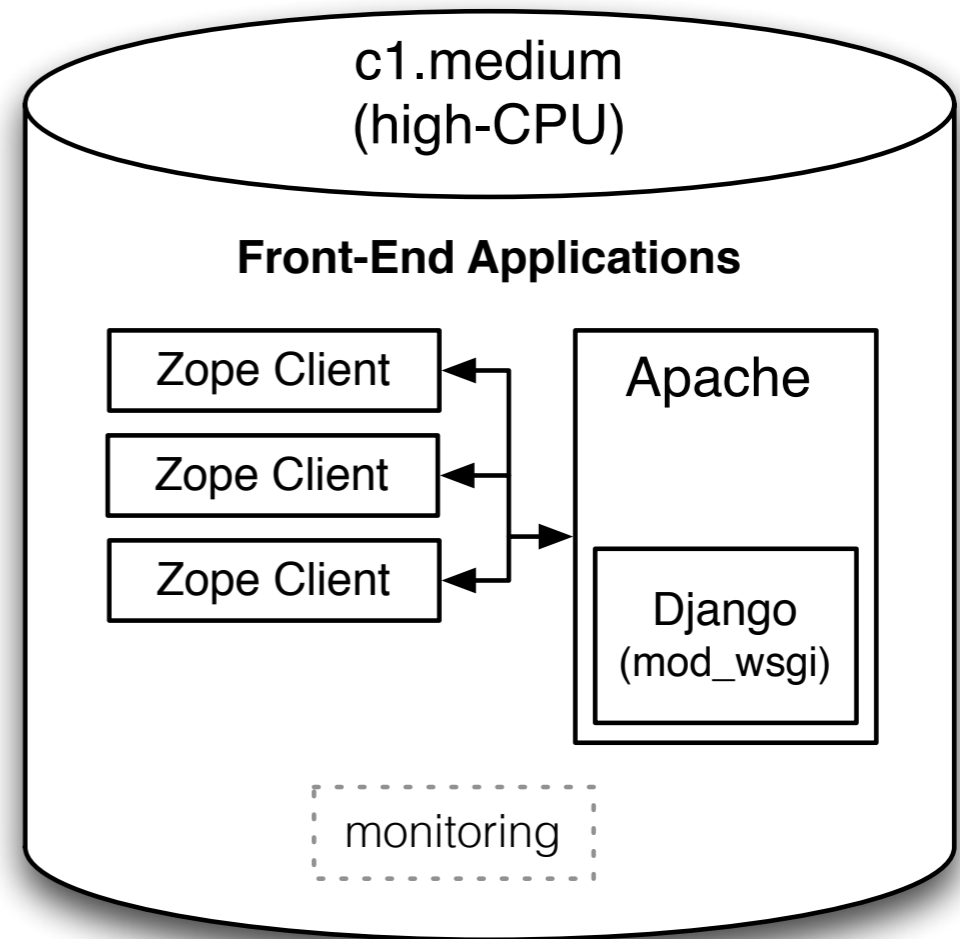


- Ubuntu 10.10 (Maverick)
- 64-bit
- Kernel Settings
- Spatial Database
- Connection Pooling
- Monitoring

```
apt-get install...  
monit  
munin-node
```

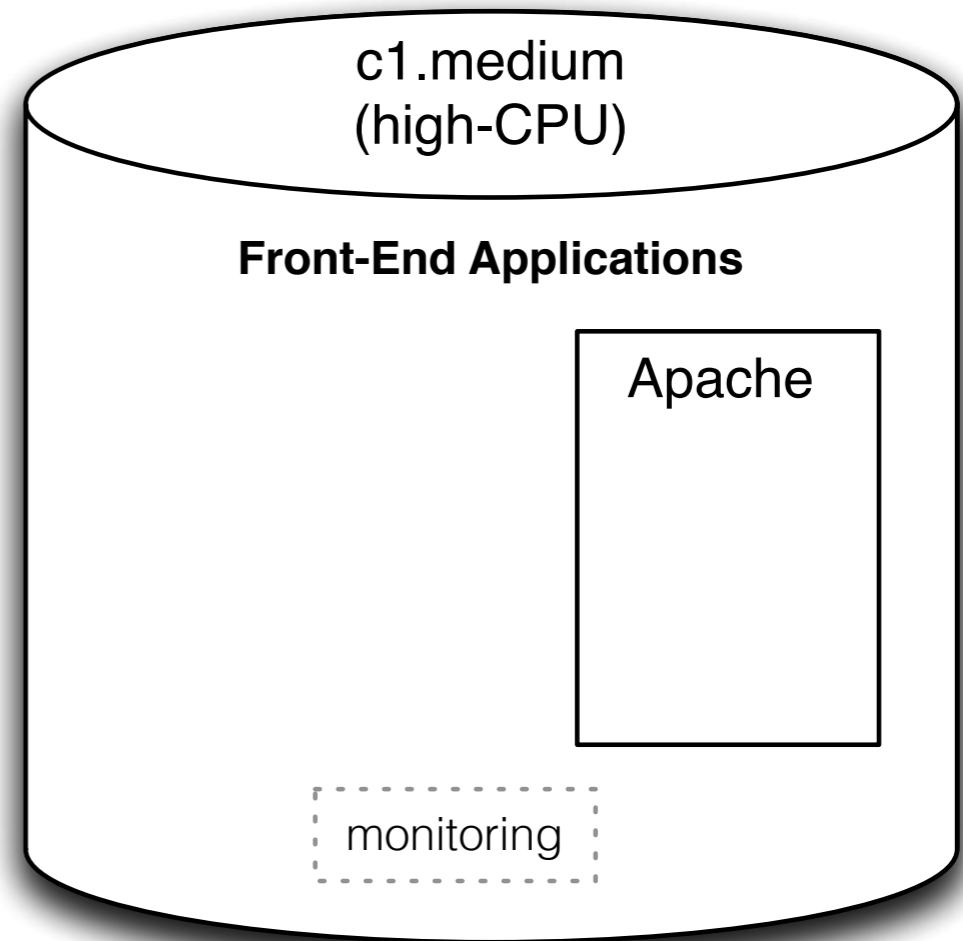


# Custom Web App AMI





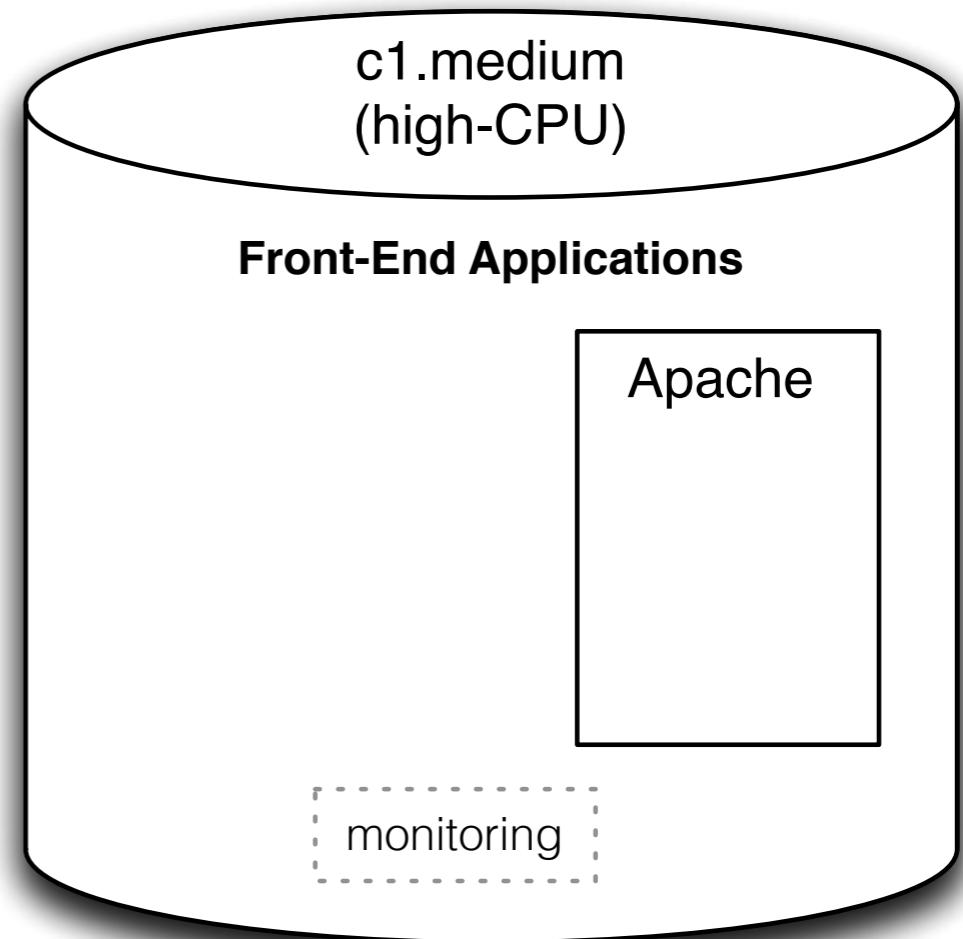
# Custom Web App AMI





# Custom Web App AMI

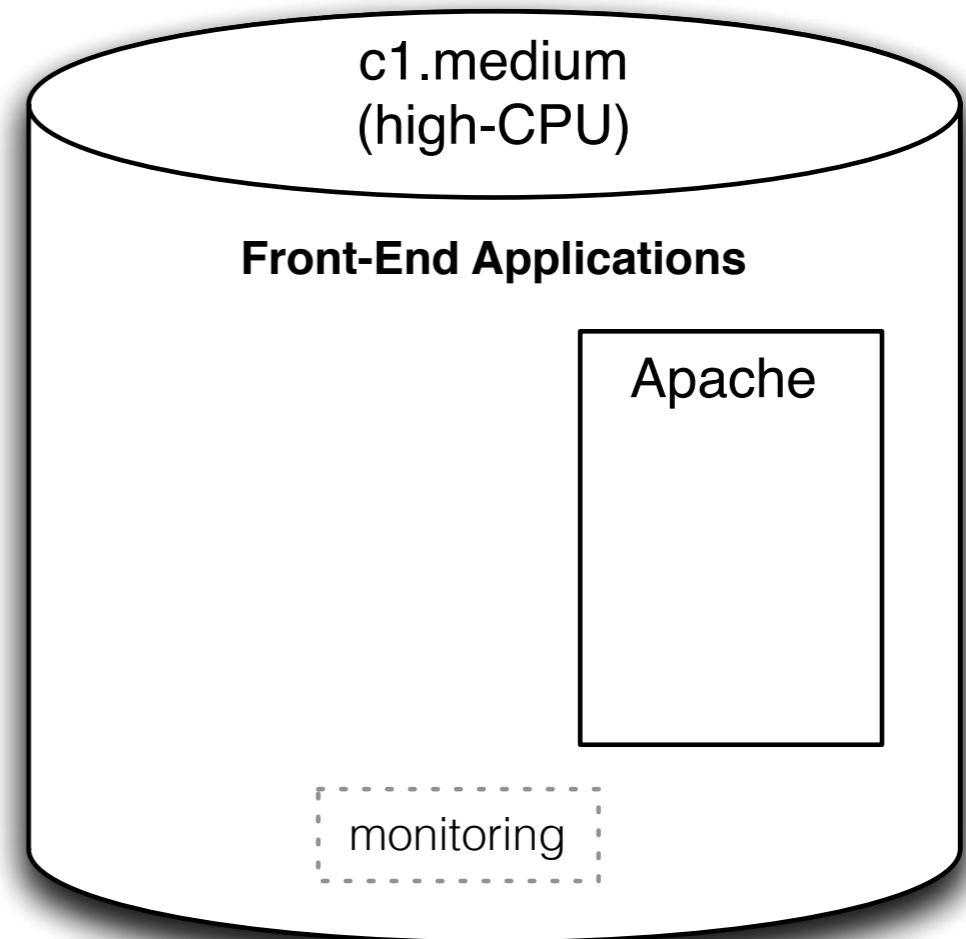
- Ubuntu 10.10 (Maverick)





# Custom Web App AMI

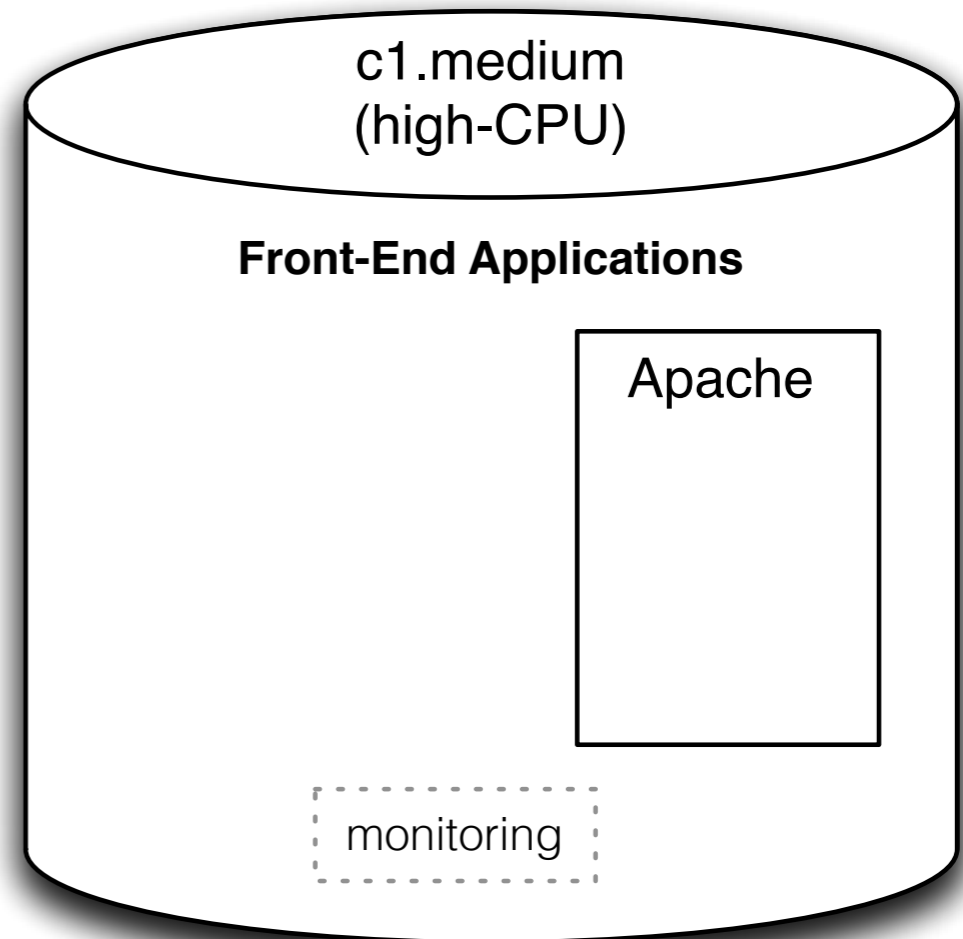
- Ubuntu 10.10 (Maverick)
- 32-bit





# Custom Web App AMI

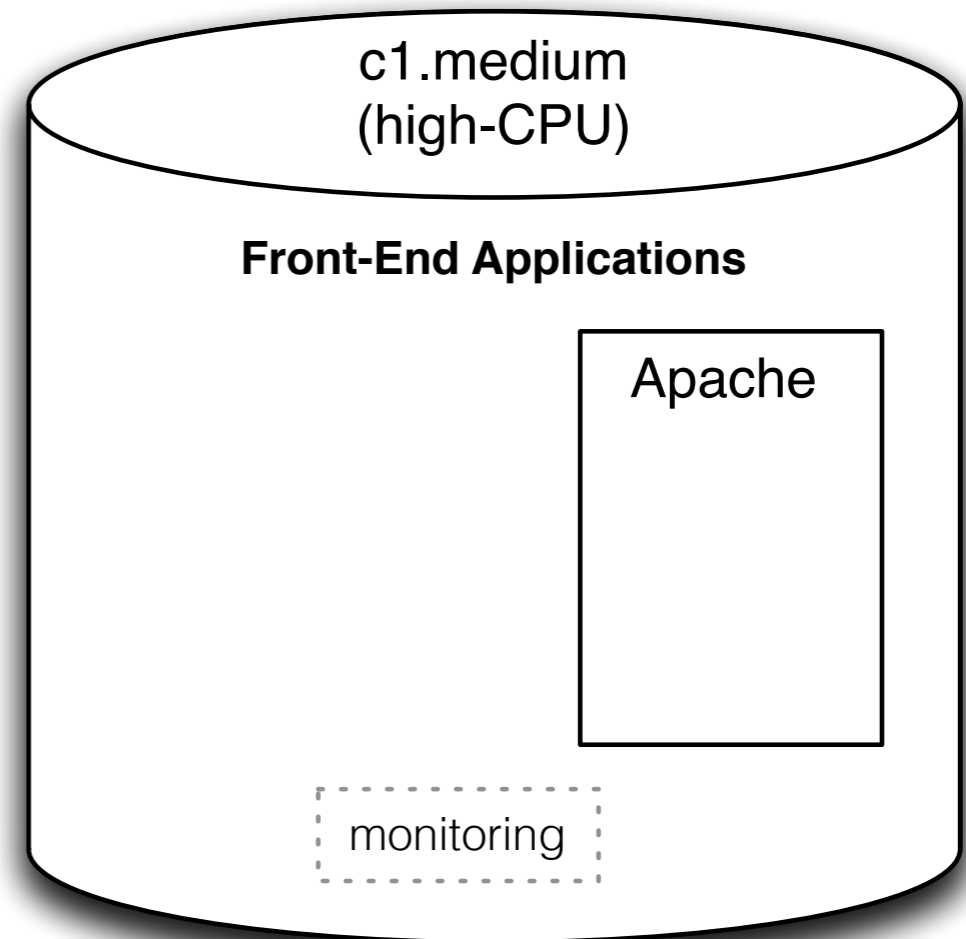
- Ubuntu 10.10 (Maverick)
- 32-bit
- Apache2





# Custom Web App AMI

- Ubuntu 10.10 (Maverick)
- 32-bit
- Apache2
- mod\_wsgi

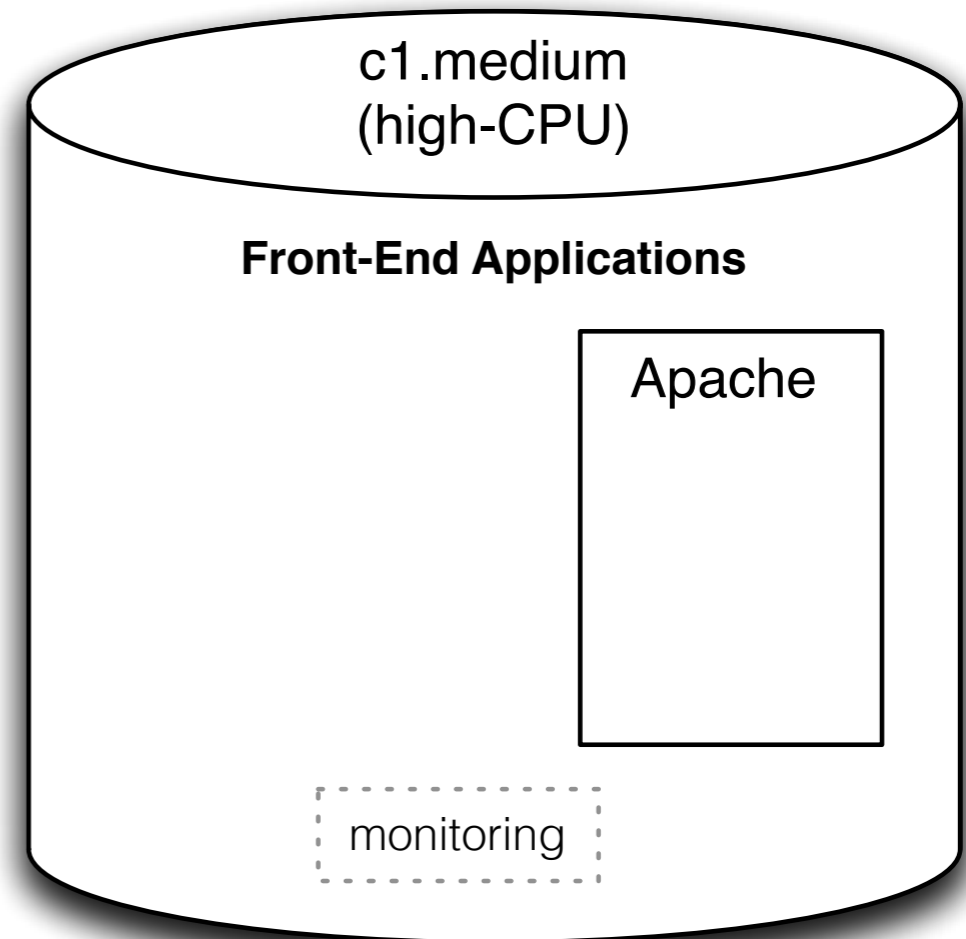






# Custom Web App AMI

- Ubuntu 10.10 (Maverick)
- 32-bit
- Apache2
- mod\_wsgi
- Monitoring





**We'll make these public**



# We'll make these public

– once I've polished them up a bit :)



# Additional Software



# Additional Software

- Back-End



# Additional Software

- Back-End
  - ZODB



# Additional Software

- Back-End
  - ZODB
  - ZEO Server



# Additional Software

- Back-End
  - ZODB
  - ZEO Server
- Front-End





# Additional Software

- Back-End
  - ZODB
  - ZEO Server
- Front-End
  - Django



# Additional Software

- Back-End
  - ZODB
  - ZEO Server
- Front-End
  - Django
  - Plone



# Additional Software

- Back-End
  - ZODB
  - ZEO Server
- Front-End
  - Django
  - Plone
    - Add-Ons



# Additional Software

- Back-End
  - ZODB
  - ZEO Server
- Front-End
  - Django
  - Plone
    - Add-Ons
  - Zope Clients

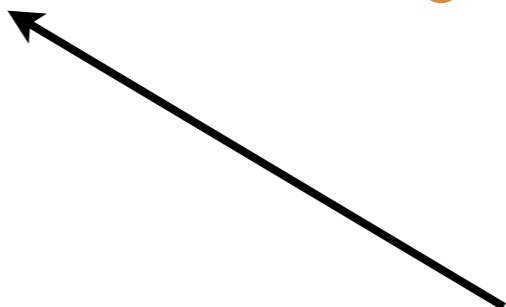


# And Configuration

- Back-End
  - ZODB
  - ZEO Server
- Front-End
  - Django
  - Plone
    - Add-Ons
  - Zope Clients



# And Configuration

- Back-End
    - ZODB
    - ZEO Server
  - Front-End
    - Django
    - Plone
      - Add-Ons
    - Zope Clients
- 
- A black arrow originates from the "Zope Clients" item in the Front-End list and points towards the "ZEO Server" item in the Back-End list, indicating a dependency or relationship between the two.



# And Configuration

- Back-End

- ZODB

- ZEO Server

PostgreSQL

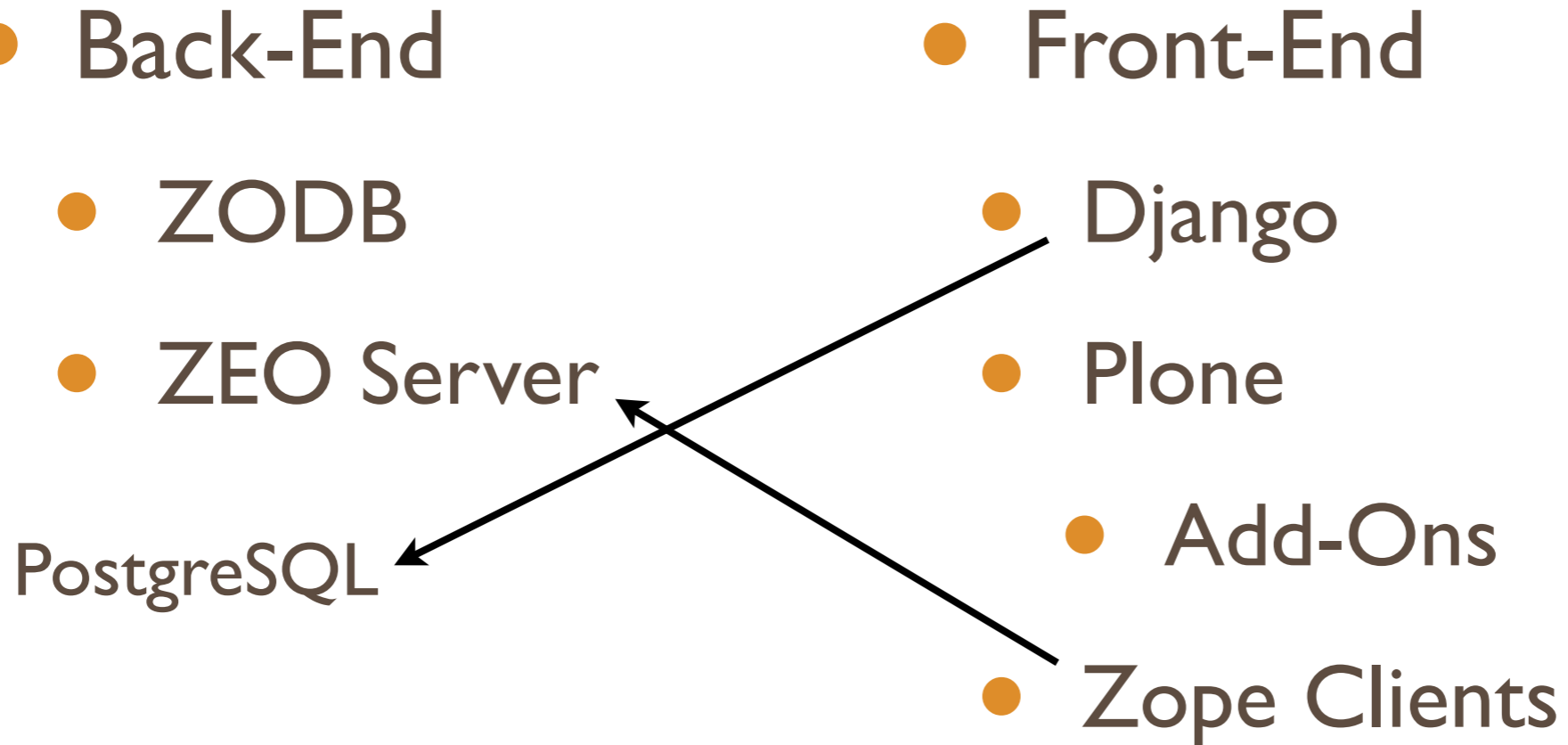
- Front-End

- Django

- Plone

- Add-Ons

- Zope Clients





# And Configuration

- Back-End

- ZODB

- ZEO Server

PostgreSQL

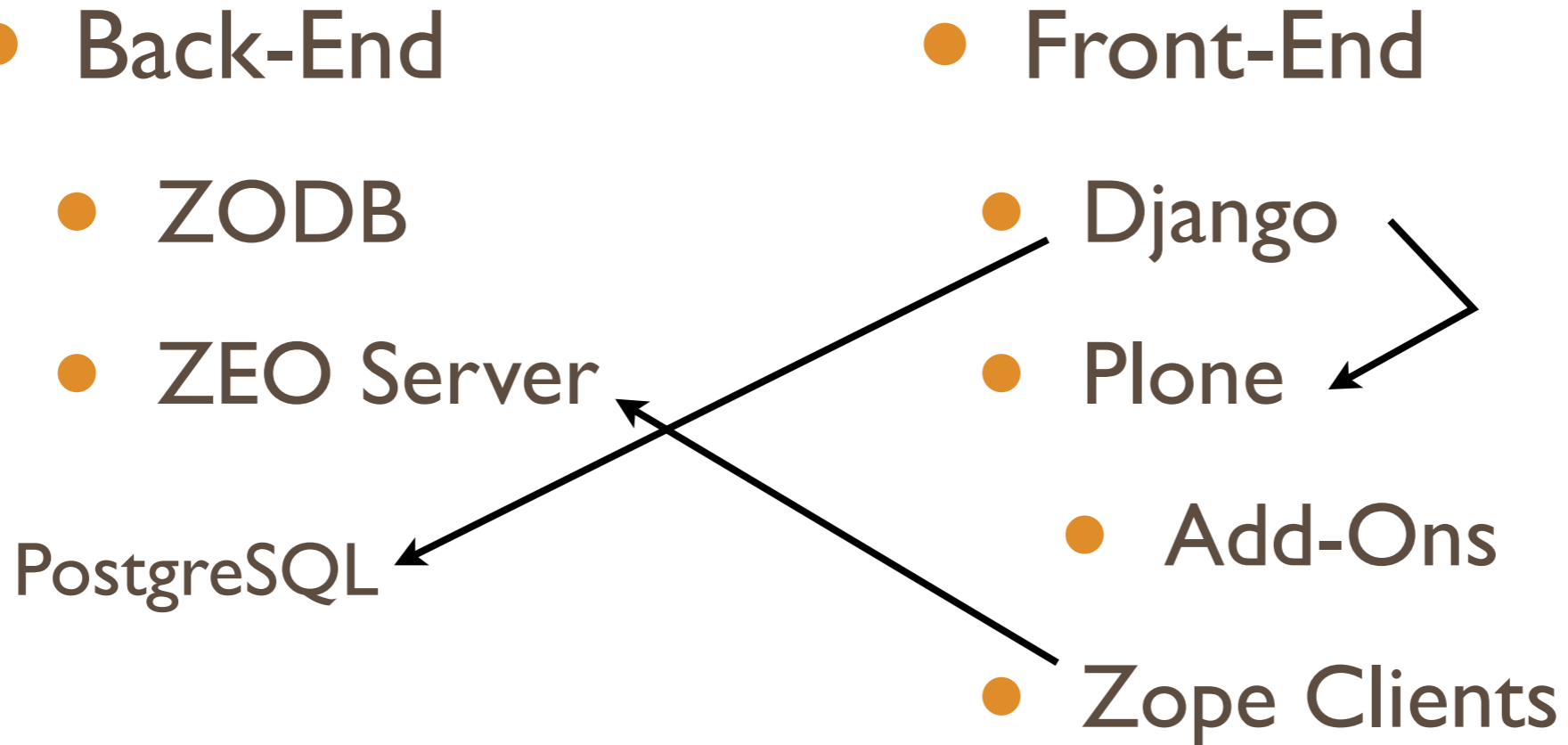
- Front-End

- Django

- Plone

- Add-Ons

- Zope Clients







# And Configuration

- Back-End

- ZODB

- ZEO Server

PostgreSQL

- Front-End

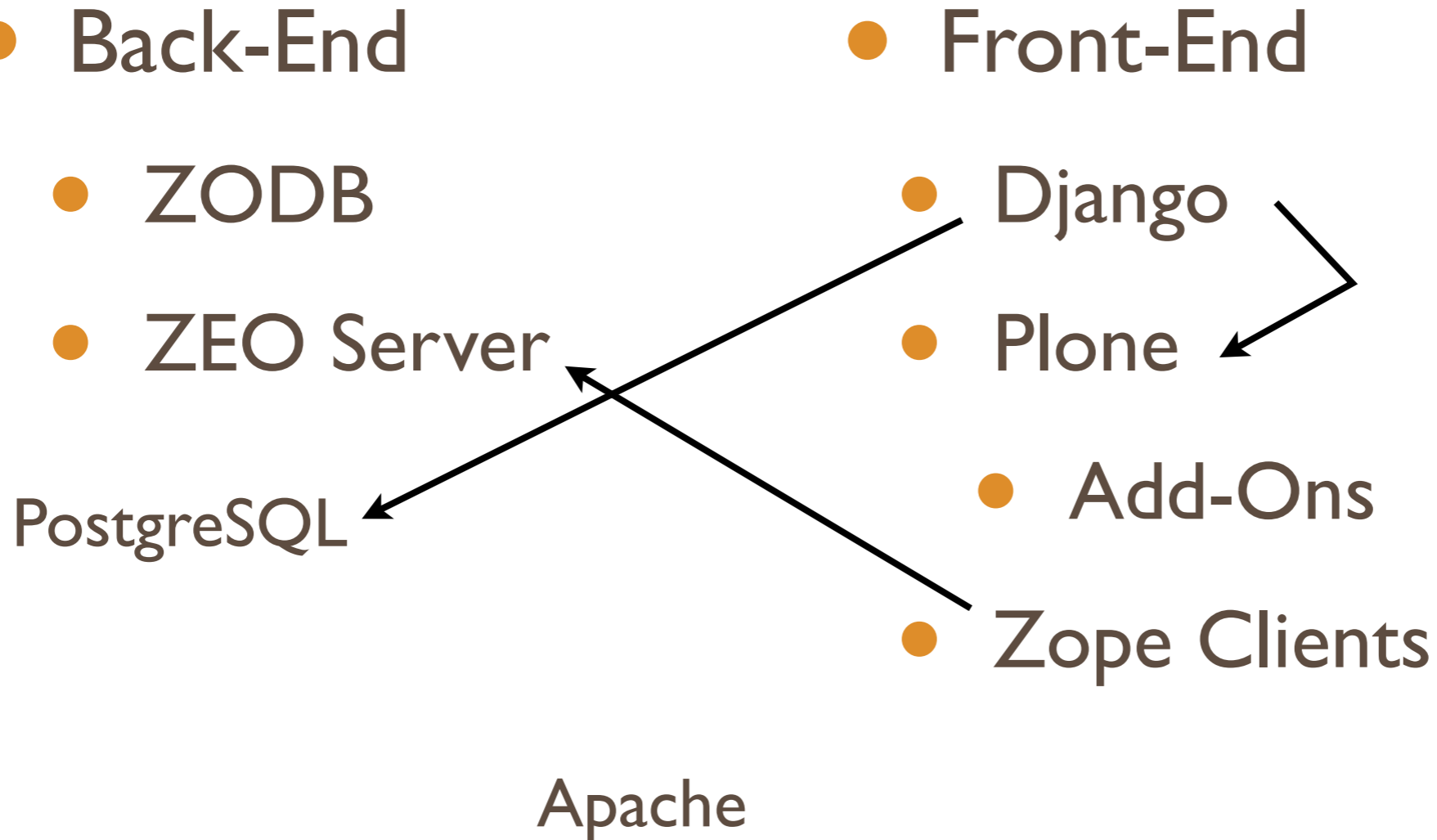
- Django

- Plone

- Add-Ons

- Zope Clients

Apache





# And Configuration

- Back-End

- ZODB

- ZEO Server

PostgreSQL

- Front-End

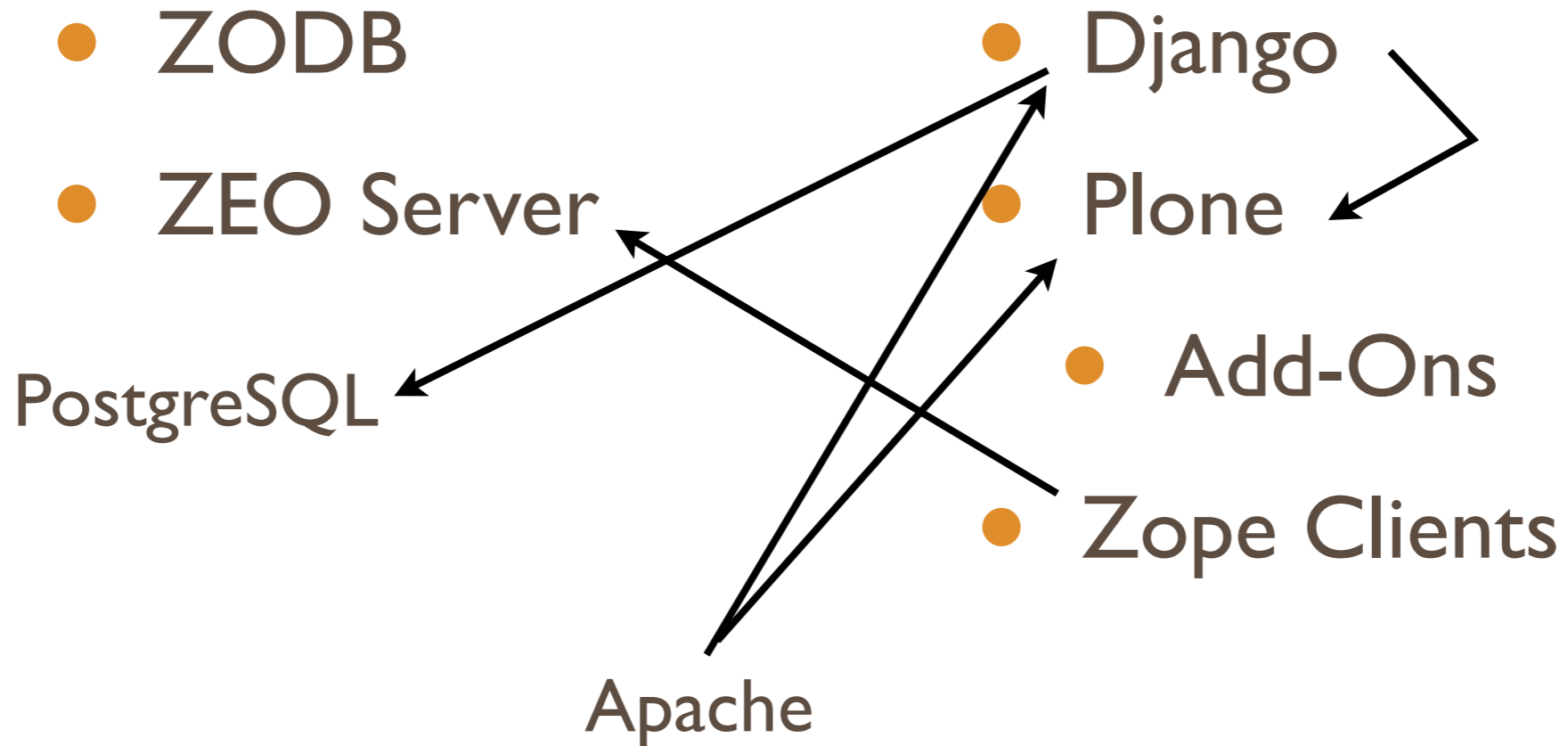
- Django

- Plone

- Add-Ons

- Zope Clients

Apache





# And Configuration

- Back-End

- ZODB

- ZEO Server

PostgreSQL

Monit

- Front-End

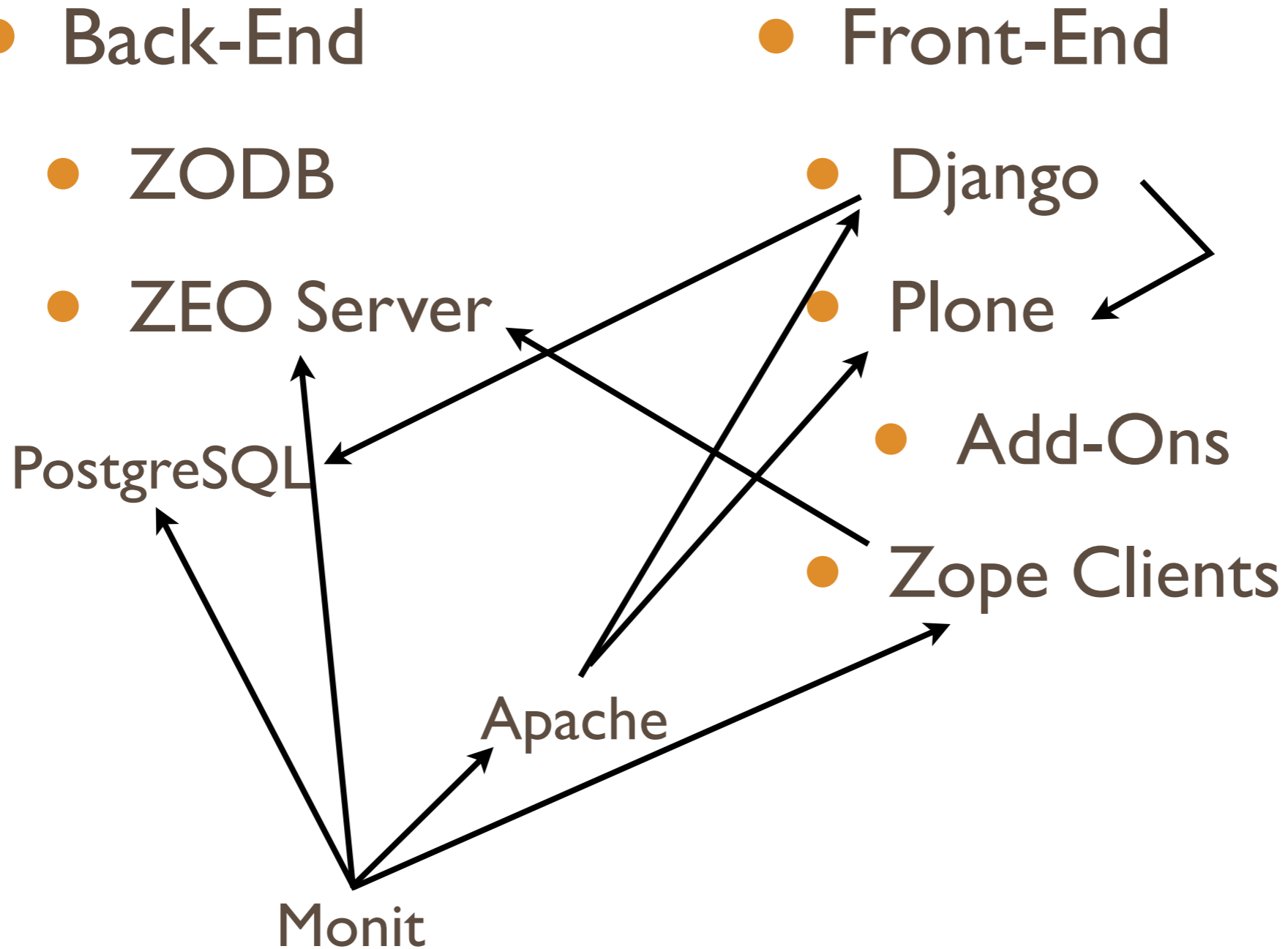
- Django

- Plone

- Add-Ons

- Zope Clients

Apache





# And Configuration

- Back-End

- ZODB

- ZEO Server

PostgreSQL

- Front-End

- Django

Plone

- Add-Ons

- Zope Clients

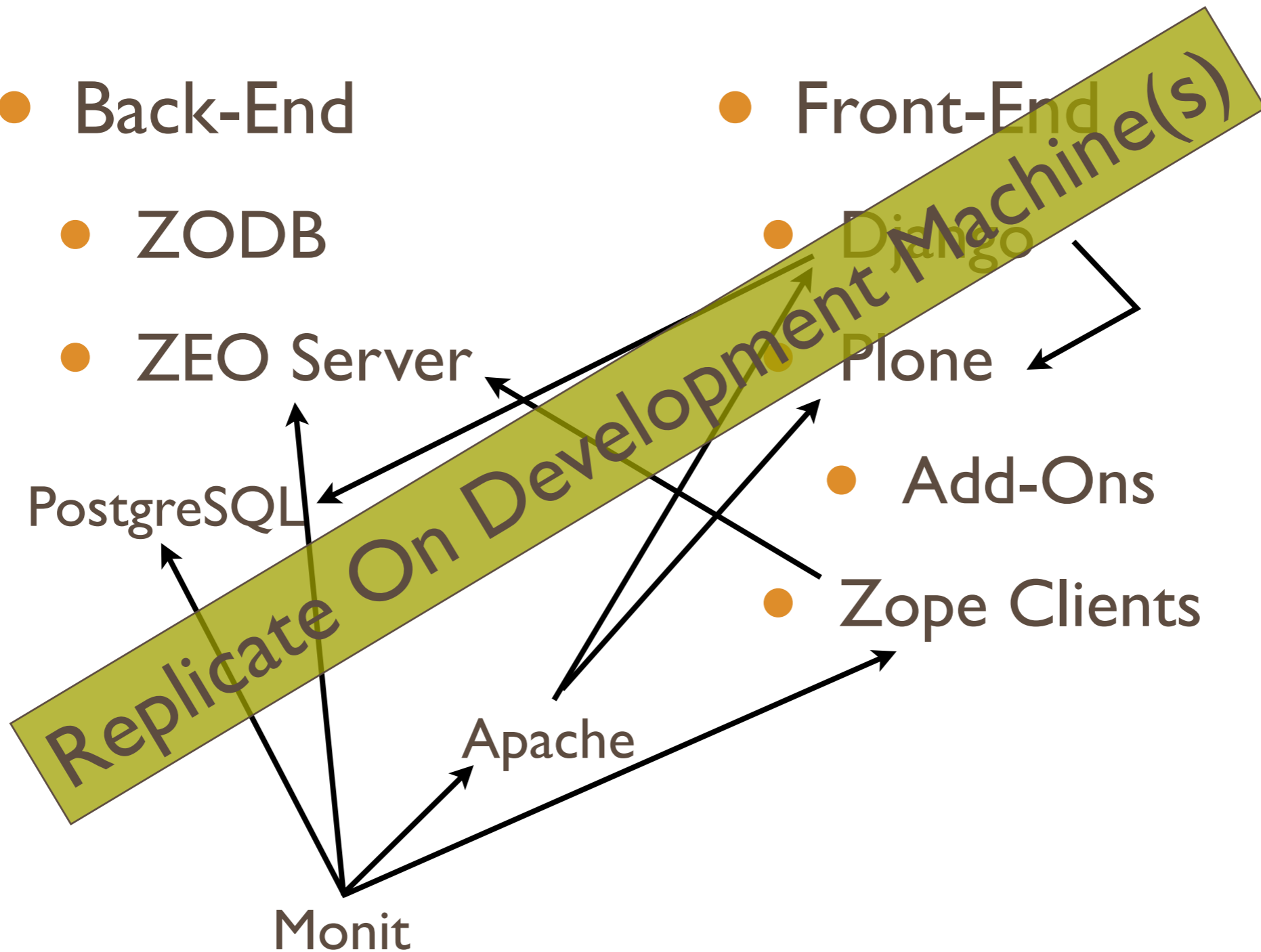






Photo by EvelynGiggles via Flickr using a CC-BY license.





**This is a job for Superman!**



**This is a job for Buildout!**



Wait – what?





**zc.buildout**

[www.buildout.org](http://www.buildout.org)



developed for





used extensively by



Plone<sup>®</sup>



also gaining traction  
with

**django**

  
**GROK**

The GROK logo features a stylized grey and yellow object resembling a rock or a piece of wood with several yellow dots on its surface, positioned above the word "GROK" in a bold, black, sans-serif font.

 Pyramid™

The Pyramid logo consists of a yellow pyramid shape inside a dark grey square, followed by the word "Pyramid" in a black, sans-serif font with a trademark symbol.



not limited to python



# dependencies



# dependencies

```
$ apt-get install python-setuptools
```



# dependencies

```
$ apt-get install python-setuptools
```

```
$ easy-install distribute
```





# dependencies

```
$ apt-get install python-setuptools
```

```
$ easy-install distribute
```

```
$ apt-get install python-dev
```



# The Basics



# The Basics

\$ ls basic-buildout

bootstrap.py

buildout.cfg



# The Basics



# The Basics

```
$ more buildout.cfg
```

```
[buildout]
```

```
parts = foo
```

```
[foo]
```

```
recipe = collective.recipe.cmd
```

```
cmd =
```

```
    echo foo
```



# Build It

```
Terminal — bash — 80x24
fiver:basic-buildout cewing$ python bootstrap.py
install_dir /var/folders/BK/BK9HvLYeFsaqGWLX1x8Mxk+++TI/-Tmp-/tmpm1quqi
Creating directory '/Users/cewing/basic-buildout/bin'.
Creating directory '/Users/cewing/basic-buildout/parts'.
Creating directory '/Users/cewing/basic-buildout/develop-eggs'.
Generated script '/Users/cewing/basic-buildout/bin/buildout'.
fiver:basic-buildout cewing$
```



# Run It

```
Terminal — bash — 80x24
fiver:basic-buildout cewing$ bin/buildout
/Users/cewing/basic-buildout/parts/buildout/site.py:262: UserWarning: Module pkg_resources was already imported from /System/Library/Frameworks/Python.framework/Versions/2.6/Extras/lib/python/pkg_resources.pyc, but /Users/cewing/.buildout/eggs/distribute-0.6.14-py2.6.egg is being added to sys.path
  import pkg_resources
/Users/cewing/basic-buildout/parts/buildout/site.py:262: UserWarning: Module site was already imported from /Users/cewing/basic-buildout/parts/buildout/site.pyc, but /Users/cewing/.buildout/eggs/distribute-0.6.14-py2.6.egg is being added to sys.path
  import pkg_resources
Installing foo.
foo
fiver:basic-buildout cewing$
```





Photo by Rich Moore via Flickr using a CC-BY license.

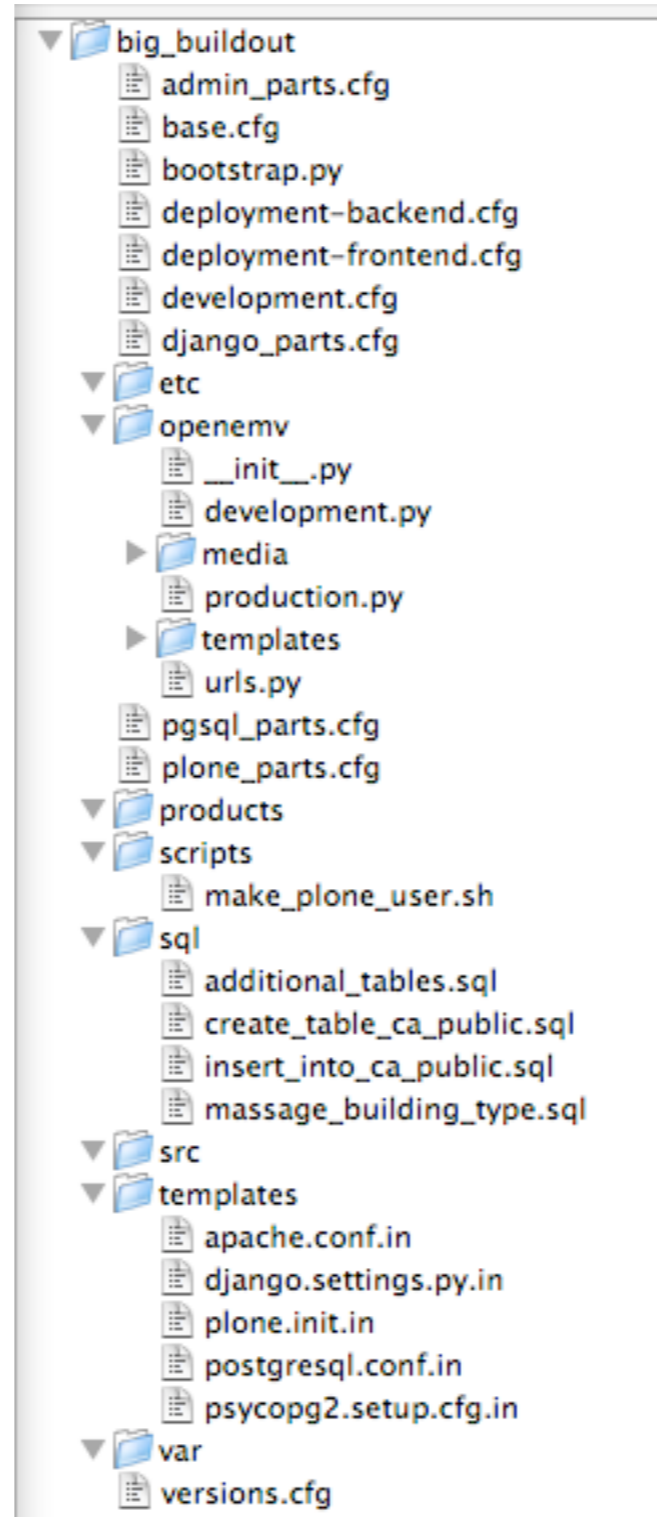




let's scale this up



# lots of cfg, and more





base.cfg

```
1 [buildout]
2 extensions =
3     mr.developer
4     buildout.threatlevel
5     buildout.dumppickedversions
6
7 sources = sources
8 auto-checkout =
9     sdm.openemv_policy
10    openemv_maps
11    openemv_theme
12
13 # Change the number here to change the version of Plone being used
14 extends =
15     http://dist.plone.org/release/4.0.3/versions.cfg
16
17 versions = versions
18
19 # Add additional egg download sources here. dist.plone.org contains archives
20 # of Plone packages.
21 find-links =
22     http://dist.plone.org/release/4.0.3
23     http://dist.plone.org/thirdparty
24     http://dist.plone.org
25     http://download.zope.org/ppix/
26     http://download.zope.org/distribution/
27     http://effbot.org/downloads
28     http://dist.repoze.org/
29
30 # Add additional eggs here
31 plone-eggs =
32     Plone
33     PIL==1.1.6
34     Products.Scrawl
35     Products.ContentWellPortlets
36     Products.Collage
37     collective.collage.portlets
38     sdm.openemv_policy
39 # we will not need to use the hotfix once we've upgraded past 4.0.3,
40 # but we need it now
41     Products.PloneHotfix20110720
42
43 plone-dev-eggs =
44     plone.reload
45     teamrubber.theoracle == 0.0.5
```



pg\_parts.cfg

```
2 #####
3 ### Parts for building spatial postgresql database (dev only)
4 #####
5
6 [postgresql]
7 recipe = zc.recipe.cmmi
8 url =
9 http://ftp9.us.postgresql.org/pub/mirrors/postgresql/source/v9.0.3/postgresql-
10 9.0.3.tar.gz
11 extra-options =
12     --with-readline
13     --enable-thread-safety
14
15 [proj]
16 recipe = hexagonit.recipe.cmmi
17 url = http://download.osgeo.org/proj/proj-4.7.0.tar.gz
18
19 [geos]
20 recipe = hexagonit.recipe.cmmi
21 url = http://download.osgeo.org/geos/geos-3.2.2.tar.bz2
22
23 [gdal]
24 recipe = hexagonit.recipe.cmmi
25 url = http://download.osgeo.org/gdal/gdal-1.7.2.tar.gz
26 configure-options =
27     --with-python
28     --with-geos=${geos:location}/bin/geos-config
29
30 [postgis]
31 recipe = hexagonit.recipe.cmmi
32 url = http://postgis.refrations.net/download/postgis-1.5.2.tar.gz
33 configure-options =
34     --with-pgsql=${postgresql:location}
35     --with-pgconfig=${postgresql:location}/bin/pg_config
36     --with-geos=${geos:location}
37     --with-geosconfig=${geos:location}/bin/geos-config
38     --with-proj=${proj:location}
39     --with-projdir=${proj:location}
40
41 [init-pgsql]
42 recipe = collective.recipe.cmd
```



# A Closer Look

```
39 [init-pgsql]
40 recipe = collective.recipe.cmd
41 on_install = true
42 on_update = true
43 cmds =
44     mkdir -c ${buildout:directory}/var/data
45     ${postgresql:location}/bin/initdb -D ${buildout:directory}/var/data -E
    . UNICODE
46     ${postgresql:location}/bin/pg_ctl -D ${buildout:directory}/var/data start
47     sleep 5
48     ${postgresql:location}/bin/createdb -E UTF8 template_postgis
49     ${postgresql:location}/bin/createlang -d template_postgis plpgsql
50     ${postgresql:location}/bin/psql -d template_postgis -f
    . ${postgresql:location}/share/contrib/postgis-1.5/postgis.sql
51     ${postgresql:location}/bin/psql -d template_postgis -f
    . ${postgresql:location}/share/contrib/postgis-1.5/spatial_ref_sys.sql
52     ${postgresql:location}/bin/pg_ctl -D ${buildout:directory}/var/data stop
53     sleep 5
```



Tie it all together...





dev.cfg

```
1 [buildout]
2 extends =
3     base.cfg
4     pgsql_parts.cfg
5     django_parts.cfg
6     plone_parts.cfg
7
8 parts =
9     postgresql
10    proj
11    geos
12    gdal
13    postgis
14    init-pgsql
15    pgsql-symlinks
16    psycopg2-package
17    psycopg2-setup
18    psycopg2-install
19    django
20    theme-symlinks
21    django-settings
22    drop-openemv
23    setup-openemv
24    zopepy
25    zeoserver
26    instance
27    test
28    omelette
29    zopeskel
30
31    zeoserver-address = 127.0.0.1
32    zeoserver-port = 8901
33    ip-address = 127.0.0.1
34    instance-port = 8080
35    debug-mode = on
36    verbose-security = on
37    deprecation-warnings = on
38    django-db-user =
39    django-db-passwd =
40    django-db-host =
41    django-db-port =
```



or, for deployment...





# backend.cfg

```
1 [buildout]
2 extends =
3     base.cfg
4     plone_parts.cfg
5     admin_parts.cfg
6
7 parts =
8     ploneuser-setup
9     zeoserver
10    zeo-init
11    backup
12    monit-setup
13    monitconf
14    monit-system
15    monit-zeoserver
16    monit-postgresql
17    monit-pgemailer
18    postgresqlconf
19
20    zeoserver-address =
21    zeoserver-port = 8901
22    monit-port = 8100
23    monit-address = 127.0.0.1
24    effective-user = plone
25    host-readable-name = OpenEMV Back-End
26
27
28 [sources]
29 sdm.openemv_policy = svn
30    https://sounddatamanagement.svn.beanstalkapp.com/openemv/sdm.openemv_policy/trunk/
31 openemv_maps = svn
32    http://sounddatamanagement.svn.beanstalkapp.com/openemv/openemv_maps/trunk
33 openemv_theme = svn
34    http://sounddatamanagement.svn.beanstalkapp.com/openemv/openemv_theme/trunk
35
36 [versions]
37 cns.recipe.symlink = 0.1
38 collective.recipe.backup = 1.7
39 collective.recipe.template = 1.8
```



# Buildout Advantages



# Buildout Advantages

- Easy to write



# Buildout Advantages

- Easy to write
- Lots of recipes, examples



# Buildout Advantages

- Easy to write
- Lots of recipes, examples
- Manage complexity



# Buildout Advantages

- Easy to write
- Lots of recipes, examples
- Manage complexity
  - Duplicate across platforms



# Buildout Advantages

- Easy to write
- Lots of recipes, examples
- Manage complexity
  - Duplicate across platforms
  - Deploy rapidly



# Buildout Advantages

- Easy to write
- Lots of recipes, examples
- Manage complexity
  - Duplicate across platforms
  - Deploy rapidly
- Versioning of software setup and config





not without issues



# not without issues

- Can be network-dependent



# not without issues

- Can be network-dependent
- Interconnected parts mean complexity



# not without issues

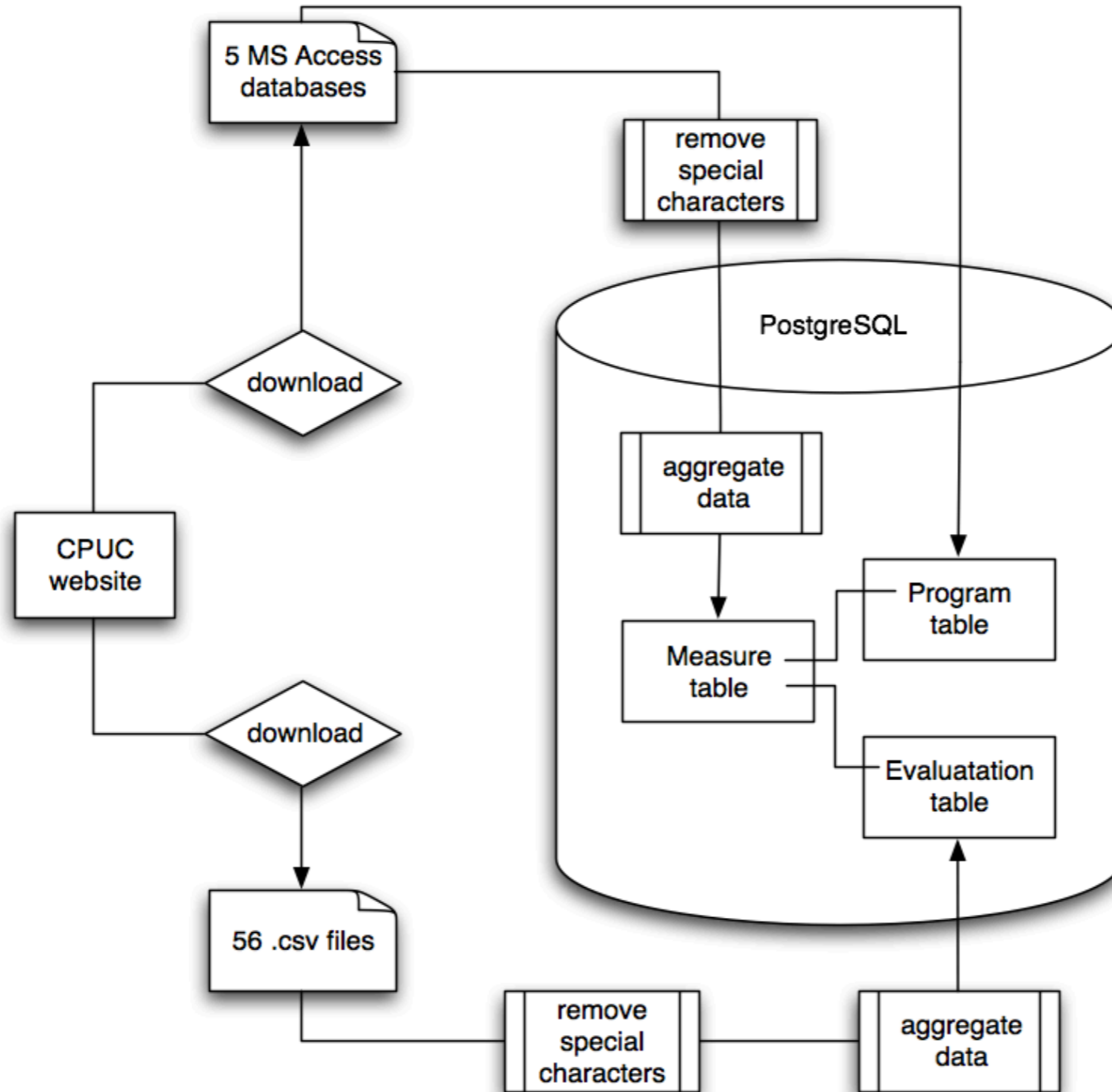
- Can be network-dependent
- Interconnected parts mean complexity
- 'extends' can be tricky to manage



use it in joy

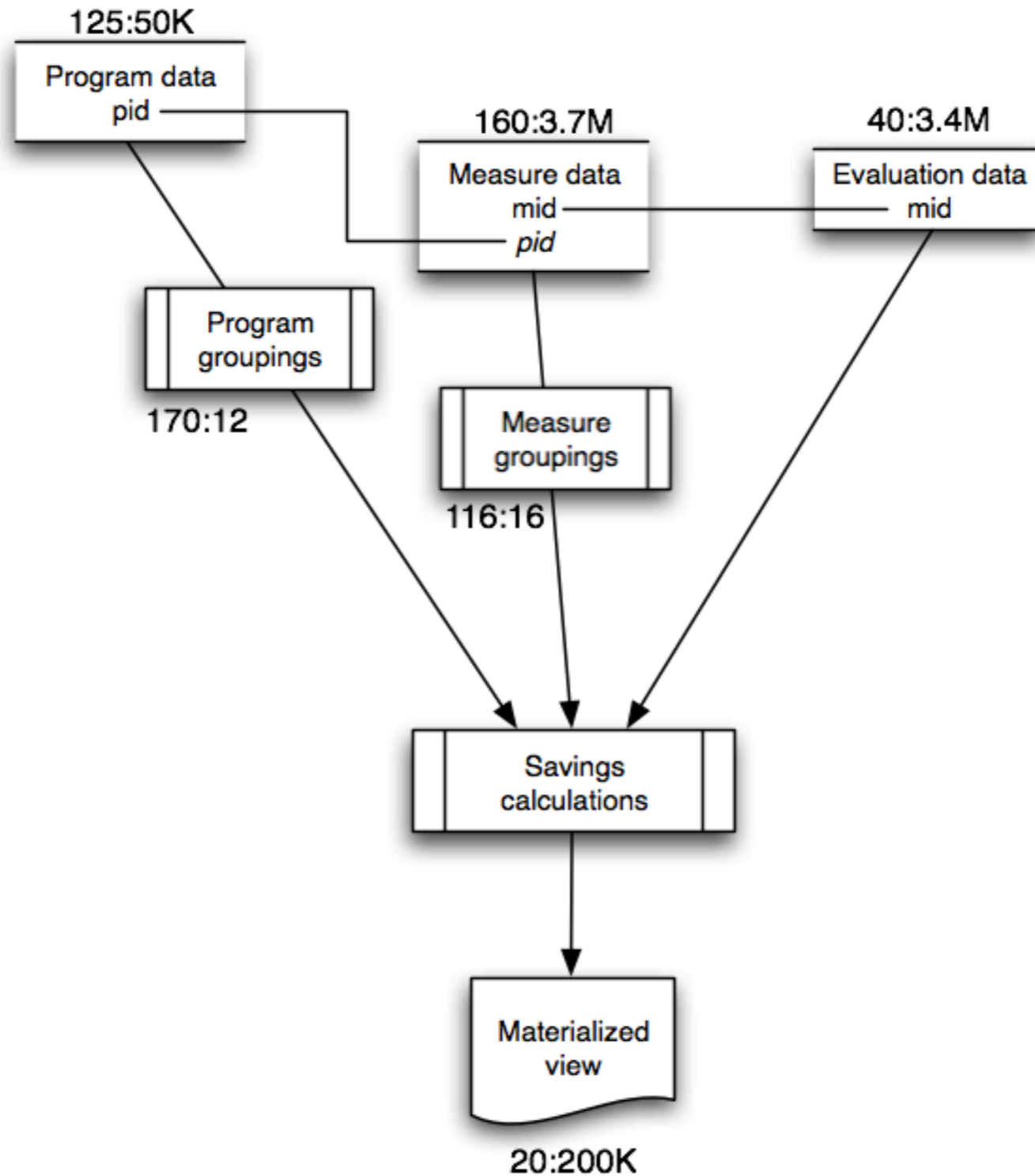


# Data ingest





# Data dimensions











# The Web Application

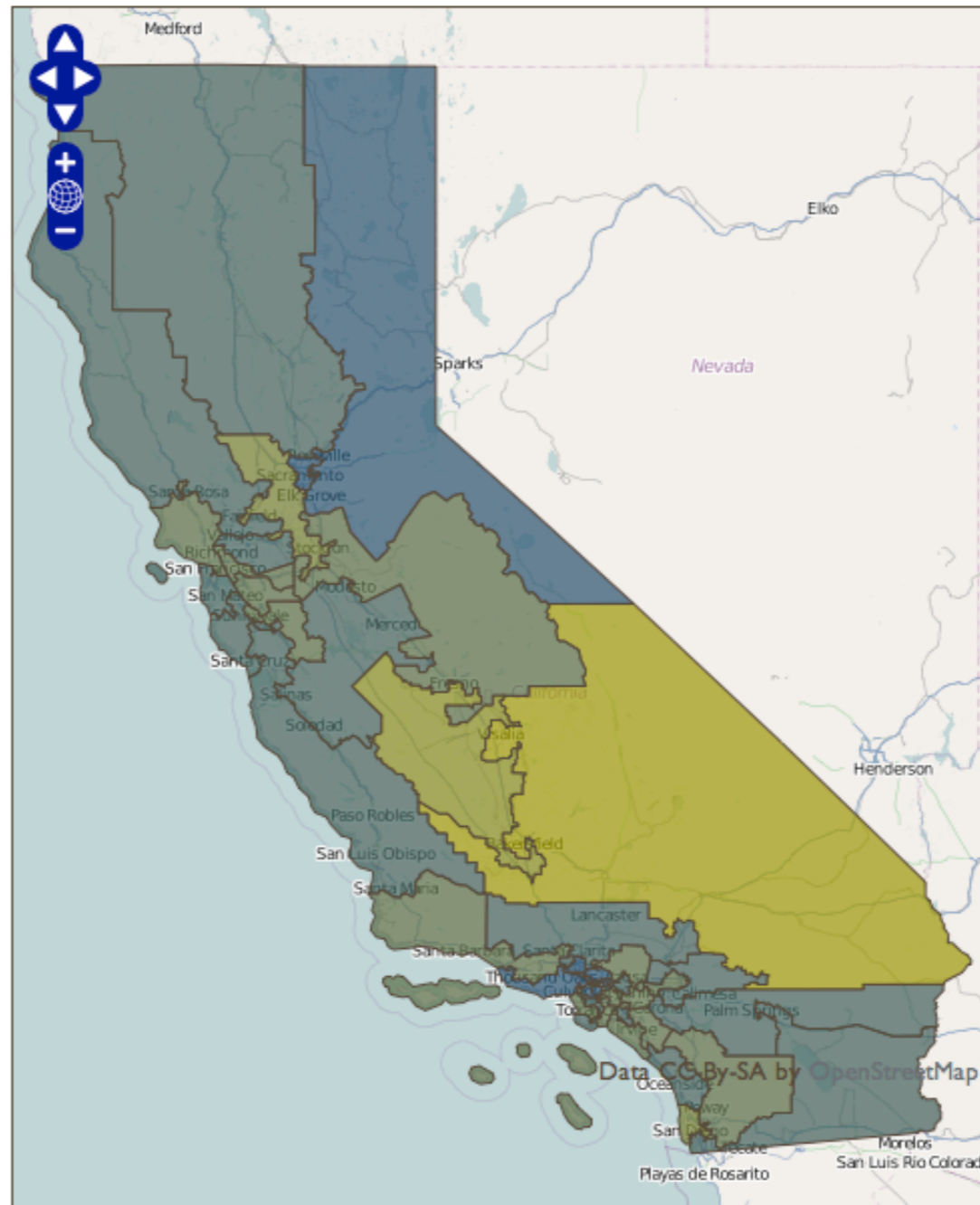
## California Energy Efficiency Program Data

Show me a map of lifecycle evaluated kWh savings by State Senate District

### Legend

-  0 – 613,600,000 kWh
-  613,600,000 – 1,200,000,000 kWh
-  1,200,000,000 – 1,800,000,000 kWh
-  1,800,000,000 – 2,400,000,000 kWh
-  2,400,000,000 – 3,000,000,000 kWh
-  3,000,000,000+ kWh

Download data: [CSV](#), [KML](#)







# 8 Axes of Data



# 8 Axes of Data

- Time Frame (lifecycle or first-year)



# 8 Axes of Data

- Time Frame (lifecycle or first-year)
- Savings Type (but cost too)



# 8 Axes of Data

- Time Frame (lifecycle or first-year)
- Savings Type (but cost too)
- Year



# 8 Axes of Data

- Time Frame (lifecycle or first-year)
- Savings Type (but cost too)
- Year
- Utility



# 8 Axes of Data

- Time Frame (lifecycle or first-year)
- Savings Type (but cost too)
- Year
- Utility
- Program Group



# 8 Axes of Data

- Time Frame (lifecycle or first-year)
- Savings Type (but cost too)
- Year
- Utility
- Program Group
- Measure Group



# 8 Axes of Data

- Time Frame (lifecycle or first-year)
- Savings Type (but cost too)
- Year
- Utility
- Program Group
- Measure Group
- Building Type





# 8 Axes of Data

- Time Frame (lifecycle or first-year)
- Savings Type (but cost too)
- Year
- Utility
- Program Group
- Measure Group
- Building Type
- Geography



only 3 are required



# 8 Axes of Data

- Time Frame (lifecycle or first-year)
- Savings Type (but cost too)
- Year
- Utility
- Program Group
- Measure Group
- Building Type
- Geography



Show me the [timeframe] [savings type] by [geography]



Choices drawn from the distinct values for a specific table column



Show me the [timeframe] [savings type] by [geography]



Choices drawn from the distinct values for a specific table column



Show me the [timeframe] [savings type] by [geography]



Choices drawn from a set of table columns (not the contained values)



Choices drawn from the distinct values for a specific table column

Choices made up but reference table-borne data indirectly

Show me the [timeframe] [savings type] by [geography]

Choices drawn from a set of table columns (not the contained values)



Show me the [timeframe] [savings type] by [geography]





Show me the [timeframe] [savings type] by [geography]

for [year]

for [utility]

for [program group]

for [measure group]

for [building type]



Show me the [timeframe] [savings type] by [geography]  
for [building type]

for [year]  
for [utility]  
for [program group]  
for [measure group]



Show me the [timeframe] [savings type] by [geography]  
for [building type]

- for [year]
- for [utility]
- for [program group]
- for [measure group]



**the sentence interface**



# the sentence interface

Thanks to FogBugz for the inspiration

**FILTER**



**All open cases** in **Open-EMV** assigned to **Cris Ewing** [+](#) [Refine Further](#)

[↶](#) sorted by **Case ID Number (reversed)** sorted by **Priority** [+](#) [Sort](#)



a Django package



# a Django package

A Form



# a Django package

A Form

A Field





# a Django package

A Form

A Field

A Widget



We'll make this public



# We'll make this public

– once I've polished it up a bit :)



How 'bout a quick  
Demo?



questions?  
comments?



jennifer@sound-data.com  
cewing@sound-data.com