

LIVV:

Design &
Architecture

Andrew Bennett, Kate Evans

What is LIVV?

- **Land Ice** This one is easy.
- **Verification** Does the code work correctly?
- **Validation** Does the code do what we want it to?

Good Design

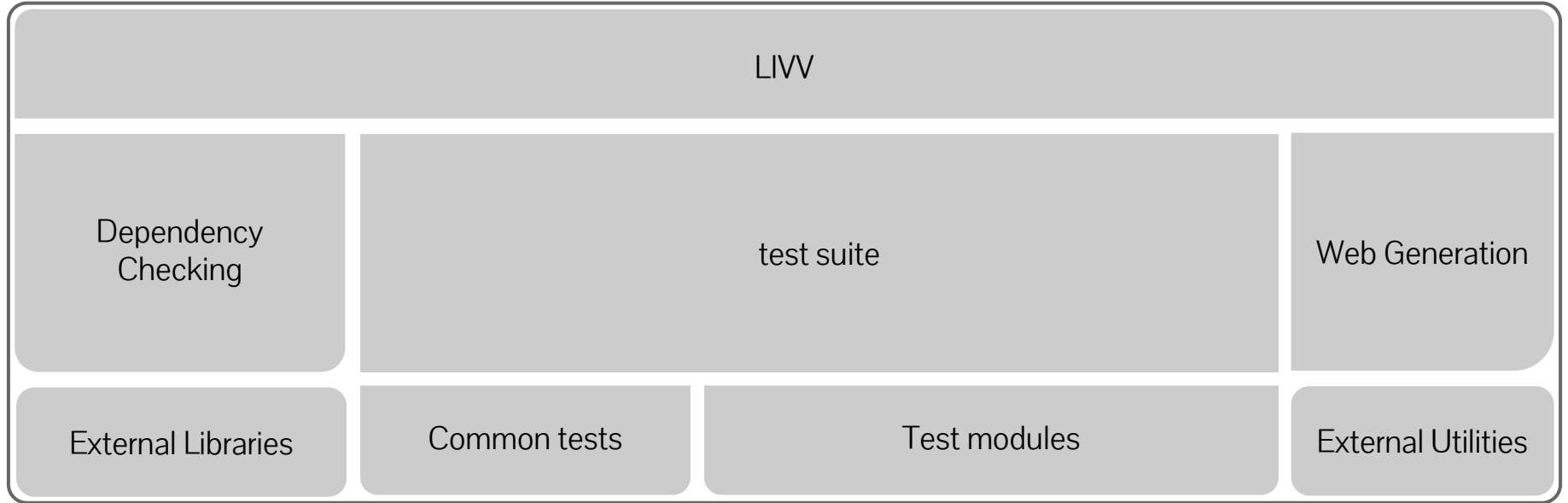
=

Good Tools

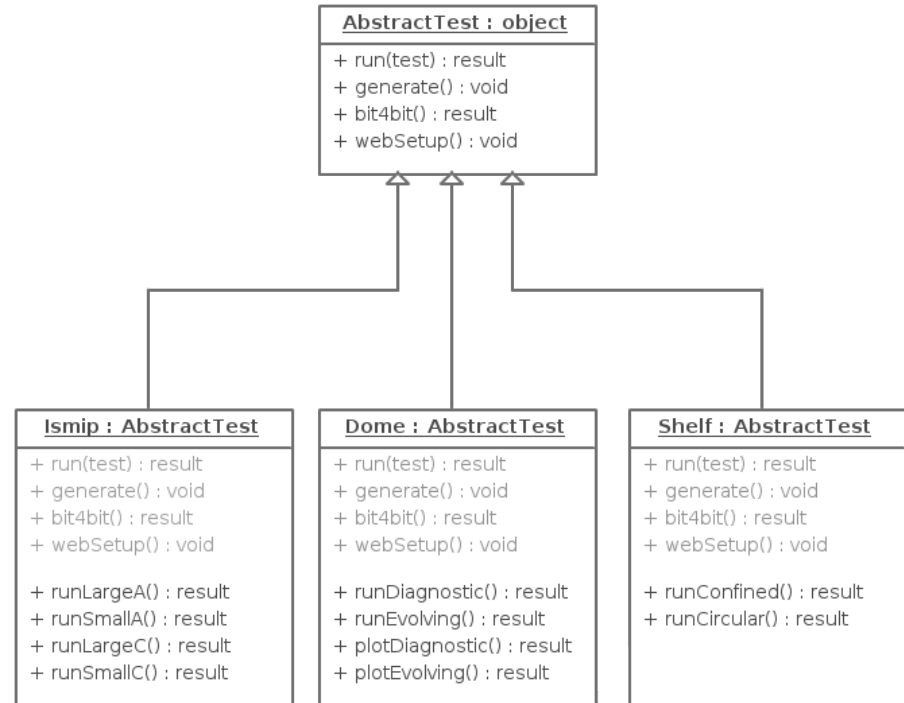
- Internal Dependency Management
- Leverage existing technologies
- Separate input, processing, and output

- Easy to set up and start using productively
- More productivity with less code
- Modular, extensible, and maintainable

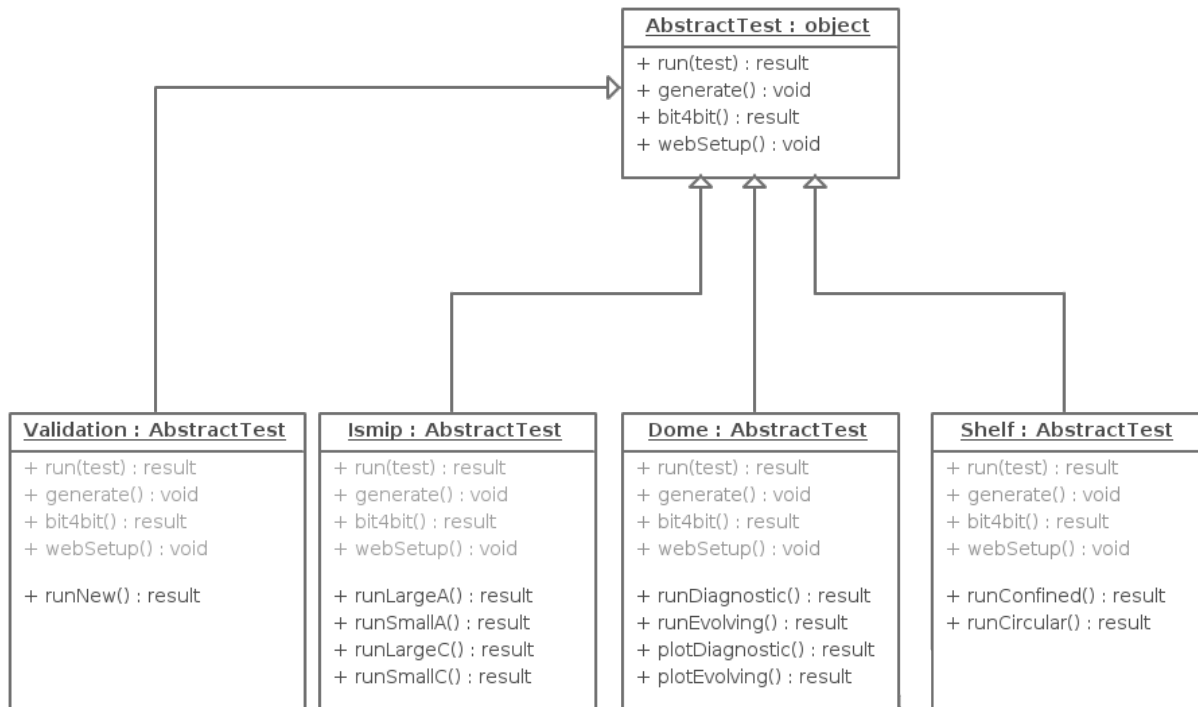
Design



Test Suites



Test Suites



Good Design?

- The code worked fine before.
- Refactoring...? Boring.
- If it's so easy to add on where's the new stuff!?

New Features!

- Internal Dependency Management
- User specified configurations
- Simplified IO via templating

Dependency Management

- Nobody likes figuring out how to install software

Dependency Management

- So far, limited to installing Python libraries
- Soon, installing system dependencies (NCL, NetCDF, etc.)

Configurations

```
Terminal
user@computer ~/livv-dev/LIVV $python livv.py -m myNewConfig -s --dome all --ismip all --
comment "Wow! Look at my configuration file!"
-----
Land ice verification & validation (Livv)
-----

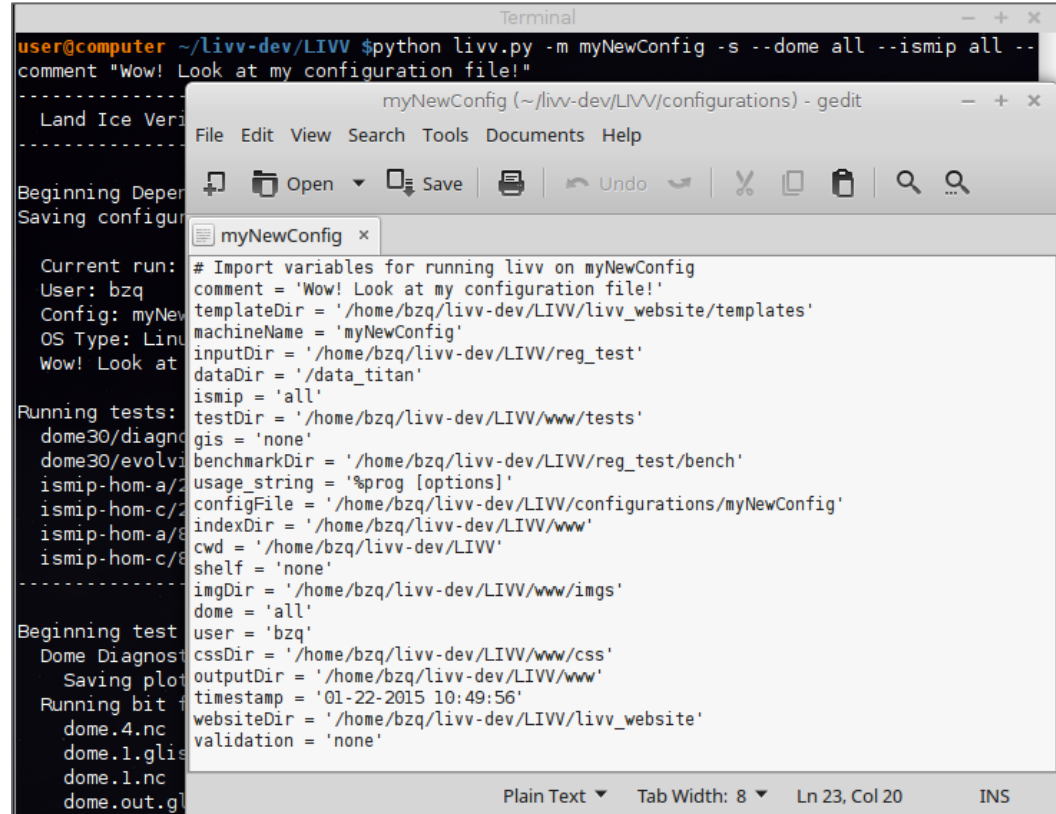
Beginning Dependency Checks..... Okay!
Saving configuration to /home/bzq/livv-dev/LIVV/configurations/myNewConfig

Current run: 01-22-2015 10:49:56
User: bzq
Config: myNewConfig
OS Type: Linux 3.13.0-24-generic
Wow! Look at my configuration file!

Running tests:
dome30/diagnostic
dome30/evolving
ismip-hom-a/20km
ismip-hom-c/20km
ismip-hom-a/80km
ismip-hom-c/80km
-----

Beginning test suite...
Dome Diagnostic test in progress...
Saving plot details to /home/bzq/livv-dev/LIVV/www/imgs/dome as dome30dvel.png
Running bit for bit tests of /dome30/diagnostic...
dome.4.nc FAILURE
dome.1.glissade.nc FAILURE
dome.1.nc FAILURE
dome.out.glissade.nc FAILURE
```

Configurations



```
Terminal
user@computer ~/livv-dev/LIVV $python livv.py -m myNewConfig -s --dome all --ismip all --
comment "Wow! Look at my configuration file!"
-----
Land Ice Ver
-----
Beginning Depen
Saving configur
-----
Current run:
User: bzq
Config: myNew
OS Type: Linu
Wow! Look at
-----
Running tests:
dome30/diagn
dome30/evolv
ismip-hom-a/2
ismip-hom-c/2
ismip-hom-a/8
ismip-hom-c/8
-----
Beginning test
Dome Diagnost
Saving plot
Running bit f
dome.4.nc
dome.1.glis
dome.1.nc
dome.out.g

myNewConfig (~/livv-dev/LIVV/configurations) - gedit
File Edit View Search Tools Documents Help
+ Open Save Undo Cut Copy Paste Find
myNewConfig x
# Import variables for running livv on myNewConfig
comment = 'Wow! Look at my configuration file!'
templateDir = '/home/bzq/livv-dev/LIVV/livv_website/templates'
machineName = 'myNewConfig'
inputDir = '/home/bzq/livv-dev/LIVV/reg_test'
dataDir = '/data_titan'
ismip = 'all'
testDir = '/home/bzq/livv-dev/LIVV/www/tests'
gis = 'none'
benchmarkDir = '/home/bzq/livv-dev/LIVV/reg_test/bench'
usage_string = '%prog [options]'
configFile = '/home/bzq/livv-dev/LIVV/configurations/myNewConfig'
indexDir = '/home/bzq/livv-dev/LIVV/www'
cwd = '/home/bzq/livv-dev/LIVV'
shelf = 'none'
imgDir = '/home/bzq/livv-dev/LIVV/www/imgs'
dome = 'all'
user = 'bzq'
cssDir = '/home/bzq/livv-dev/LIVV/www/css'
outputDir = '/home/bzq/livv-dev/LIVV/www'
timestamp = '01-22-2015 10:49:56'
websiteDir = '/home/bzq/livv-dev/LIVV/livv_website'
validation = 'none'
Plain Text Tab Width: 8 Ln 23, Col 20 INS
```

Configurations

```
Terminal
user@computer ~/livv-dev/LIVV $python livv.py -m myNewConfig
-----
Land Ice Verification & Validation (LIVV)
-----
Beginning Dependency Checks..... Okay!
Loading configuration file from /home/bzq/livv-dev/LIVV/configurations/myNewConfig

Current run: 01-22-2015 10:56:59
User: bzq
Config: myNewConfig
OS Type: Linux 3.13.0-24-generic
Wow! Look at my configuration file!

Running tests:
dome30/diagnostic
dome30/evolving
ismip-hom-a/20km
ismip-hom-c/20km
ismip-hom-a/80km
ismip-hom-c/80km
-----
Beginning test suite...
Dome Diagnostic test in progress...
Saving plot details to /home/bzq/livv-dev/LIVV/www/imgs/dome as dome30dvel.png
Running bit for bit tests of /dome30/diagnostic...
dome.4.nc FAILURE
dome.1.glissade.nc FAILURE
dome.1.nc FAILURE
dome.out.glissade.nc FAILURE
dome.out.nc SUCCESS
```

Templating

```
<HTML>
<HEAD>
  {% include 'header.html' %}

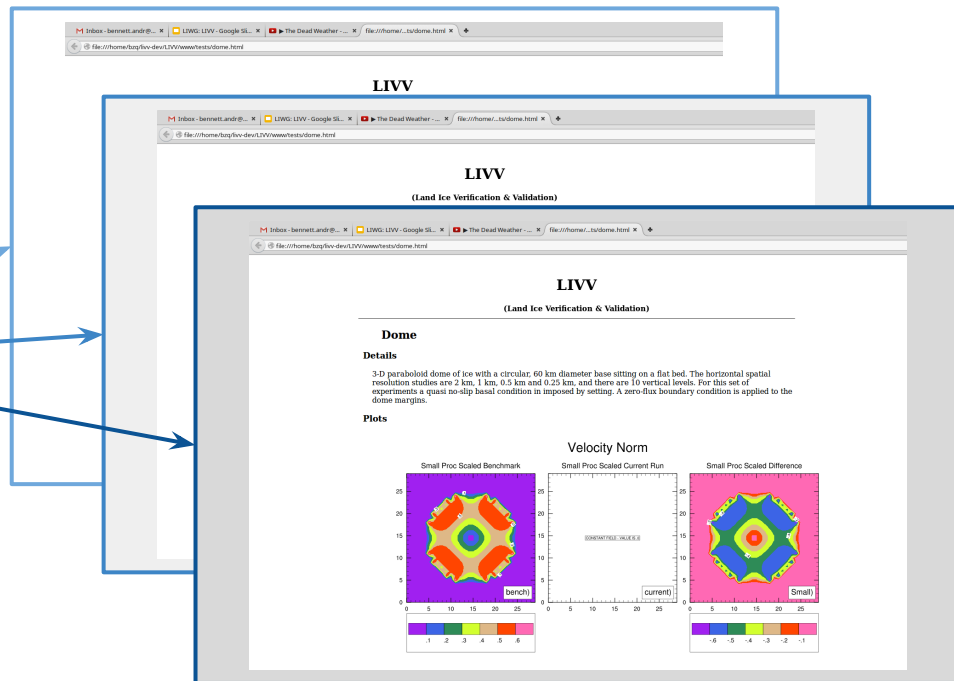
</HEAD>
<body>
  <h2>{{testName}}</h2>

  <h3>Details</h3>
  <p>
    {{testDescription}}
  </p>

  <h3>Plots</h3>
  <p>
    {% for img in testImages %}
      
      <br>
    {% endfor %}
  </p>

  <h3>Results</h3>
  {% for case in testsRun %}
    <h4>{{case}}</h4>
    <p id="bold"> Bit for Bit </p>
    <p>Bit for bit results of {{case}}.</p>
    <p id="bold"> Standalone </p>
    <p>Results for {{case}} go here.</p>
  {% endfor %}
  <a href="{{indexDir}}/index.html">Home</a>
</body>

  {% include 'footer.html' %}
</HEAD>
</HTML>
```



The Future

- Support for additional dycores
- Upgrades to performance testing
- Comparison of models and observational data
- Tighter integration with CISM