

Open Source – Of Course!



Lessons Learned from two decades of
Scientific Open Source Software development

Dr Ole Nielsen

Last Century

- Became aware of Linux, GNU emacs and Netlib (1994): Disks mailed to Bulgaria!
- Published Matlab, C and F90 toolboxes for wavelet transforms (1997): A procrastinating PhD student.
- Discovered Python (1999): Boss didn't want to pay for Matlab

Early Days

- Datamining Web Tools (Australian National University 2001)
- Required caching module (persistent memoization)

result = func(args, kwargs)

result = cache(func, args, kwargs, dependencies = <files>)

130 O.M. Nielsen et al.

Table 1. Function Caching Statistics

Function Name	Hits	Time (sec)		Gain(%)	Size (MB)
		Exec	Cache		
execquery	4,149	130	6	91.43	4.53
get_mbs_patients	172	1,281	76	93.92	48.53
get_selected_transactions	420	1,507	5	99.33	6.67
multiquery	46	133	0	99.69	0.76
simplequery	5	50	0	99.86	0.08
get_cohort	168	489	0	99.92	0.20
get_drug_usage	95	1,388	0	99.99	0.02

The user can take advantage of this caching technique by applying it to arbitrary Python functions. However, this technique has already been employed extensively in the *Data Manager* module so using the high level toolbox routines will utilise caching completely transparently with no user intervention – the caching supervision has been done in the toolbox design. For example, most of

Pushing Python (pypar - 2002)

[My favorites](#) ▼ | [Sign In](#)



[Project Home](#) [Downloads](#) [Wiki](#) [Issues](#) [Source](#)

[Summary](#) [Updates](#) [People](#)

Project Information

[Activity](#) Medium
[Project feeds](#)

Code license
[GNU GPL v3](#)

Labels
[Pypar](#), [Python](#), [MPI](#), [ParallelProgramming](#),
[HighPerformanceComputing](#), [ScientificComputing](#)

Members
[Ole Moller-Nielsen@gmail.com](#)
[1 contributor](#)

Featured

Downloads
[pypar-2.1.4_94.tgz](#)
[Show all »](#)

Links

External links
[Old SourceForge home](#)
[Managing a pool of MPI processes with Python and Pypar](#)
[Vladimir Lazunin's Pypar page](#)
[Parallel Programming with Python](#)

Groups
[Help and support](#)

Pypar

Pypar is an efficient but easy-to-use module that allows programs written in Python to run in parallel on multiple processors and communicate using message passing. Pypar provides bindings to a subset of the message passing interface standard MPI.

Features

- **Flexibility:** Pypar allows communication of general Python objects of any type.
- **Intuitive API:** The user need only specify what to send and to which processor. Pypar takes care of details about data types and MPI specifics such as tags, buffers, communication status and communicators. Receiving is analogous.
- **Efficiency:** Full bandwidth of C-MPI programs is achieved for consecutive Numerical arrays. Latency is less than twice that of pure C-MPI programs. Test programs to verify this are included (pytiming.py, ctiming.c)
- **Lightweight:** Pypar consists of just two files: mpiext.c and pypar.py
- **Python interpreter is not modified:** Parallel python programs need only import the pypar module.

Dependencies and Installation

- Pypar requires Python, numpy, C compiler and MPI C library such as openmpi
- Pypar installs with distutils: python setup.py install
- Pypar is mostly used on Linux systems, but has been tested on others too.

Example

```
import pypar                                # Import module and initialise MPI

proc = pypar.size()                         # Number of processes as specified by mpirun
myid = pypar.rank()                        # Id of this process (myid in [0, proc-1])
node = pypar.get_processor_name()          # Host name on which current process is running

print 'I am proc %d of %d on node %s' % (myid, proc, node)

if myid == 0:                               # Actions for process 0:
    msg = 'P0'
    pypar.send(msg, destination=1)          # Send message to proces 1 (right hand neighbour)
    msg = pypar.receive(source=proc-1)      # Receive message from last process

    print 'Processor 0 received message "%s" from processor %d' % (msg, proc-1)

else:                                       # Actions for all other processes:

    source = myid-1                         # Source is the process to the left
    destination = (myid+1)%proc             # Destination is process to the right
                                           # wrapped so that last processor will
                                           # send back to proces 0

    msg = pypar.receive(source)             # Receive message from source
    msg = msg + 'P' + str(myid)            # Update message
    pypar.send(msg, destination)           # Send message to destination

pypar.finalize()                           # Stop MPI
```

Some projects and publications that use Pypar

- [Managing a pool of MPI processes with Python and Pypar](#)
- [Simplifying the parallelization of scientific codes by a function-centric approach in Python](#)
- [Parallelizing PDE Solvers Using the Python Programming Language \(PDF\)](#)
- [Python for CFD \(PDF\)](#)
- [ANUGA Hydrodynamic Modelling](#)
- [EQRM Earthquake Risk Modelling](#)
- [Volcanic Ash Modelling](#)


```

"""Parallel program computing the Mandelbrot set using static,
cyclic load balancing.

This is probably the best approach for this problem.

Ole Nielsen, SUT 2003
"""

from mandelbrot import calculate_region_cyclic
from mandelplot import plot
import pypar

# User definable parameters
kmax = 2**15 # Maximal number of iterations (=number of colors)
M = N = 700 # width = height = N (200 or 700)

# Region in complex plane [-2:2]
real_min = -2.0
real_max = 1.0
imag_min = -1.5
imag_max = 1.5

# Initialise
t = pypar.time()
P = pypar.size()
p = pypar.rank()
processor_name = pypar.get_processor_name()
print 'Processor %d initialised on node %s' % (p, processor_name)

# Parallel computation
A = calculate_region_cyclic(real_min, real_max, imag_min,
                           imag_max, kmax,
                           M, N, p, P)

print 'Processor %d: time = %.2f' % (p, pypar.time() - t)

# Communication phase
if p == 0:
    for d in range(1, P):
        A += pypar.receive(source=d)

    print 'Computed region in %.2f seconds' % (pypar.time() - t)
    plot(A, kmax)
else:
    pypar.send(A, destination=0)

pypar.finalize()

```

• Standard MPI

```

int MPI_Send(
    void *buf,
    int count,
    MPI_Datatype datatype,
    int dest,
    int tag,
    MPI_Comm comm
);

```

Pypar

```
pypar.send(A, p)
```

10 years on Pypar is still used

The `handympi` Module

Steve Spicklemire

April 2, 2012

Why another MPI module?

MPI [1] is a technology that allows multiple processes, even on separate computing platforms, to interoperate cooperatively to generate higher performance than would be possible with a single process. The `handympi` module leverages the simple python package `pypar` [2] to provide a novice programmer easy access to some the potential of MPI without requiring mastery of any of the intricacies of the MPI/pypar API.

The `handympi` module provides two basic functions: `foreach` (inspired by the similarly named `handythread` module) and `RunMasterSlave`.

The easy way: `foreach`

The `foreach` function is the easiest to use. Just pass in a function and a list. The function is applied to each element in the list. If you set `return_` to 'True', then the return values for each element of the list are accumulated and returned. What's so amazingly cool about that? Well, what's fun is that if there is no MPI environment, the function just iterates

Python in Government (2004-2009)

ANUGA

Open Source

Hydrodynamic Modelling

Ole Nielsen

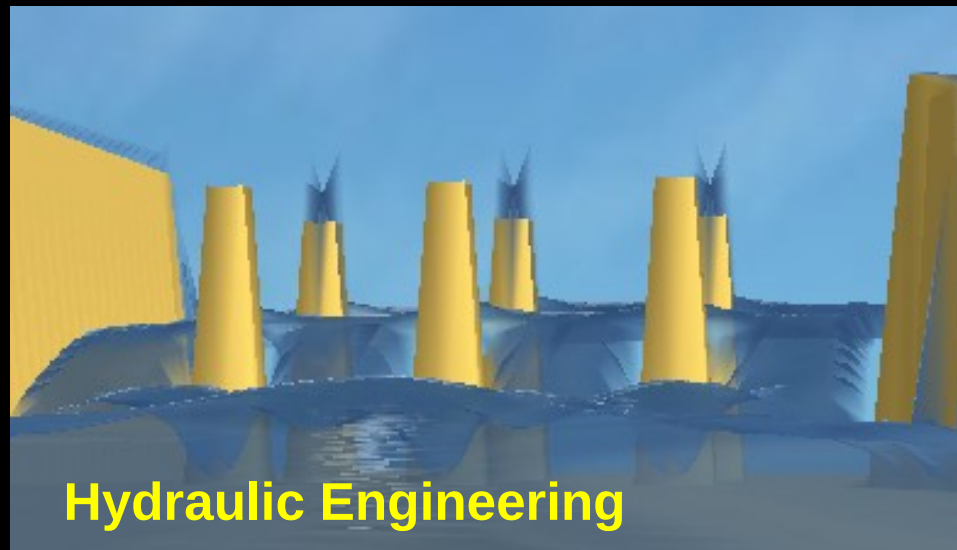
What is ANUGA



Tsunami Inundation



Riverine Flooding



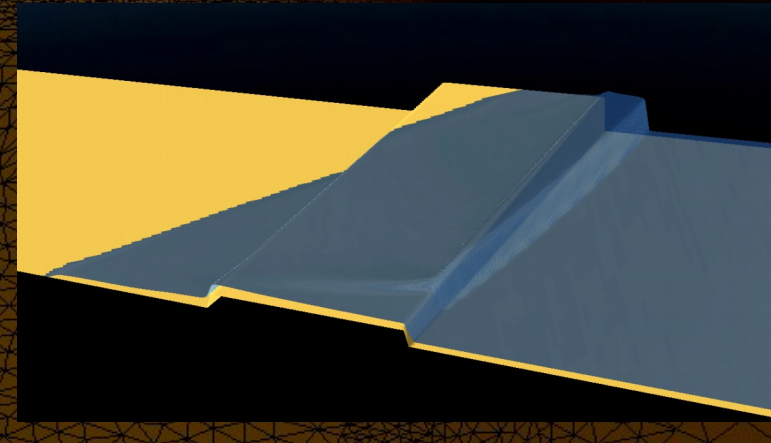
Hydraulic Engineering



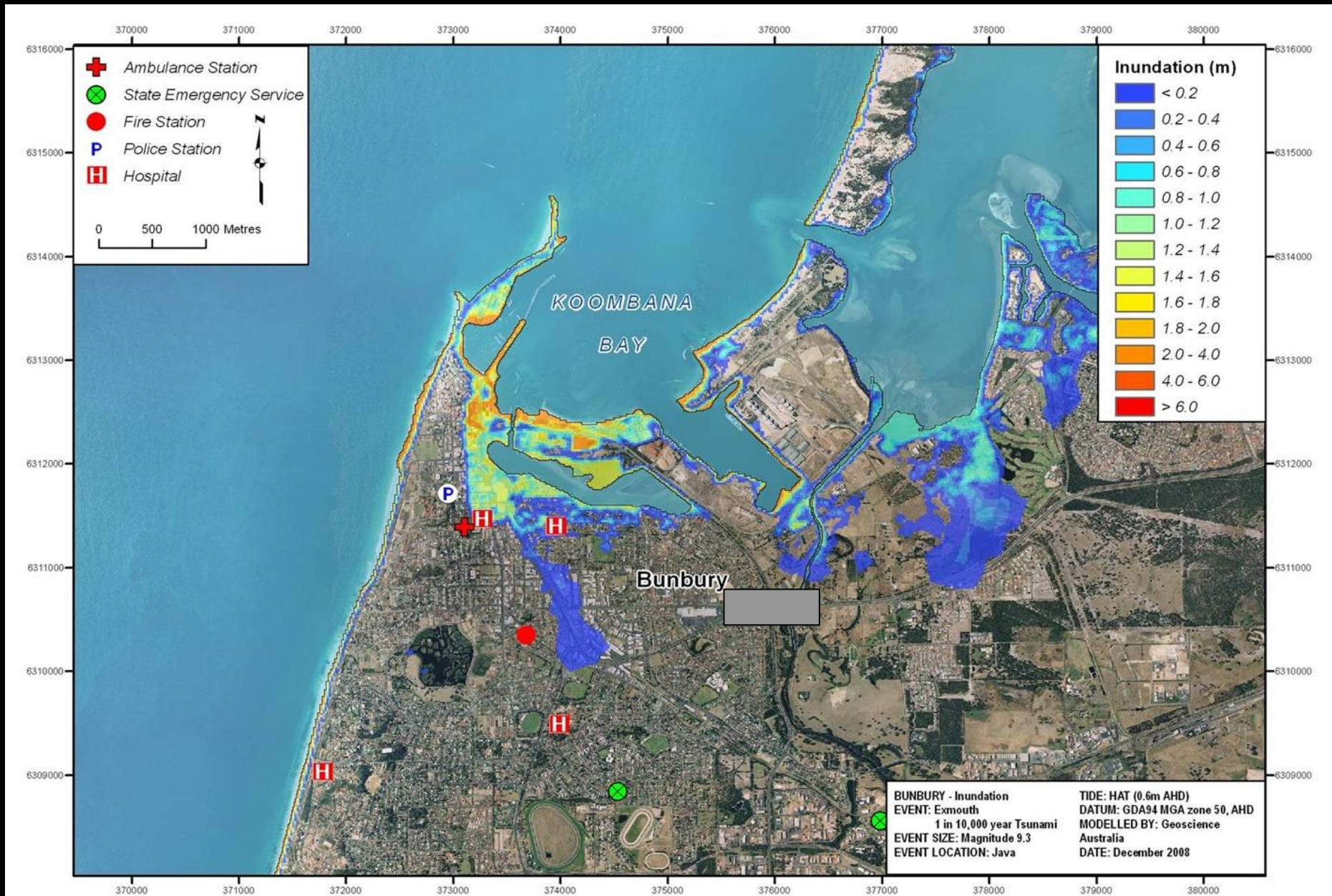
Dam Breaks

ANUGA Capabilities

- Resolution of Hydraulic shocks
- Transitions from sub- to super critical flows
- Robust wetting/drying capability
- Flexible API (e.g. dynamic typing)

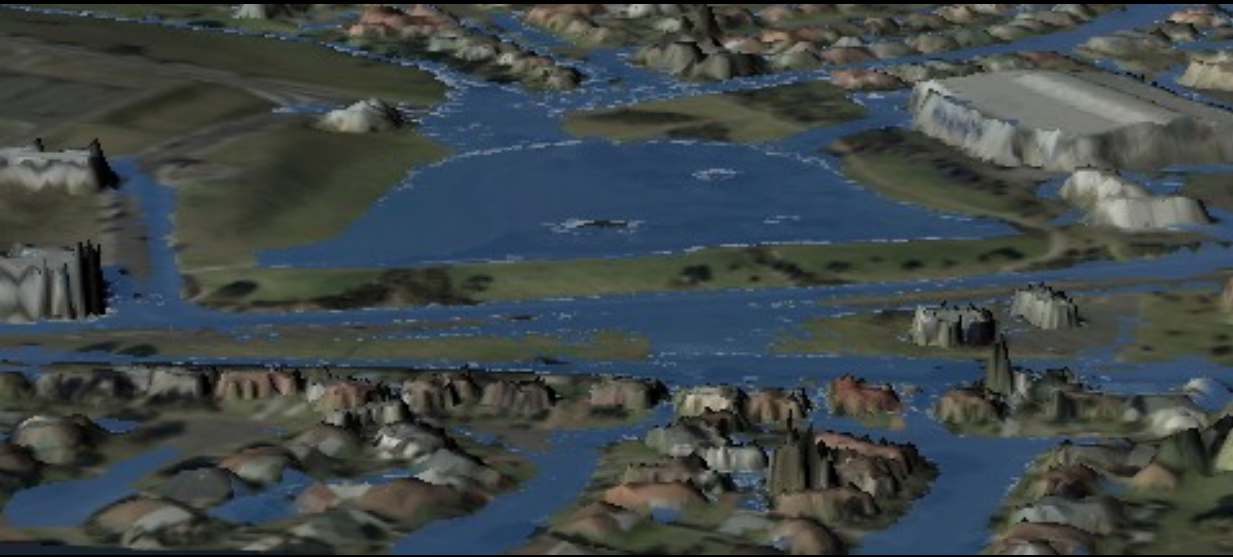


Australian Tsunami Inundation



Flood Related Applications:

- Assess the impact of new developments



Without proposed buildings



With proposed buildings

- 110km² Full 2-D Model

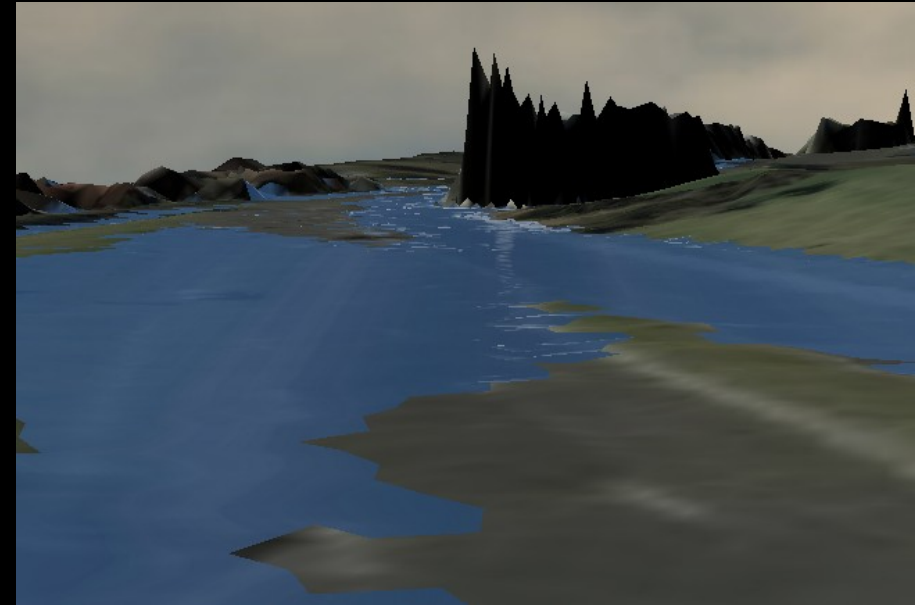


- Investigating existing Flood problems...

Real World Flood...



ANUGA interpretation



Predicting Tsunami Run Up – The Importance of Modelling

Okushiri Island 1993 Tsunami

- Magnitude 7.8 earthquake
- 32 m run up height
- Numerical Simulation of wave tank experiment shows why



Sea of Japan

**Okushiri
Island**

**Monai
Valley**



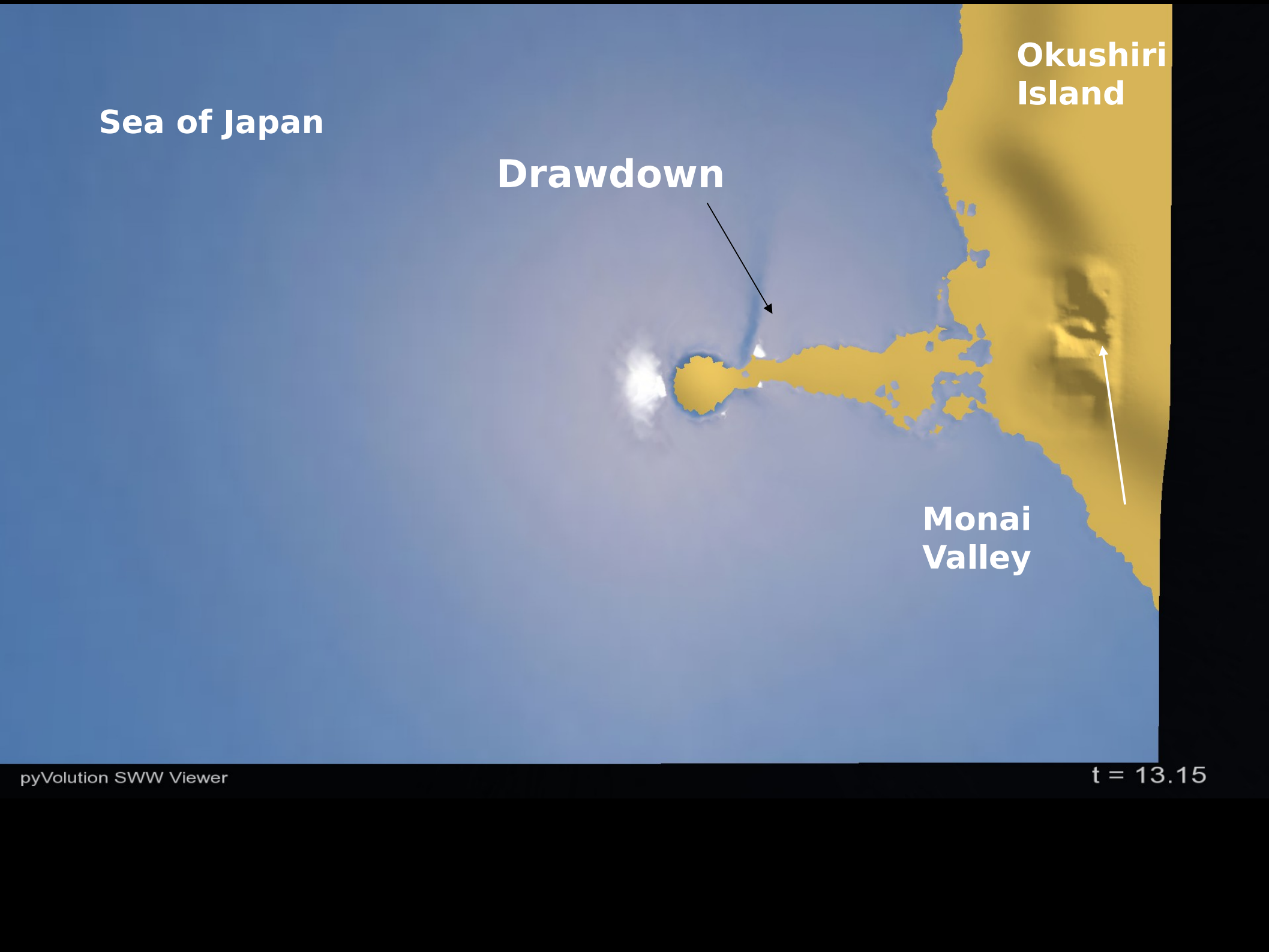
Sea of Japan

**Okushiri
Island**

Drawdown

**Monai
Valley**

t = 13.15



Sea of Japan

**Okushiri
Island**

**Monai
Valley**



Sea of Japan

**Okushiri
Island**

**Monai
Valley**

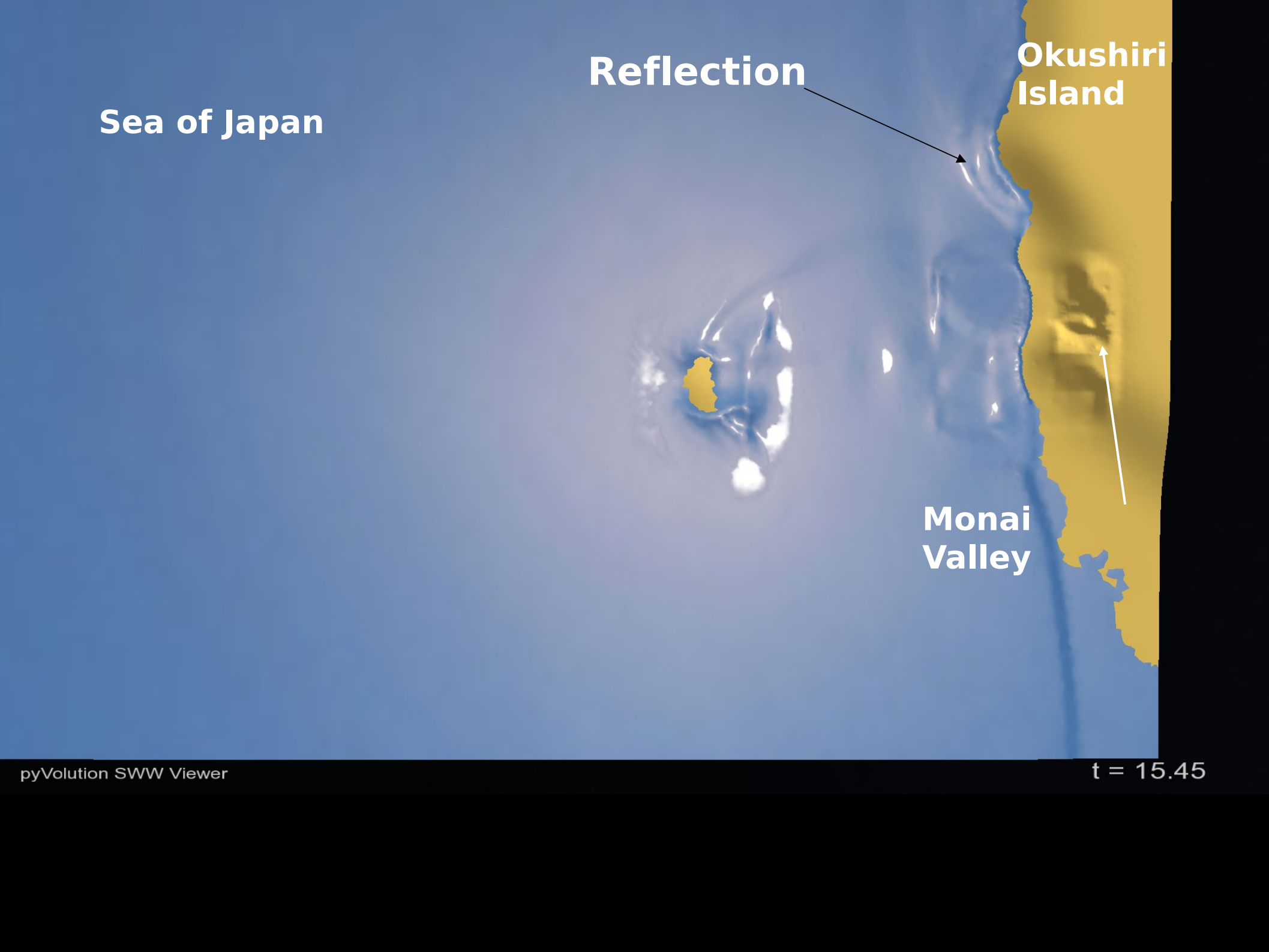
t = 15.15

Sea of Japan

Reflection

**Okushiri
Island**

**Monai
Valley**

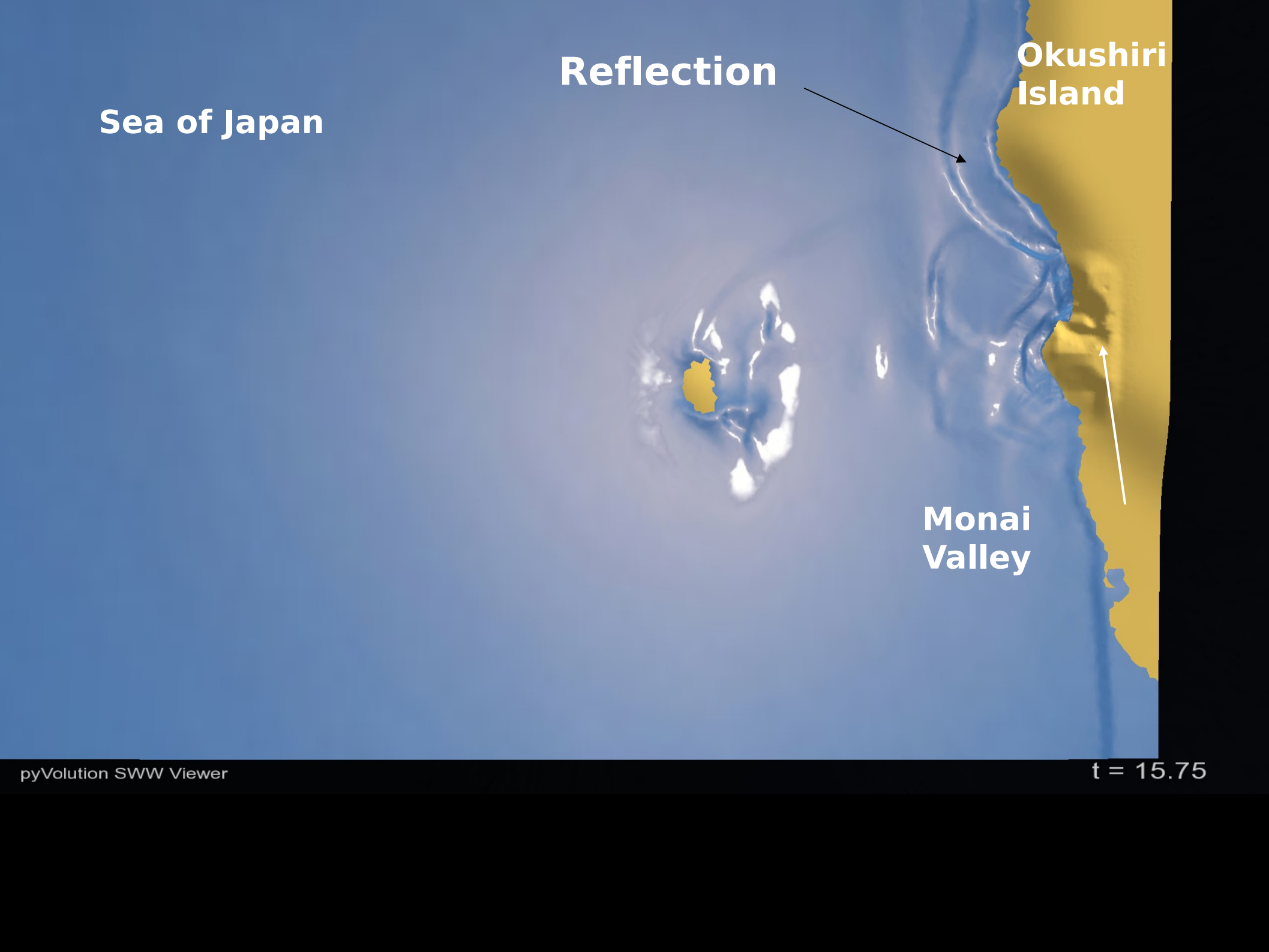


Sea of Japan

Reflection

**Okushiri
Island**

**Monai
Valley**



Sea of Japan

**Okushiri
Island**

Shoaling

**Monai
Valley**

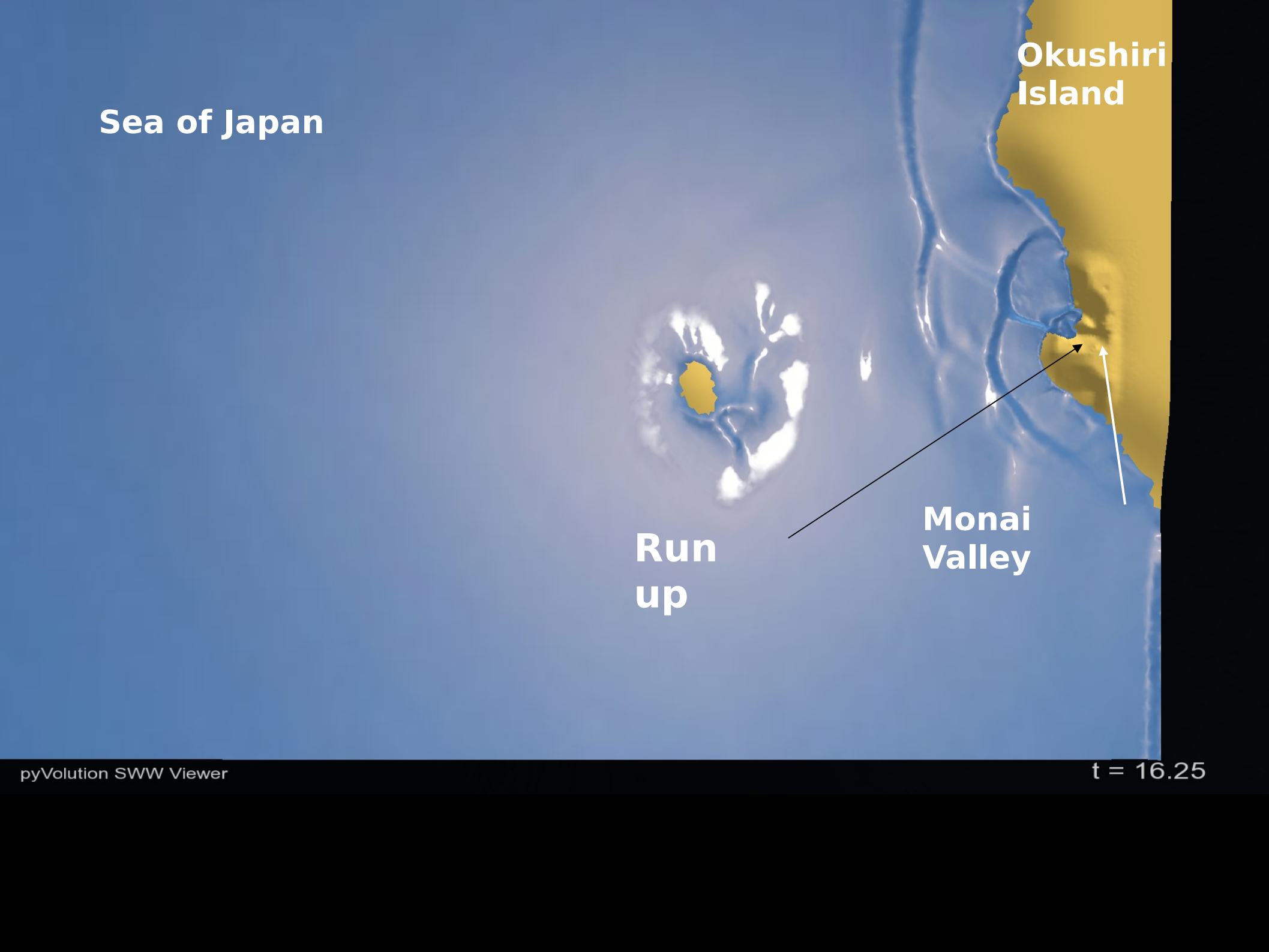


Sea of Japan

Okushiri
Island

Run
up

Monai
Valley



Sea of Japan

**Okushiri
Island**

**Run
up**

**Monai
Valley**



Sea of Japan

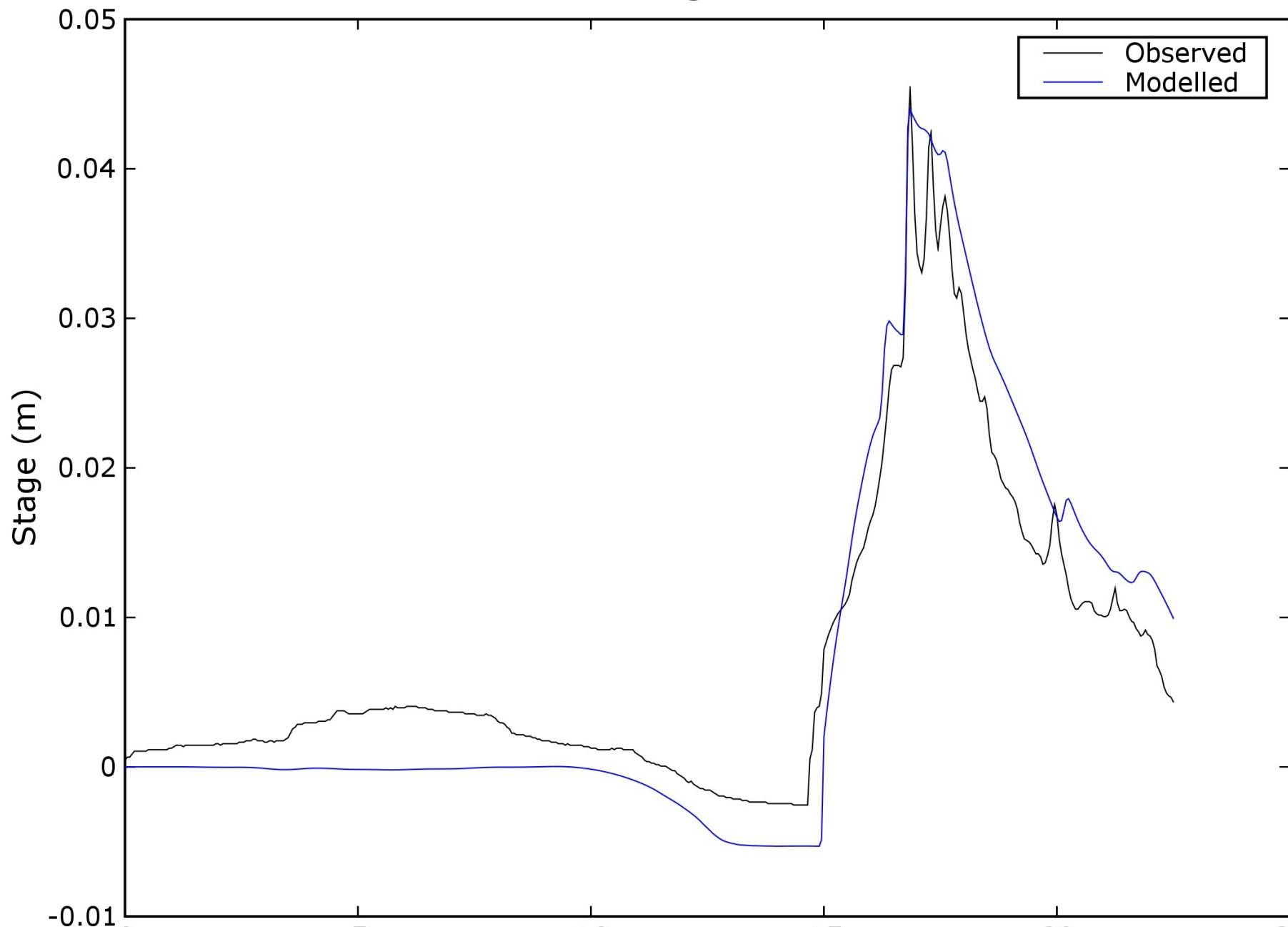
**Okushiri
Island**

**Run
up**

**Monai
Valley**

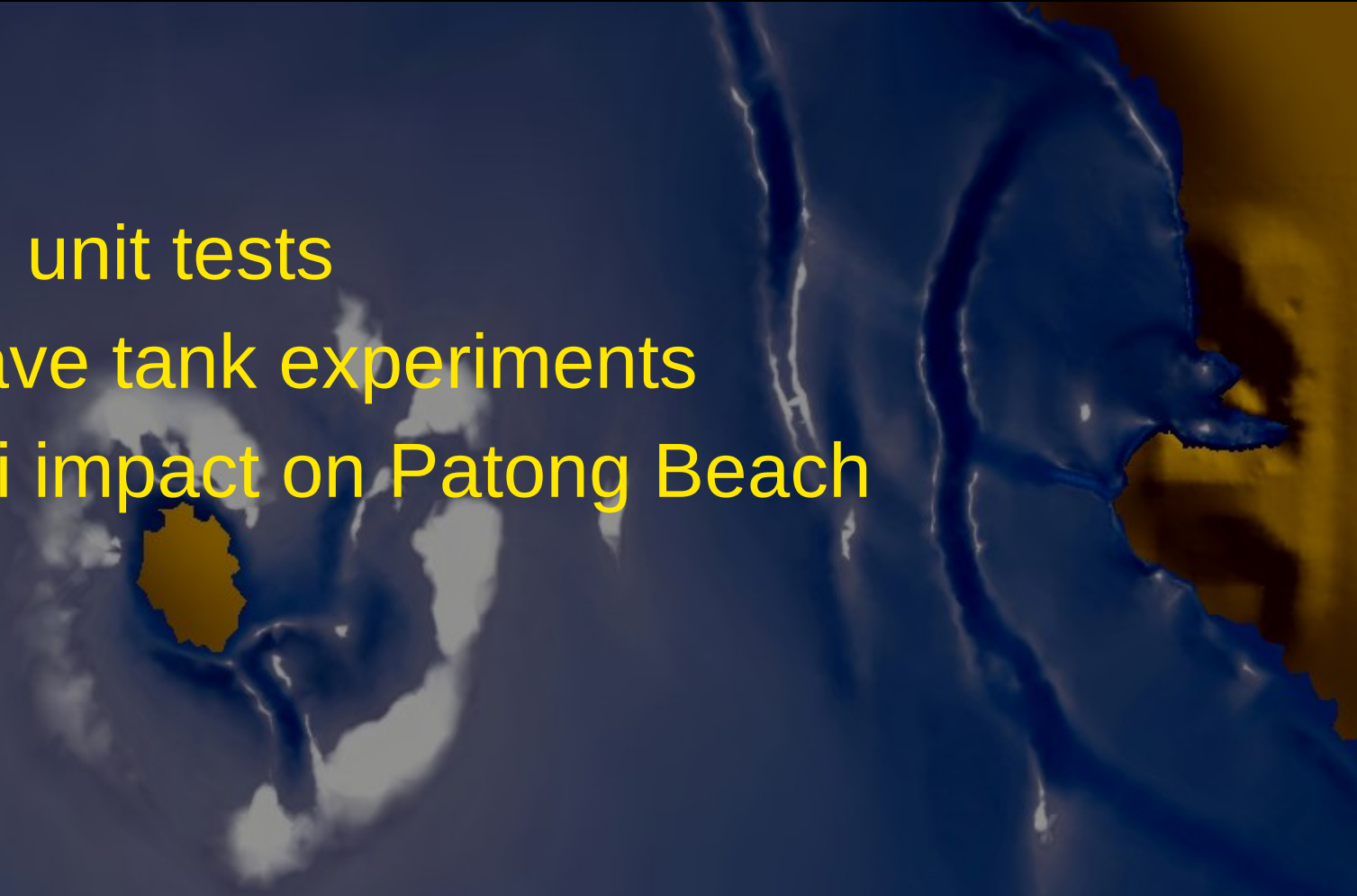


Gauge ch9



Testing and validation

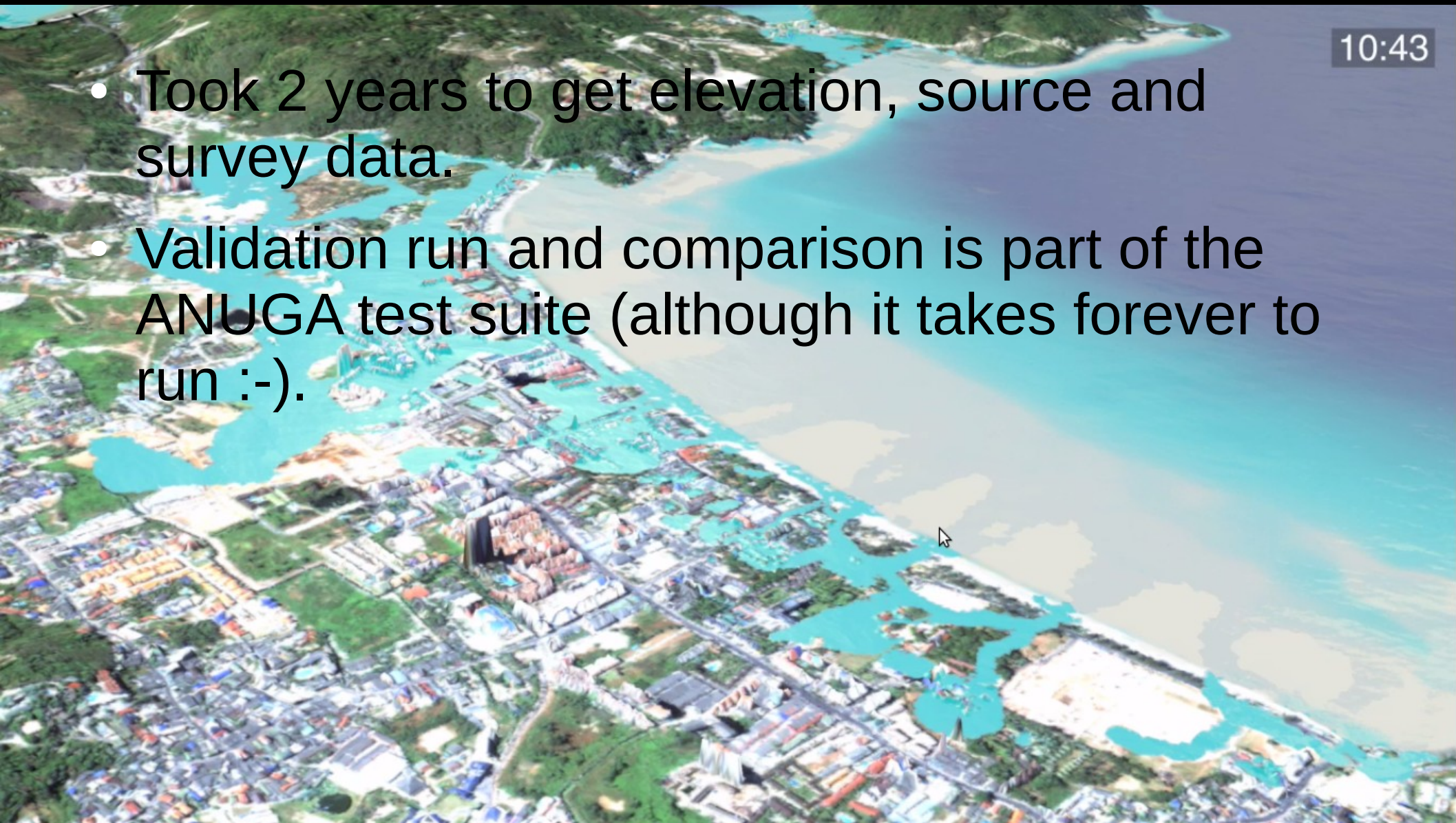
- 890 individual unit tests
- Numerous wave tank experiments
- 2004 Tsunami impact on Patong Beach



Validation 2004 Tsunami

10:43

- Took 2 years to get elevation, source and survey data.
- Validation run and comparison is part of the ANUGA test suite (although it takes forever to run :-).



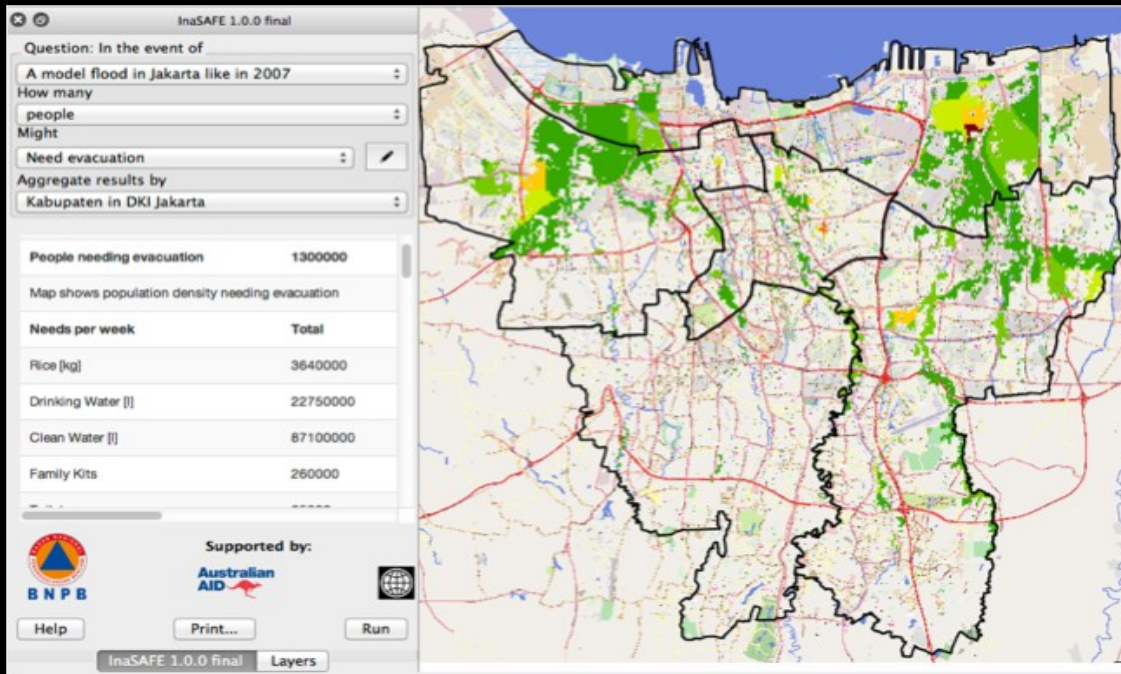
Featured on Australian TV



Further information:

Email: Ole.Nielsen@ga.gov.au
Stephen.Roberts@anu.edu.au

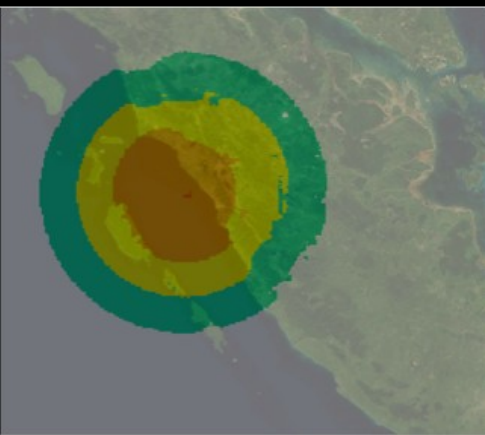
Google Search: [ANUGA software](#)
(ANUGA by itself will get German food fair)



InaSAFE

Indonesia Scenario Assessment For Emergencies





Hazard Maps
(Ground shaking, Inundation, etc)
Science Agencies



Population and Buildings
Mapping and statistical agencies
Also Communities

Decision Making





NathanW: "I must say, I think the thing that has really attracted me to try and do some development on QGIS is just how well the code is organized. It's very easy to just pick a class and go and build something. Good work guys. I don't even really know C++ very well and I have found it very easy to get into."

Quantum GIS

Version 1.8.0

"Lisboa"



Main Menu

- » [Home](#)
- » [About QGIS](#)
- » [Community](#)
- » [Documentation](#)
- » [Download](#)
- » [Commercial Support](#)
- » [Developer Meetings](#)
- » [User Meetings](#)
- » [Sponsorship](#)
- » [Advanced Search](#)



Upcoming Events

QGIS Developer Meeting, Valmiera
(April 11 - 14, 2013)



Support QGIS!

[Donate](#)

[more information](#)

Welcome to the Quantum GIS Project

Quantum GIS (QGIS) is a user friendly Open Source Geographic Information System (GIS) licensed under the GNU General Public License. QGIS is an official project of the Open Source Geospatial Foundation (OSGeo). It runs on Linux, Unix, Mac OSX, Windows and Android and supports numerous vector, raster, and database formats and functionalities.

Our latest release is QGIS 1.8.0 you can read the release announcement [here](#)

Learn more about QGIS

Quantum GIS provides a continuously growing number of capabilities provided by core functions and plugins. You can visualize, manage, edit, analyse data, and compose printable maps. Get a first impression with some [screenshots](#) and a more detailed [feature list](#).

Want to learn even more?

Check the latest [User Guide](#) or learn how you can customize QGIS to fit your needs with our [API Documentation](#) and [PyQGIS Cookbook](#).

How to contribute

Quantum GIS is a volunteer driven project. We welcome contributions in the form of code contributions, bug fixes, bug reports, contributed documentation, advocacy and supporting other users on our [mailing lists](#) and [gis.stackexchange.com](#). If you are interested in actively supporting the project, you can find more information under the development menu and on the QGIS Wiki. We also welcome financial contributions in the form of sponsoring and funding.

Download Now
Free!



Get the
User Manual

Need help?
Find it here!



We are proud to present our first

Gold Sponsor :

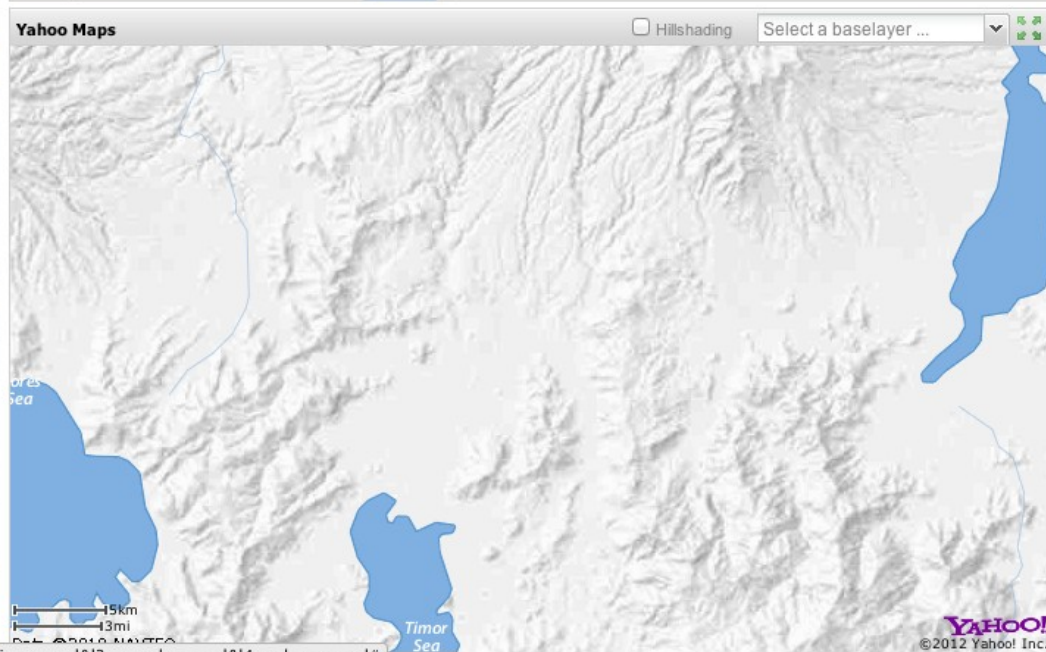
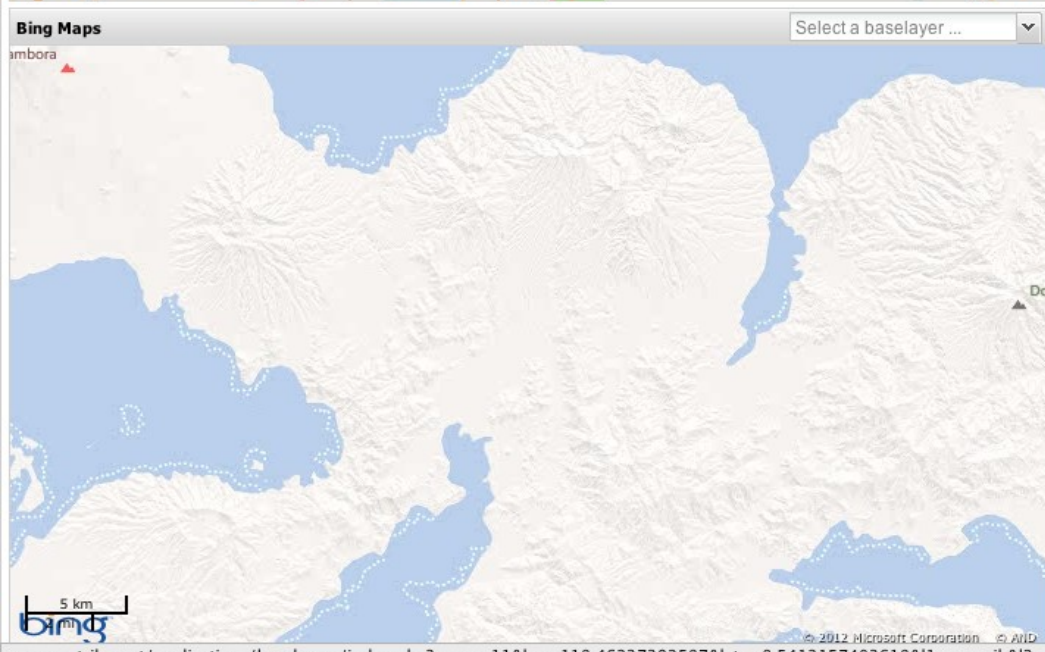
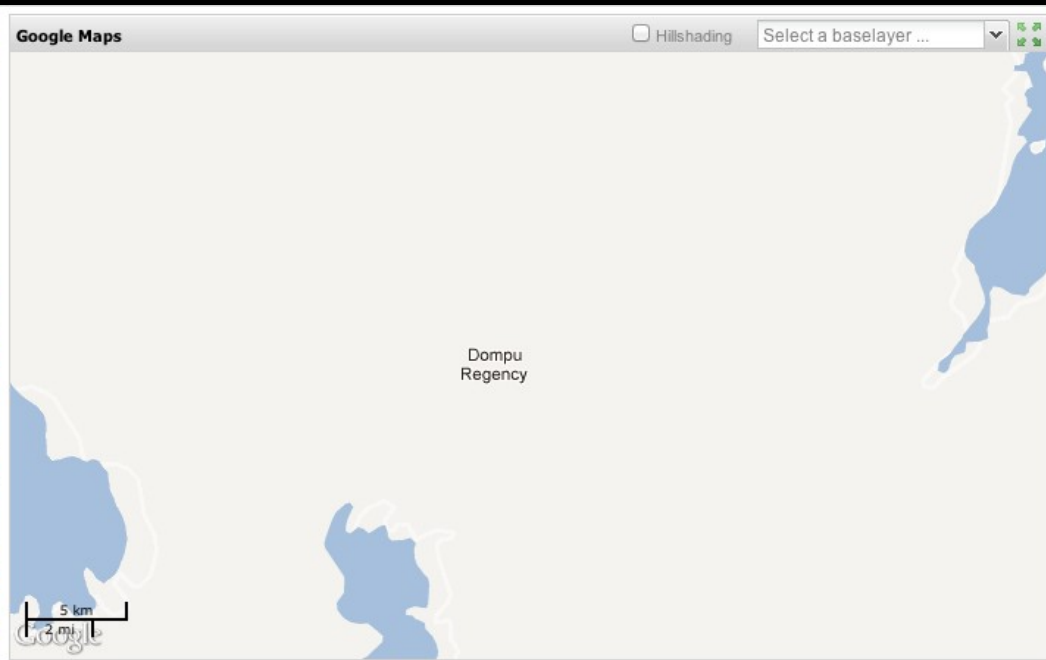
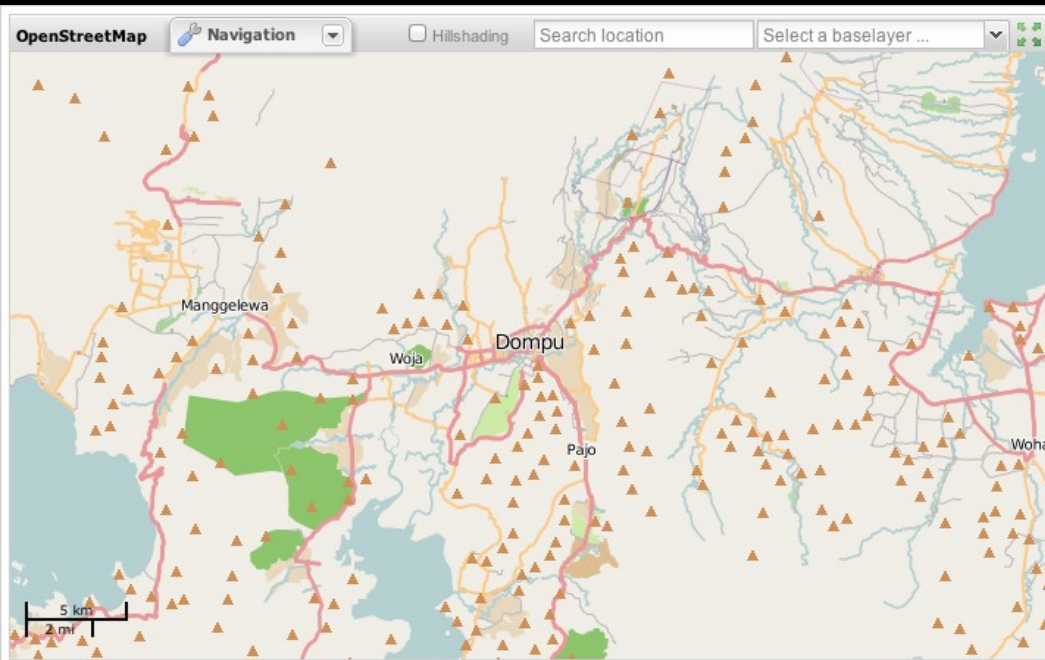
ASIA AIR SURVEY

Who has used OpenStreetMap?

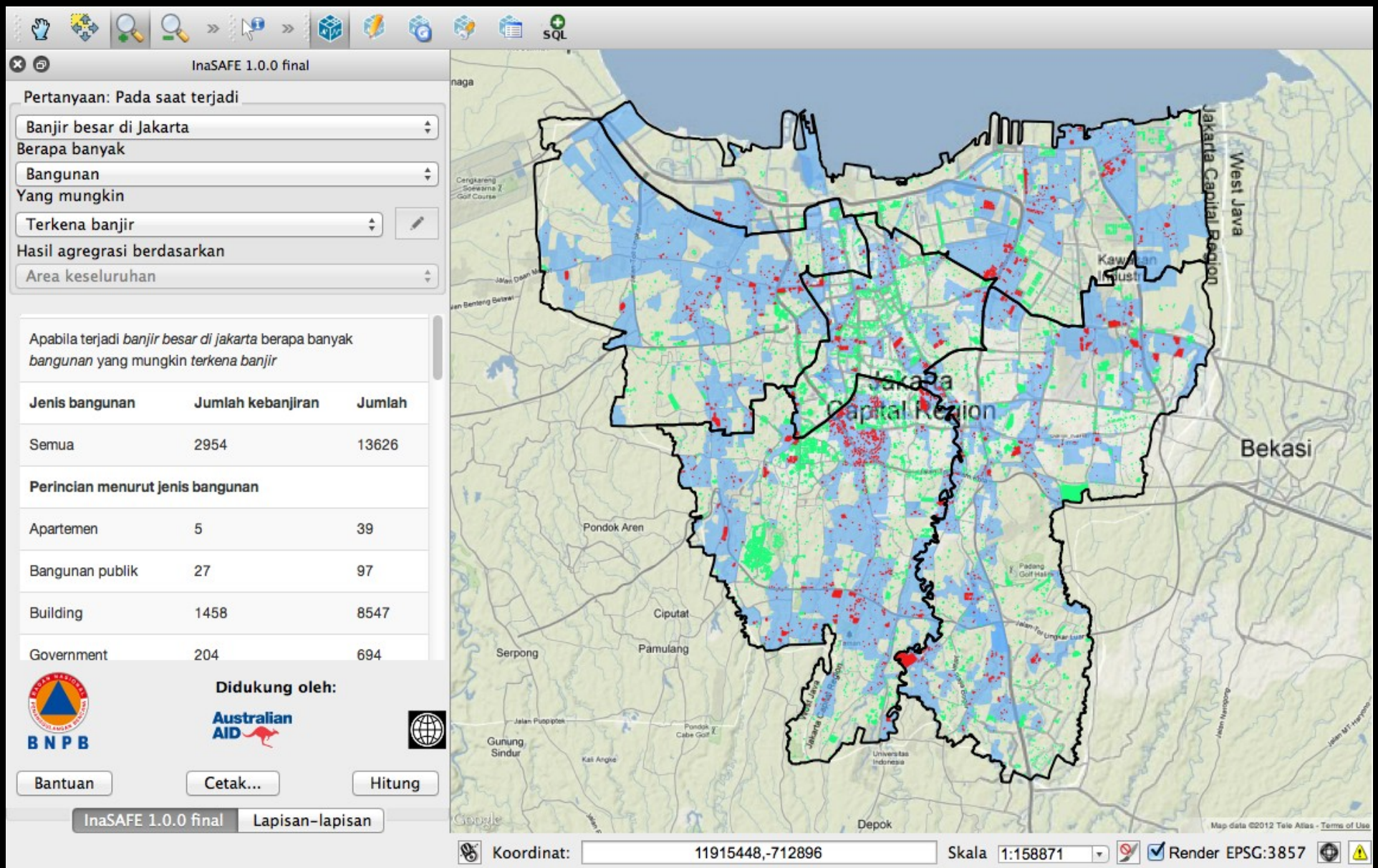
- OSM is a community driven, community owned version of Google Maps
- - **Free** (download the data and do anything with it)
 - **Open** (anyone can create new tools on top of it)
 - **Editable** (you can map what is important to you)



Spatial Data is often not available.
Open Street Map allows communities to self map







So what have I learned (if anything)?

7 noble steps to software that works



No news is good news

If it can't install – it doesn't exist

Without docs – it doesn't exist

Without testing – it doesn't exist

No future without a past

Sleep in the bed you make

Get it right – then fast

No news is good news

- Either do as requested **or** raise an exception
- Good error messages help everyone
- Exception handling allows the error to be treated at the appropriate level
 - Index error at low level becomes a matter of selecting a viewport at a higher level
 - XML parse error is typically not helpful at application level – it should explain why
 - Conditions like “file cannot open” or “division by zero” can often be handled sensibly without raising errors.

My online visa application



Best viewed in Chrome/IE 8.0 and above

If you face any issue related to security certificate with other browsers, you can [INSTALL SECURITY CERTIFICATE](#)

Indian Mission*

Please note down the Temporary Application ID: **39070935SHPVRLC**. Your Information will be saved if you click save button or continue to next page. If you do not save, your information will be lost. (Minimum field required for Partial Save is upto Date of Birth).

Applicant Details

Surname

Surname/Family Name (As in Passport)

Given Name/s*

Given Name/s (As in Passport)

Have you ever changed your name? If yes, click the box ☐ and give details.

Sex*

Date of Birth*

Date of Birth as in Passport in DD/MM/YYYY format

Town/City of birth*

Province/Town/City of birth

Country of birth*

Citizenship/National Id No.*

If not applicable Please Type NA

Religion*

If Others .Please specify

Visible identification marks*

Educational Qualification*

Nationality*

Did you acquire Nationality by birth or by naturalization?*

If you have acquired Nationality by naturalization
specify Previous Nationality

Prev. Nationality

Passport Details

Passport No.*

Applicant's Passport Number

Saving the application

Modal Page - Chromium

indianvisaonline.gov.in/visa/msg.html

Please note your Temporary File Number

39070935SHPVRLC

Temporary File Number will help you to retrieve your application data again till Final Registration of your Application.

Please do not use Back/Refresh button during your Registration.

Before final Registration you can edit your data

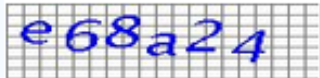
Or You can press Save And Exit to register Application later with Temporary File Number.

Close

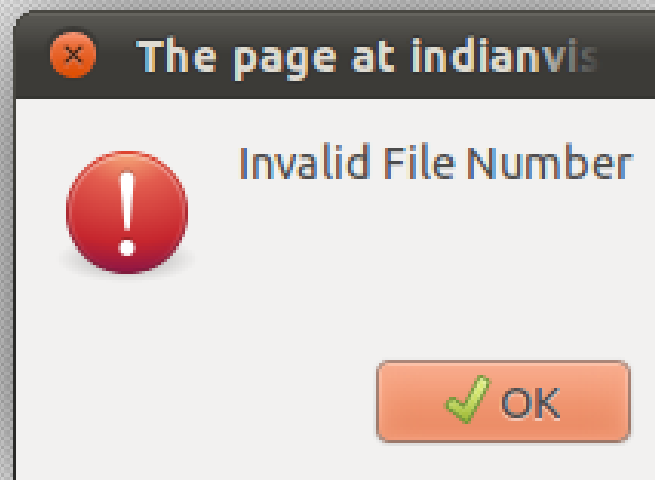
Continue To Next Page Temporary Exit

Loading it again

Online Indian Visa Form

Temporary Application ID	<input type="text" value="39070935SHPVRLC"/>
Access Code	
Enter Access Code *	<input type="text" value="e68a24"/>
	<input type="button" value="Submit"/>

What?



The Story of Ø



Online Indian

Best viewed in Chrome/IE 8.0 and above

If you face any issue related to security certificate with other browsers, you can [INSTALL SECURITY CERTIFICATE](#)

Indian Mission*

Please note down the Temporary Application ID: **39070935SHPVRLC**. Your Information will be saved if you click save but be lost. (Minimum field required for Partial Save is upto Date of Birth).


Applicant Details

Surname

Given Name/s*

Have you ever changed your name? If yes, click the box ☐ and give details.

Sex*

Date of Birth*  (DD/MM/YYYY)

Town/City of birth*

Country of birth*

Citizenship/National Id No.*

Religion*

Visible identification marks*

Educational Qualification*

Nationality*

Did you acquire Nationality by birth or by naturalization?*

Surname,

Given Na

Date of B

Province/


If not app

If Others

Other error messages I have seen

- (my>ky) failed for hidden my: regrid_smth:my=2
- syntax error: line 1, column 0
- write_dbs_grid: error in nf90_enddef

Add data

Title: 

Data:

SLD:

Abstract:

Upload


Permissions

Who can view and download this data?


- ☒ Anyone
- ☐ Any registered user
- ☐ Only users who can edit this data

Who can edit this data?

- ☐ Any registered user
- ☒ Only the following users



Who can manage and edit permissions?



Error



Unexpected error during upload: 'CoverageStore' object has no attribute 'delete'

OK

Powered by [GeoNode](#) | Need Help?



AUSTRALIA-INDONESIA
FACILITY FOR
DISASTER REDUCTION



GFDRR

github

Search...



Explore Gist Blog Help



unlornl



PUBLIC



AIFDR / riab

Admin

Pull Request

Watch

14

Fork

5

Code

Network

Pull Requests 0

Issues 80

Wiki

Graphs

Contributors

Commits

Code Frequency

Impact

Punchcard

abort: abort

Connection Error



The server returned an error: -1 transaction aborted

Details...


Cancel

Good error message

Error: An exception occurred when setting up the impact calculator. **Problem:**

No features fall within the clip extents. Try panning / zooming to an area containing data and then try to run your analysis again.

Toggle traceback...



Example of error messaging

```
try:
    infile = open(self.projection_file)
except IOError:
    msg = ('Projection file %s could not be opened. '
          'The topography file must have a projection file with '
          'extension .prj to georeference the model outputs '
          'correctly. The projection file is assumed to follow '
          'the ESRI WKT format ' % self.projection_file)
    raise Exception(msg)
```

Without docs – it doesn't exist

- Minimum
 - installation documentation
 - how to run a few examples
 - docstrings
- Ideally – documentation driven
- Critical if software is to take on its own life



Table Of Contents

[Contents](#)
[Indices and tables](#)

Next topic

[InaSAFE user documentation](#)

This Page

[Show Source](#)

Quick search

Enter search terms or a module,
class or function name.

Contents

- InaSAFE user documentation
 - Installing InaSAFE
 - Using the InaSAFE Plugin
 - Keywords System
 - InaSAFE Options
 - Impact Functions
 - Data Types
 - Using QGIS
 - Frequently Asked Questions
 - Known Issues
 - Whats new?
- InaSAFE developer documentation
 - Development under Gnu/Linux
 - Development under MS Windows
 - Coding Standards
 - Internationalisation Support
 - Version Control
 - Maintaining the documentation
 - Reporting bugs and getting help
 - Preparing a release
 - Frequently Asked Questions
 - Todo List
- InaSAFE's API documentation
 - Packages
- Authors, Contributors and Funders

Indices and tables

- [Index](#)
- [Module Index](#)
- [Search Page](#)

Bad docstrings

```
def delete_style(self, style_name):  
    """docstring for delete_style"""
```

```
def compatible_layers(func, layer_descriptors):  
    """Fetches all the layers that match the plugin requirements.  
    """
```

Good docstrings

```
def osm2padang(E):
```

```
    """Map OSM attributes to Padang vulnerability classes
```

```
    This maps attributes collected in the OpenStreetMap exposure data  
(data.kompemisiosm.org) to 9 vulnerability classes identified by  
Geoscience Australia and ITB in the post 2009 Padang earthquake  
survey (http://trove.nla.gov.au/work/38470066).
```

```
    The mapping was developed by Abigail Baca, GFDRR.
```

```
    Input
```

```
        E: Vector object representing the OSM data
```

```
    Output:
```

```
        Vector object like E, but with one new attribute ('VCLASS')  
        representing the vulnerability class used in the Padang dataset
```

```
    Algorithm
```

```
    ...
```

```
    ...
```

```
def interpolate2d(x, y, Z, points, mode='linear', bounds_error=False):  
    """Fundamental 2D interpolation routine
```

Input

x: 1D array of x-coordinates of the mesh on which to interpolate
y: 1D array of y-coordinates of the mesh on which to interpolate
Z: 2D array of values for each x, y pair
points: Nx2 array of coordinates where interpolated values are sought
mode: Determines the interpolation order. Options are
 'constant' - piecewise constant nearest neighbour interpolation
 'linear' - bilinear interpolation using the four
 nearest neighbours (default)
bounds_error: Boolean flag. If True (default) an exception will
 be raised when interpolated values are requested
 outside the domain of the input data. If False, nan
 is returned for those values

Output

1D array with same length as points with interpolated values

Notes

Input coordinates x and y are assumed to be monotonically increasing,
but need not be equidistantly spaced.

Z is assumed to have dimension M x N, where M = len(x) and N = len(y).
In other words it is assumed that the x values follow the first
(vertical) axis downwards and y values the second (horizontal) axis
from left to right.

Without testing – it doesn't exist

- Not possible to manually test all cases of a complex software project – must be automatic!
 - Known as unit-, regression-, system- or integration-testing
 - Have tests for the functionality you care about
 - The extra time invested comes back 100 fold when debugging, refactoring and moving to new platforms
 - Crucial when optimising (“Faster and faster – but wrong”!)
 - Also serves as trusted documentation
- **The single most important practice – I'd rather keep my tests than the source code!**
- InaSAFE has over 300 tests - ANUGA almost 1000,

What makes good testing

- Sets up all conditions, call the test, verify the result.
- Verify result using known invariant conditions (e.g. conservation of mass, linearly interpolated points must lie within certain bounds, inverse transforms reproduce original inputs, ...)
- Not only “happy path” but also corner cases (None, NaN, 0, max, min, %, &, \$): E.g. intersection (Parallel lines, coinciding lines, intersecting at endpoints, ...)
- Input that causes error conditions
- If a new bug is found, write new test revealing it *“It is a lot easier to debug a test than running the application”* (Tim Sutton)
- Independent of context, time and other tests
- Named after what is being tested: (see next slides)
- Good coverage (>80%) - but not 100%!
- Programmers need GUTs

Simple but real test example

```
def test_populate_polygon(self):
    """Polygon can be populated by random points
    """

    # Create non trivial polygon
    polygon = [[0, 0], [10, 10], [15, 5], [20, 10], [25, 0],
               [30, 10], [40, -10]]

    # Call function to be tested
    points = populate_polygon(polygon, 5)

    # Check that result is as expected
    assert len(points) == 5
    for point in points:
        msg = 'Point "%s" is not inside test polygon.' % str(point)
        assert is_inside_polygon(point, polygon), msg
```

#148 Linear and quantile bins are correct... ok
#149 Bounding boxes can be converted between list and string ... ok
#150 Intersections of bounding boxes work ... ok
#151 Bounding box can be buffered ... ok
#152 Centroid point data can be derived from polygon data ... ok
#153 Projections that are compatible but not identical are recognised ... ok
#154 Donut polygon can be read, interpreted and written correctly ... ok
#155 Empty keywords can be handled ... ok
#156 Bounding box can be extracted from geotransform ... ok
#157 Resolution can be extracted from geotransform ... ok
#158 Bounding box is correctly extracted from file. ... ok
#160 Vector and Raster objects can be instantiated with None ... ok
#161 Keywords can be written and read ... ok
#162 Keywords and values with colons raise error messages ... ok
#163 Vector and Raster instances have a similar API ... ok
#164 Points along line are computed correctly ... ok
#165 Bounding box minimal size can be controlled ... ok
#166 Multipart polygons are be converted to singlepart ... ok
#167 Raster layers with no projection causes Exception to be raised ... ok
#168 NODATA value is correctly handled for GDAL layers ... ok
#169 Ordering of polygon vertices is preserved when writing and reading ... ok
#170 Polygon areas are computed correctly ... ok
#171 Polygon centroids are computed correctly ... ok
#172 Polygons with inner rings can be written and read ... ok
#173 Projection information can be correctly compared ... ok
#174 Raster extrema (including NAN's) are correct. ... ok
#175 Raster layers can be converted to vector point layers ... ok

Other types of testing

- PEP8: Style checking
- Pylint: Static code analysis
- Customised checks:
 - Scan for unwanted strings, e.g. assert, settrace, ...
 - IP status of bundled data (is it ok to publish?)
 - Missing translations
 - - anything you care about, that can be automated

- [Back to Dashboard](#)
- [Status](#)
- [Changes](#)
- [GitHub](#)
- [Coverage Report](#)
- [Violations](#)
- [SLOCCount](#)
- [GitHub Hook Log](#)

Build History (trend)	
#685	Dec 16, 2012 6:49:55 AM
#684	Dec 14, 2012 7:45:38 AM
#683	Dec 14, 2012 6:42:11 AM
#682	Dec 14, 2012 6:07:30 AM
#681	Dec 13, 2012 8:35:31 AM
#680	Dec 13, 2012 7:51:34 AM
#679	Dec 12, 2012 4:02:11 PM
#678	Dec 12, 2012 2:54:21 PM
#677	Dec 12, 2012 8:01:09 AM
#676	Dec 11, 2012 11:00:39 PM
#675	Dec 7, 2012 9:54:49 AM
#674	Dec 7, 2012 5:23:39 AM
#673	Dec 6, 2012 6:30:55 PM
#672	Dec 6, 2012 3:44:43 PM
#671	Dec 6, 2012 3:30:30 PM
#670	Dec 6, 2012 3:18:00 PM
#669	Dec 6, 2012 3:11:14 PM
#668	Dec 6, 2012 7:07:54 AM
#667	Dec 5, 2012 12:43:30 PM
#666	Dec 5, 2012 10:26:58 AM
#665	Dec 5, 2012 9:53:20 AM
#664	Dec 5, 2012 9:20:40 AM
#663	Dec 4, 2012 12:31:51 PM
#662	Dec 4, 2012 11:09:32 AM
#661	Dec 4, 2012 10:35:55 AM
#660	Dec 4, 2012 8:35:14 AM
#659	Nov 30, 2012 6:23:53 PM
#658	Nov 30, 2012 3:46:06 PM
#657	Nov 30, 2012 2:22:25 PM
#656	Nov 30, 2012 9:35:18 AM
#655	Nov 30, 2012 9:00:46 AM

[More ...](#)

[RSS for all](#) [RSS for failures](#)

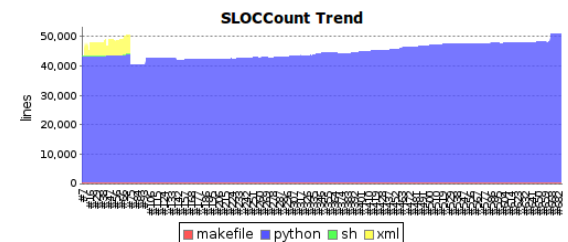
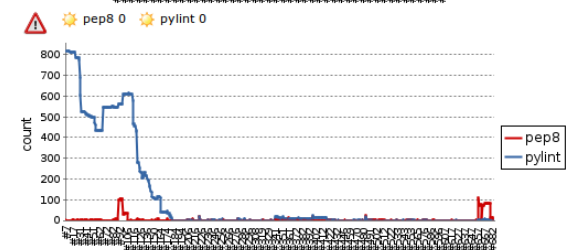
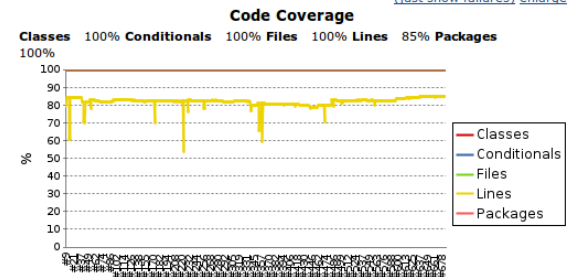
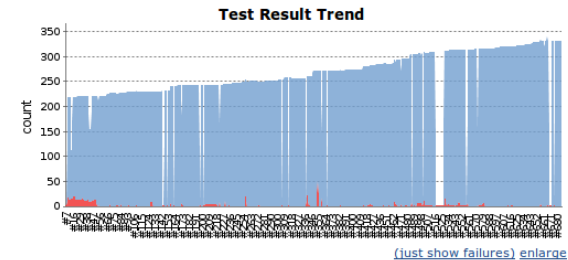
Project InaSAFE-QGIS1

Automated testing for InaSAFE against QGIS version 1.8 Because realtime work has a QGIS 2 dependency, it is excluded from these tests.

- [Coverage Report](#)
- [Recent Changes](#)
- [Latest Test Result](#) (no failures)

Permalinks

- [Last build \(#685\), 1 day 21 hr ago](#)
- [Last stable build \(#685\), 1 day 21 hr ago](#)
- [Last successful build \(#685\), 1 day 21 hr ago](#)
- [Last failed build \(#669\), 11 days ago](#)
- [Last unstable build \(#667\), 12 days ago](#)
- [Last unsuccessful build \(#676\), 6 days 4 hr ago](#)



Get it right – Then fast

Recipe refined during development of ANUGA

1. Write one or two tests
2. Implement brute force simple algorithm
3. Implement smarter algorithm and more tests
4. After profiling, write bottlenecks in C using test suite to maintain correctness

Never optimise without profiling and testing!

Why use and make Open Source

- Open Source makes software better
 - Generates interest and attracts feedback
 - Facilitates re-use
 - Allows large teams to collaborate
 - Critical to scientific reproducibility
(please include the source code in papers!)

Appeal: Get involved in QGIS!



Spatial Capabilities

Numerical Algorithms



<http://www.qgis.org>

Thank You So Much!

Questions?