Software Development Tools for SWAT

Seshu Tirupathi

IBM Research - Ireland

2017 International SWAT Conference - Warsaw

June 28, 2017

Acknowledgments : Prof. Gosain, Prof. Srinivasan

IBM

Outline

SmarterCities Technology Centre

Outline for section 1

SmarterCities Technology Centre

Outline for section 1

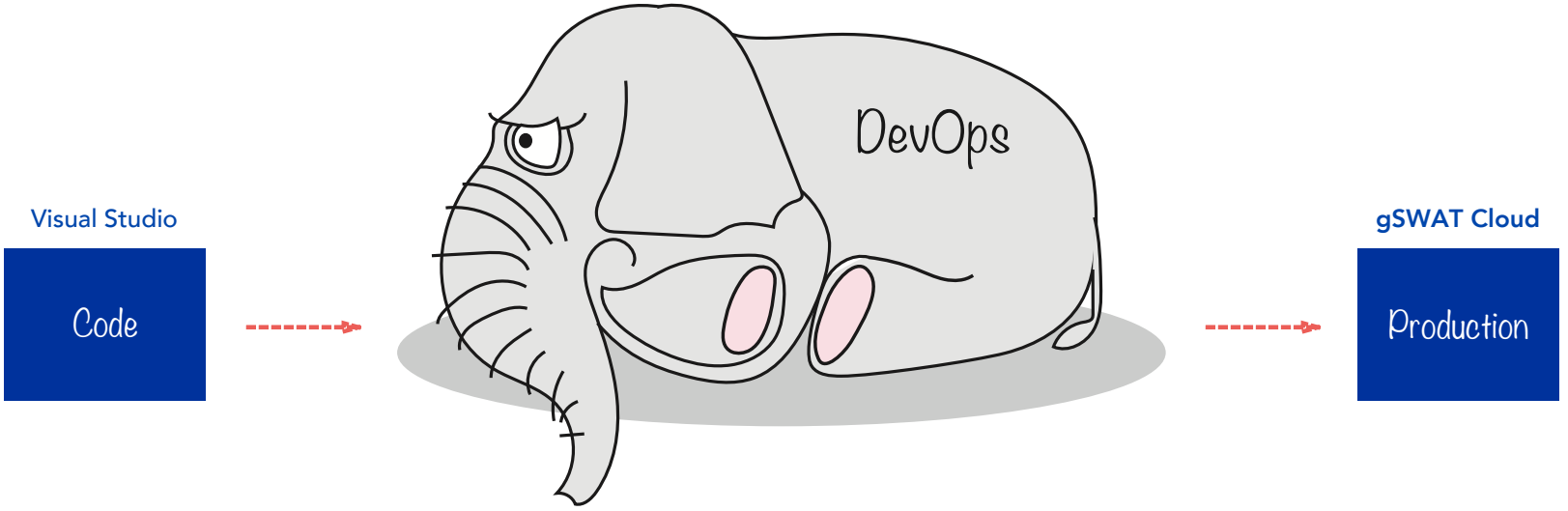
SWAT Inner Workings - Outsider's perspective

- Fortran source code.
- SWAT to SWAT+ to gSWATCloud.
- Primarily based on Intel Fortran Compiler (ifort), Visual Studio IDE.
- Open source code in the true sense. Open source standards?

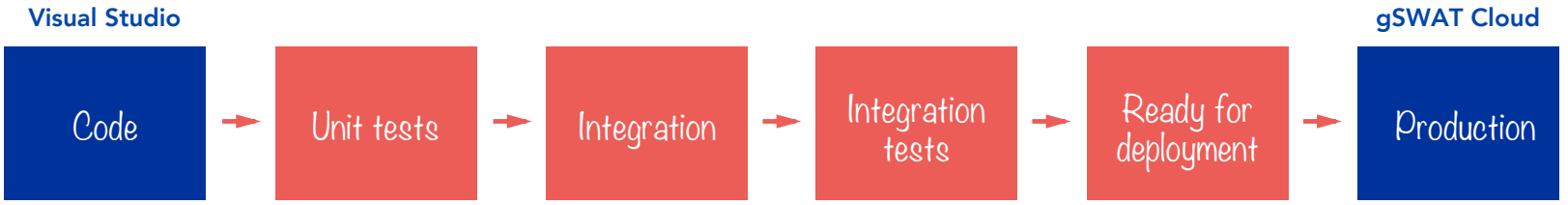
Outline for section 1

Constraints

Opportunities

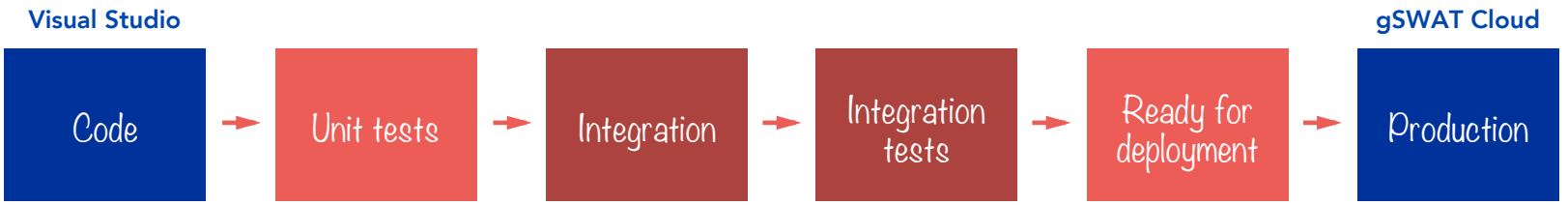


Opportunities



SmarterCities Technology Centre

Opportunities



Opportunities

SmarterCities Technology Centre

Outline for section 2

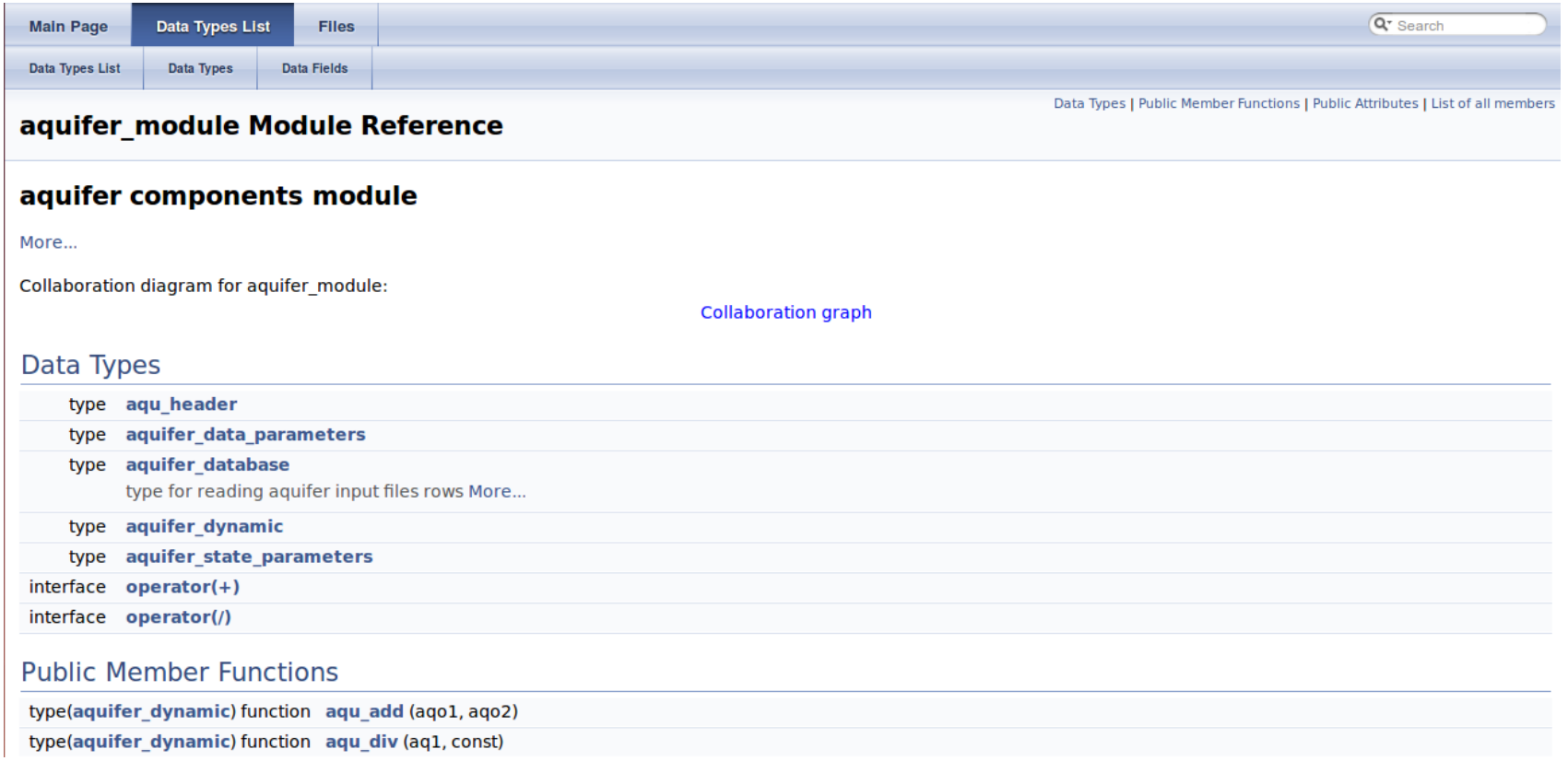
SmarterCities Technology Centre

Outline for section 2

Documentation[6]

- Easier to use and maintain the code.
- User-friendly approach to understand code logic.
- Graphical visualization of interdependencies.
- Auto-generating documentation packages for SWAT. Examples:
  - ROBODoc
  - Doxygen
  - sphinx-fortran
  - FORD
  - doctran

Doxygen SWAT Example (Credit: Unknown authour SWAT+)



Main Page | Data Types List | Files | Search

Data Types List | Data Types | Data Fields

Data Types | Public Member Functions | Public Attributes | List of all members

aquifer_module Module Reference

aquifer components module

More...

Collaboration diagram for aquifer_module:

Collaboration graph

Data Types

type	**aqu_header**
type	**aquifer_data_parameters**
type	**aquifer_database** type for reading aquifer input files rows More...
type	**aquifer_dynamic**
type	**aquifer_state_parameters**
interface	**operator(+)**
interface	**operator(/)**

Public Member Functions

type(**aquifer_dynamic**) function	**aqu_add** (aqo1, aqo2)
type(**aquifer_dynamic**) function	**aqu_div** (aq1, const)

SmarterCities Technology Centre

Doxygen SWAT Example

Public Attributes

integer	**iaq**
integer	**iaqdb**
type(**aquifer_database**), dimension(:), allocatable	**aqudb** read from the aquifer database file named aquifer.aqu More...
type(**aquifer_data_parameters**), dimension(:), allocatable	**aqu_prm**
type(**aquifer_state_parameters**), dimension(:), allocatable	**aqu_st**
type(**aquifer_dynamic**), dimension(:), allocatable	**aqu**
type(**aquifer_dynamic**), dimension(:), allocatable, save	**aqu_m**
type(**aquifer_dynamic**), dimension(:), allocatable, save	**aqu_y**
type(**aquifer_dynamic**), dimension(:), allocatable, save	**aqu_a**
type(**aquifer_dynamic**)	**aquz**
type(**aqu_header**)	**aqu_hdr**

Detailed Description

Doxygen SWAT Example

Detailed Description

aquifer components module

Author
Jeff Arnold

Version
0.0.0

Date
06 28 2016

This module includes the aquifer database parameters

REVISION HISTORY: 2016.24 test

Member Function/Subroutine Documentation

```
type (aquifer_dynamic) function aquifer_module::aqu_add ( type (aquifer_dynamic), intent(in)  aqo1,
                                                          type (aquifer_dynamic), intent(in)  aqo2
                                                        )
```

```
type (aquifer_dynamic) function aquifer_module::aqu_div ( type (aquifer_dynamic), intent(in)  aq1,
                                                          real, intent(in)                    const
                                                        )
```

Doxygen SWAT Example

Member Data Documentation

type (aquifer_dynamic), dimension(:), allocatable aquifer_module::aqu

type (aquifer_dynamic), dimension(:), allocatable, save aquifer_module::aqu_a

type (aqu_header) aquifer_module::aqu_hdr

type (aquifer_dynamic), dimension(:), allocatable, save aquifer_module::aqu_m

type (aquifer_data_parameters), dimension(:), allocatable aquifer_module::aqu_prm

type (aquifer_state_parameters), dimension(:), allocatable aquifer_module::aqu_st

type (aquifer_dynamic), dimension(:), allocatable, save aquifer_module::aqu_y

type (aquifer_database), dimension(:), allocatable aquifer_module::aqudb

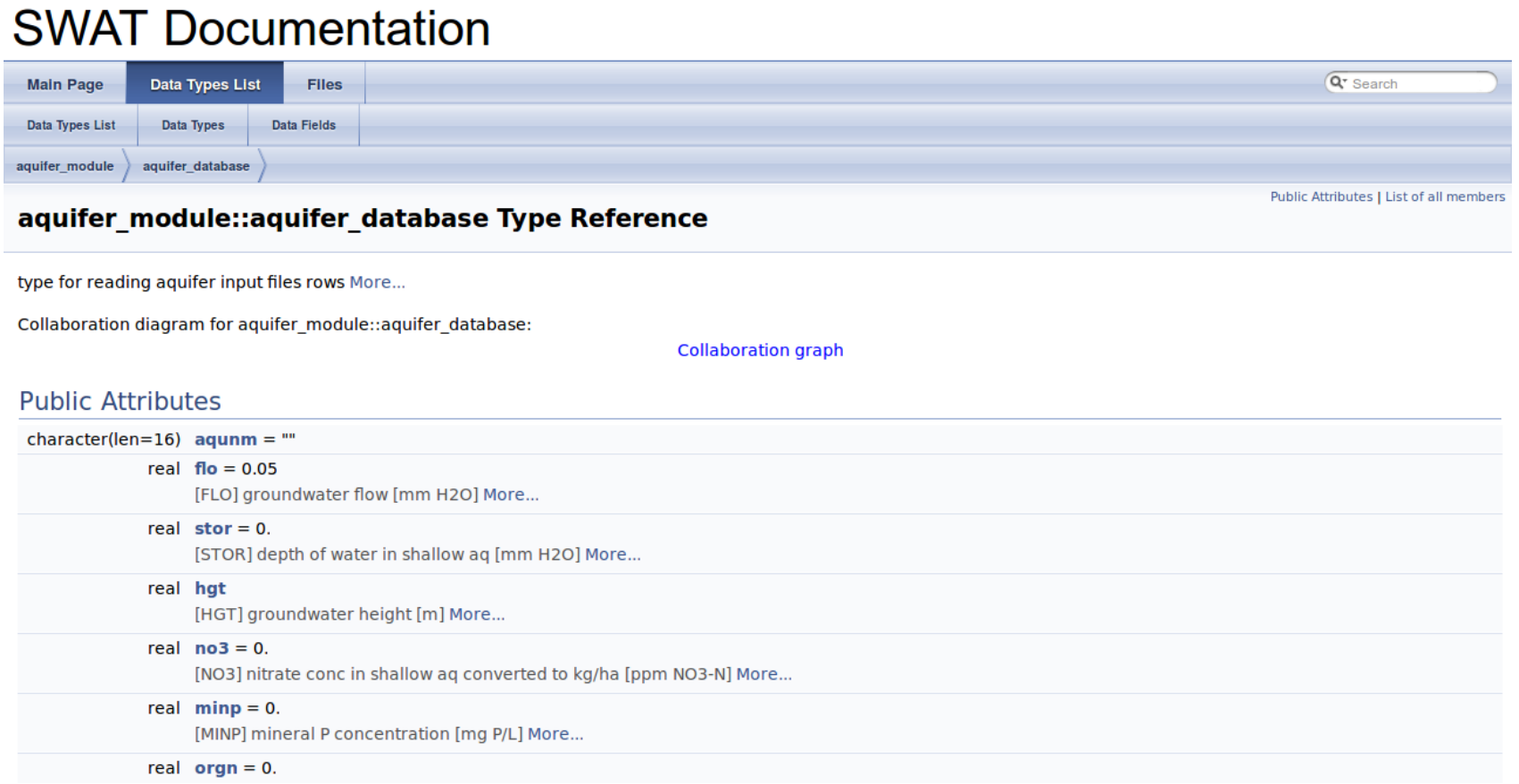
read from the aquifer database file named aquifer.aqu

See Also

aqu_read

SmarterCities Technology Centre

Doxygen SWAT Example



SWAT Documentation

Main Page | Data Types List | Files | Search

Data Types List | Data Types | Data Fields

aquifer_module | aquifer_database

Public Attributes | List of all members

aquifer_module::aquifer_database Type Reference

type for reading aquifer input files rows More...

Collaboration diagram for aquifer_module::aquifer_database:

Collaboration graph

Public Attributes

character(len=16)	**aqunm** = ""
real	**flo** = 0.05 [FLO] groundwater flow [mm H2O] More...
real	**stor** = 0. [STOR] depth of water in shallow aq [mm H2O] More...
real	**hgt** [HGT] groundwater height [m] More...
real	**no3** = 0. [NO3] nitrate conc in shallow aq converted to kg/ha [ppm NO3-N] More...
real	**minp** = 0. [MINP] mineral P concentration [mg P/L] More...
real	**orgn** = 0.

SmarterCities Technology Centre

Outline for section 2

Overview

- Test Driven Development (TDD).
- When/if done wisely, avoids glaring mistakes during code development.
- Easier integration of contributions.
- Easier maintenance of code for owners/administrators.

Options

Name	XUnit	MPI	SWAT	Comments
FRUIT[1]	✓	×	×	Ruby + Fortran, User friendly Last update: 2016-10-23. Discontinued
FLIBS[2]	✓	×	×	Tcl Make, simple to implement
ObjexxFTK[3]	✓	×	×	Python + Fortran, Not open source, paid service
FRUITPy[4]	✓	✓	✓	Python + Fortran,limited documentation. Last update 1 year back.
pFUnit[5]	✓	✓	✓	Python + Fortran, extensive documentation, regular updates

pFUnit Example

- Usage given in http://pfunit.sourceforge.net/page_Usage.html
- Write preprocessor input files which describes the modules required from pFUnit. Also includes the "assert" statements.

```
!testTheta.pf
@test
subroutine testTheta()
   use pfunit_mod
   implicit none
   @assertEqual(1.0,theta(1.0,1.0,20)) !r20*thk**(tmp-20.)
end subroutine testTheta
```

- Create a file called 'testSuites.inc' which includes information on the tests that are to be checked.

```
ADD_TEST_SUITE(testTheta_suite)
```

- Makefile to automate the build and test process.

SmarterCities Technology Centre

Outline for section 2

Build Tools

- Operating System independent build tools.
  - **Makefile**: Dependency resoulution is very difficult. Some successful efforts are available online. Not a viable long-term solution.
  - **CMake**: More viable approach for dependency hierarchy.
  - **FoBiS.py**: Automatic dependency hierarchy. Simple to implement.
  - **Other Options**[7]: Meson, Waf, fake etc
- Integrated Development Environment (IDE) beyond Visual Studio.
  - **Photran**: Eclipse Plugin + Fortran. Universal IDE.
  - **IntelliJ**: Incredible success for Java, Python, C++ etc. IntelliJ IDEA plugin for Fortran available since May 2017.
  - **Code::Blocks**: Customized version for Fortran.

Outline for section 3

Outline for section 3

SWAT Requirements

- Future of SWAT? Single platform rigid development process or platform independent process.
- Framework for integration with other open-source libraries.
- Multiple code development environments.
- Code deployment/Cloud Ready/Live in production.

Outline for section 3

Virtualization - Vagrant

Current Process

- Standard solution is to create manual instructions for getting the code to work.
- Lots of constraints and unknowns.
- Time consuming process.

Vagrant - Developers tool for creating a virtual environment using command line.

- Development environment isolated to ensure that SWAT code works.
- Easier to get started with the core development of the code.

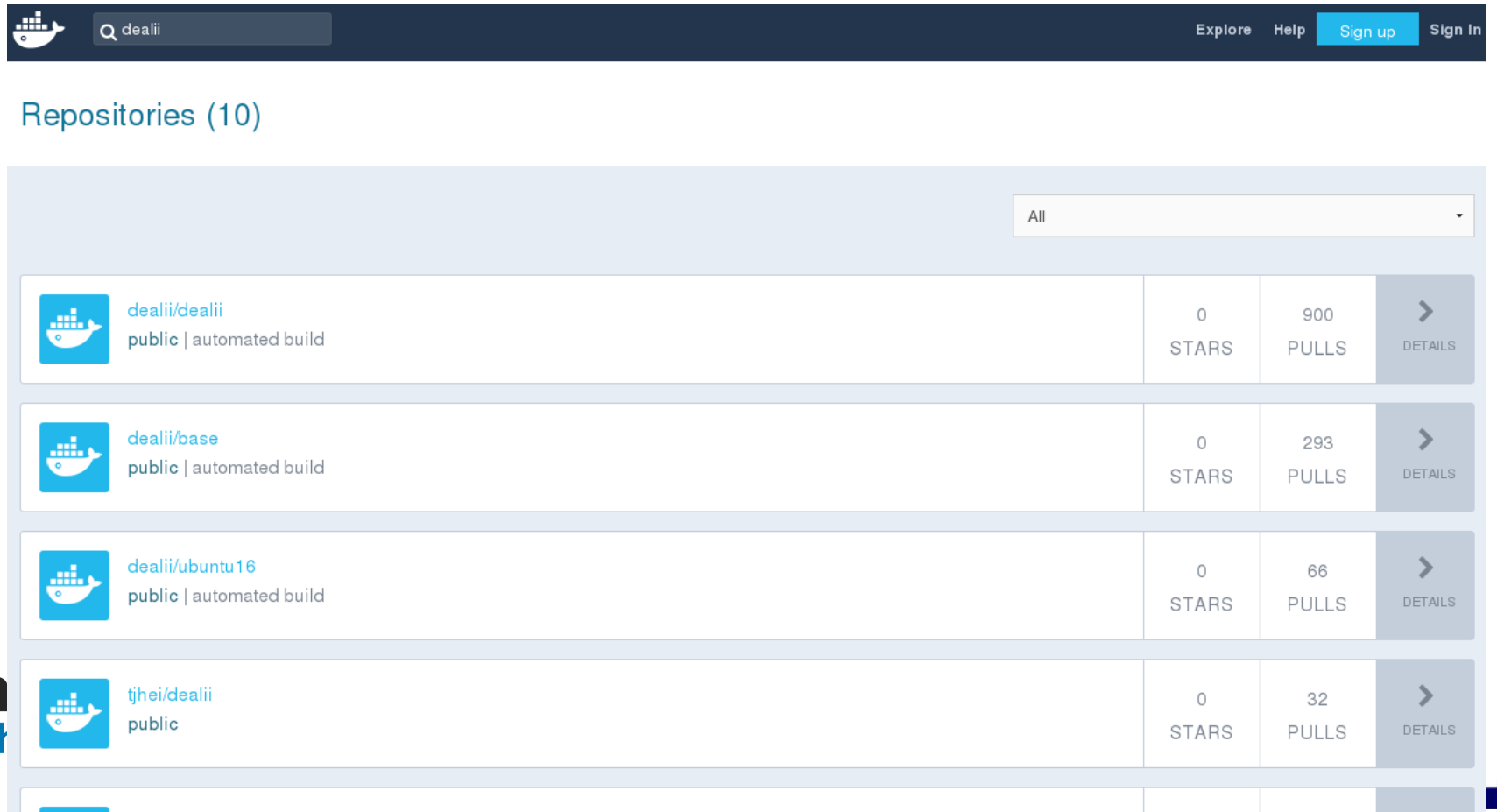
Example Vagrant File Snippet

```
Vagrant.configure(2) do |config|
  config.vm.box = "ubuntu/trusty64"
  config.vm.provider "virtualbox" do |vb|
    vb.memory = "1024"
    vb.cpus = 1
  end

------
apt-get update
apt-get install gfortran
echo "fetch swat repo"
wget http://swat.tamu.edu/media/115510/rev664_source.zip
echo "unzip swat source code"
unzip rev664_source.zip
```

Outline for section 3

Containers - Docker

- Containers - Application's components + dependencies + binaries + libraries
- Lightweight deployments that can run on any computer, infrastructure or cloud.
- hub.docker.com



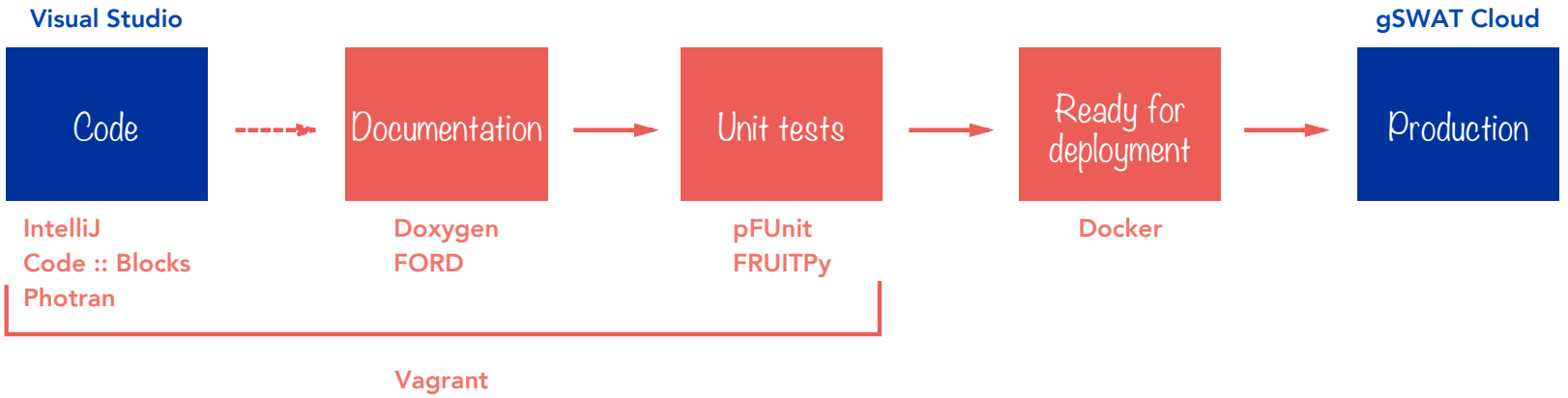
Docker File Example

```
FROM alpine:latest

MAINTAINER John Doe "johndoe@blah.com"

RUN apk add --update \
    python \
    gfortran


WORKDIR /swat
ADD requirements.txt /swat
RUN pip install -r requirements.txt
```

Outline for section 4

SmarterCities Technology Centre

Conclusion



SmarterCities Technology Centre

Appendix

1. https://sourceforge.net/projects/fortranxunit/
2. http://flibs.sourceforge.net/
3. http://objexx.com/ObjexxFTK.html
4. https://github.com/acroucher/FRUITPy
5. http://pfunit.sourceforge.net/
6. http://fortranwiki.org/fortran/show/Automatic+documentation
7. http://fortranwiki.org/fortran/show/Build+tools