

Introduction to GenApp

Emre H Brookes

4th CCP-SAS Project Meeting

Cardiff, Wales, UK
21 June 2017



Prerequisites

- Participants should bring a laptop with an SSH client and modern web browser installed.
- Working knowledge of some text editor under linux is required, such as nano, vi or emacs.
- Network access: Should be available via Wi-Fi

What we hope to accomplish today

- A bit of background on GenApp
- Create a science gateway with GenApp
 - Gain familiarity with JSON

GenApp

- Given an existing executable “module”:
 - Create a definition file describing inputs and outputs
 - Wrap or modify that executable to accept inputs and outputs as defined
 - Run the GenApp “compiler”
 - →

→ A fully functional Science Gateway

- Users, user management and statistics
- Job management, reattach
- “Cloud” file system
- Optional messaging for “live” updates
- Caching
- Multiple execution models
 - local to remote HPC and cloud
 - Airavata integration
- Can also simultaneously create “GUI” applications over the same modules
 - e.g. QT, JAVA
- Extensible!
 - features added on an as-needed basis

GenApp Apps

In production:

SASSIE-web (J.E. Curtis)
SCT (S. Perkins)

Denfert (J. Perez)
Will it Fit (L. Arleth)
US-SOMO (E. Brookes)
Vortex Shedding (A. Perlstein)
NAMDrunner (A. Savelyev)
QuaFit (F. Spinozzi)

<https://sassie-web.chem.utk.edu/sassie2>

300+ users, 9500+ jobs in 2016, 40+ papers

<http://genapp.rocks/denfert>
<http://genapp.rocks/willitfit>
<http://genapp.rocks/somo>
<http://genapp.rocks/vortexshedding>
<http://genapp.rocks/namrunner>
<http://genapp.rocks/quafit>

In Development:

BioMolAnalysis Suite (A. Savelyev)

Further out:

Bunch (Trewella)
GenFit (F. Spinozzi)
Memprot (J. Perez)



Engineering and Physical Sciences
Research Council

Demo websites

Section 0 – Access

- You should have a paper with a username and password
- ssh *username@demo.genapp.rocks*
 - if DNS is not updated, *username@129.114.17.9*
 - if you are logging in from a linux machine, you can use
 - ssh -Y *username@demo.genapp.rocks*
 - this should allow you to also test the qt4 generated applications
- Enter the password
 - Zeros have a dot in them, the capital letter O does not

Section 1.1 – The demo directory

- there is a directory under your directory named your user name
- `cd username`
 - this is the “base directory” for the application named “*username*”
- Look around and see what .json files are present
 - `ls *.json`
 - `ls modules/*.json`

GenApp JSON definitions

File	Contents
directives.json	Global application definitions. e.g. application name, title, default colors, target languages
menu.json	Selection and organization of modules
appconfig.json	Definitions that can be modified after application generation. e.g. compute resources, messaging, email information
“module”.json	Module specific definitions. e.g. inputs, outputs, underlying executable, preferred resources
“language”.json	Target language assembly information

Section 1.2 – Generate an application

- genapp
 - The executable is in your “PATH”
 - needs to be run from the “base” application directory
 - i.e. the one with directives.json

Section 1.3 – Run the application

- point your web browser to
 - `http://demo.genapp.rocks/username`
- If you have a linux system
 - from the command line
 - `cd output/qt4 && username/username`
 - If you try and run from `output/qt4/username` it won't work as it is looking for some style sheets

JSON

Example 1:

```
{  
    "name" : "value",  
    "othername" : "othervalue",  
    "almostpi" : 3.14159  
}
```

Example 2:

```
{  
    "vectorname" : [ 1, 2, 4, "five"]  
}
```

Example 3:

```
{  
    "arbitrarily-nested" :  
        [ { "this" : [ 1, 2, 4, "five"] } ,  
          "that" ]  
}
```

JSON

Example 1:

```
{  
    "name"          : "value",  
    "othername"     : "othervalue",  
    "almostpi"      : 3.14159  
}
```

- N.B.
 - Trailing commas not generally allowed
 - Use `check_json.pl` *filenames* to verify after editing
 - It is in your path
 - We allow comments beginning a line with # in all json files **except** appconfig.json

JSON

- <http://json.org/>
- Nice list of JSON libraries for various languages
- ABAP, ActionScript, Ada, AdvPL, ASP, AWK, Bash, BlitzMax, C, C++, C#, Ciao, Clojure, Cobol, ColdFusion, D, Dart, Delphi, E, Fantom, FileMaker, Fortran, Go, Groovy, Haskell, Java, JavaScript, LabVIEW, Lisp, LiveCode, LotusScript, LPC, Grimoire, LPCJSON.Lua, M, Matlab, Net.Data, ObjectiveC, OCaml, PascalScript, Perl, Photoshop, PHP, PicoLisp, Pike, PL/SQL, pljson, PowerShell, Puredata, Python, R, Racket, Rebol, RPG, Rust, Ruby, Scheme, Squeak, Symbian, Tcl, VisualBasic, VisualFoxPro
- You will need some JSON library to map your executable's existing inputs and outputs to those defined in the module file
- Examples for Perl, Python available on the wiki
<http://genapp.rocks/wiki>

Section 1.4 – Modify the module

- edit modules/penergy.json and change one of the labels
 - Editors available:
 - emacs, vi, pico, nano
- N.B. run check_json.pl on the modified json to verify you didn't make a syntax error!
- When I say “regen”
 - this means change to the application's “base” directory and run “genapp”
- take a look at the website
 - Web browsers will cache, so you may need to clear the cache

Clearing the browsers cache

- Control+Shift R or Control+Shift (left) click Reload
- OSX
 - Chrome, Firefox: Command+Shift R
 - Safari: Command+Shift click Reload
- Install a plugin
- Do a web search

Section 1.5 – Change the name

- edit directives.json
- change the “title”
- “genapp”
- clear you browser's cache
- take a look

Section 1.6 – A new module

- edit menu.json
- add a module “energy”
- “genapp”
- clear browser's cache
- take a look

Section 1.7 – change the submit policy for the new module

- edit modules/energy.json
- uncomment the “submitpolicy” line
- “genapp”
- try to run it again

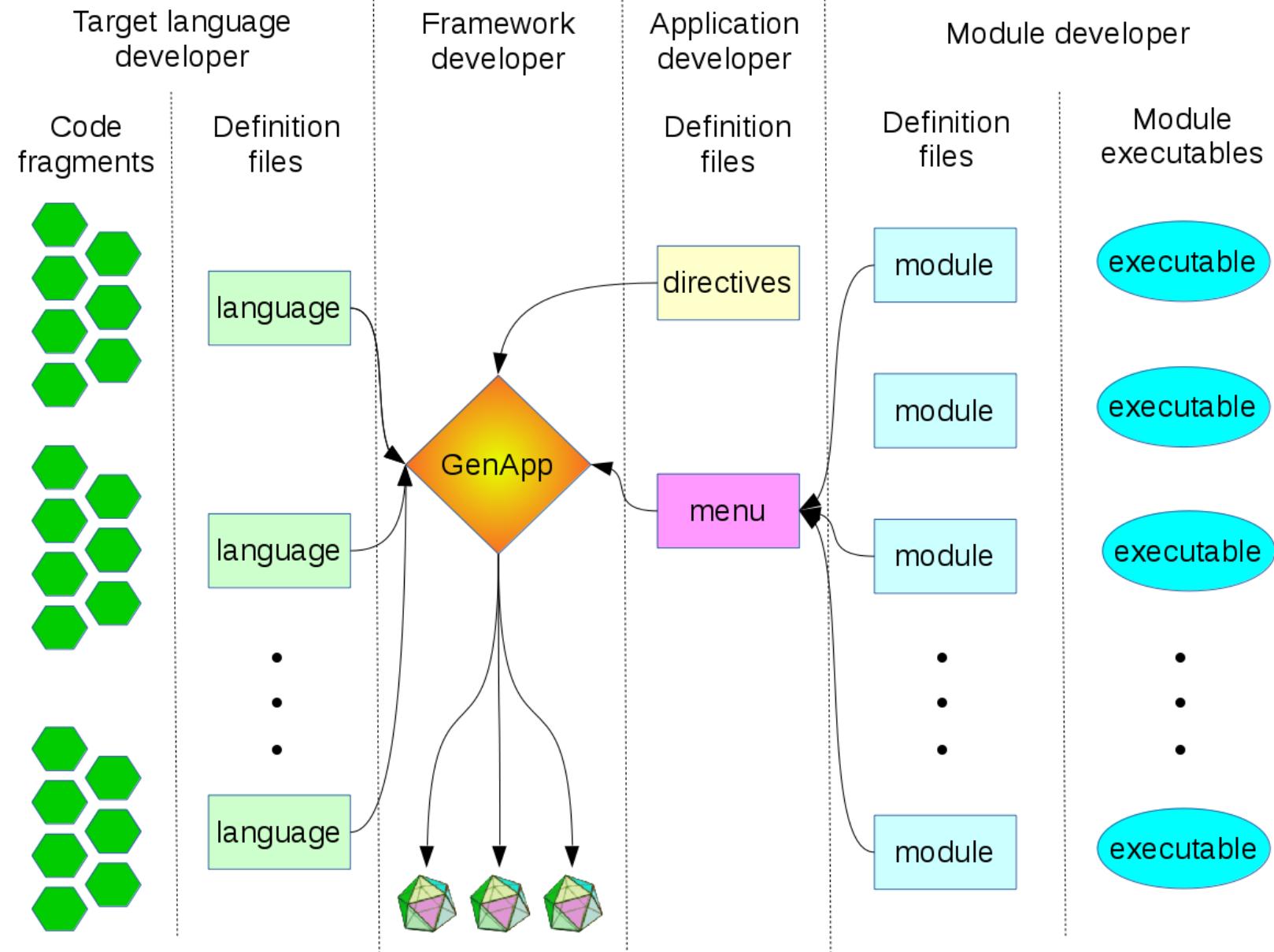
Section 1.8 – turn on splash screen / user colors

- edit directives.json
- uncomment the “usesplash” line
- add a tag “usercolors”:”true”
- “genapp”
- clear browser's cache
- try to run it again

Section 1.9 – technical details

Computer Scientists
maintaining the framework

Researchers
with code to deploy



directives.json

```
{  
    "title" : "genapptest",  
    "application" : "genapptest",  
    "version" : "2.0",  
    "languages" : [ "html5", "qt3", "qt4", "qt5", "java" ],  
    "footer" : "genapp environment",  
    "footersize" : "2.5rem",  
  
    "executable_path" :  
    {  
        "html5" : "/share/apps/genapp/genapptest/bin",  
        "qt3" : "/home/ehb/genapptest/bin",  
        "qt4" : "/home/ehb/genapptest/bin",  
        "qt5" : "/home/ehb/genapptest/bin",  
        "java" : "/home/ehb/genapptest/bin",  
    },  
  
    "docroot" :  
    {  
        "html5" : "/share/apps/genapp",  
        "qt3" : "/tmp/qt3",  
        "qt4" : "/tmp/qt4",  
        "qt5" : "/tmp/qt5",  
        "java" : "/tmp/java",  
    },  
  
    "helper" :  
    {  
        "txt" : "emacs",  
        "pdb" : "rasmol"  
    },  
  
    "appconfig" : "/home/ehb/genapptest/appconfig.json"  
}
```

```
"text_color_rgb" : "220,210,210",  
"error_color_rgb" : "255,0,0",  
"background_color_rgb" : "0,0,95",  
"select_color_rgb" : "200,128,0",  
"button_color_rgb" : "228,255,250",  
"button_color_hex" : "#E4FFFA",  
"button_g_color_rgb" : "128,170,150",  
"button_g_color_hex" : "#80AA06",  
"button_hover_color_rgb" : "255,255,255",  
"button_hover_color_hex" : "#FFFFFF",  
"button_hover_g_color_rgb" : "228,255,250",  
"button_hover_g_color_hex" : "#E4FFFA",  
"header1_color" : "228,255,250",  
"header2_color" : "228,255,250",  
"header3_color" : "228,255,250",  
"header4_color" : "228,255,250",  
"help_background_color_rgb" : "0,0,75",  
"help_text_color_rgb" : "240,240,210"  
}
```

menu.json

```
# this is a project menu file
{
  "menu" : [
    {
      "id"      : "simulate",
      "label"   : "Simulate",
      "icon"    : "pngs/simulate.png",
      "modules": [
        {
          "id"      : "penergy",
          "label"   : "Energy"
        }
      ]
    }
  ]
}
```

“module”.json : energy.json

```
{  
    "moduleId"      : "energy"  
    , "label"       : "Energy"  
    , "help"        : "help for Energy"  
    , "executable"  : "energy"  
    , "fields"      : [  
        {  
            "role"        : "input"  
            , "id"         : "m"  
            , "label"      : "mass [kg]"  
            , "type"       : "float"  
            , "required"   : "true"  
            , "help"        : "Enter the mass in kilograms"  
        }  
        , {  
            "role"        : "input"  
            , "id"         : "c"  
            , "label"      : "Speed of light [m/s]"  
            , "type"       : "float"  
            , "default"    : 299792458  
            , "required"   : "true"  
            , "help"        : "Enter the speed of light in meters/second"  
        }  
        , {  
            "role"        : "output"  
            , "id"         : "e"  
            , "label"      : "Energy [J]"  
            , "type"       : "text"  
        }  
    ]  
}
```

Module definition

```
{  
  "moduleid" : "center",  
  "label" : "Center",  
  "executable" : "center",  
  "submitpolicy" : "all",  
  "fields" : [  
    {  
      "role" : "input",  
      "id" : "input1",  
      "label" : "Center value 1",  
      "type" : "float",  
      "default" : 0.0,  
      "min" : 0.0,  
      "max" : 1e50,  
      "step" : 0.01,  
      "required" : "true"  
    },  
    {  
      "role" : "input",  
      "id" : "runminutes",  
      "label" : "Minutes to run",  
      "type" : "float",  
      "default" : 1,  
      "min" : 0,  
      "max" : 10000,  
      "required" : "true"  
    },  
    {  
      "role" : "input",  
      "id" : "usemessage",  
      "label" : "Message results",  
      "type" : "checkbox",  
      "checked" : "true",  
      "repeater" : "true"  
    },  
    {  
      "role" : "input",  
      "id" : "messagesecs",  
      "label" : "Results interval in seconds",  
      "type" : "integer",  
      "default" : 15,  
      "min" : 1,  
      "max" : 1000,  
      "repeat" : "usemessage"  
    },  
  ]  
}
```

Executable Input

```
{  
  "input1":"0",  
  "runminutes":1,  
  "usemessage":on,  
  "messagesecs":15,  
  "_logon":"emre",  
  "_project": "",  
  "_window":"d67fa871-46ea-4af3-d384-",  
  "_uuid":"197f3ea0-a2b4-11e4-ae42-518",  
  "_base_directory":"\\vshare\\apps\\gena",  
  "_udphost":"160.36.200.43",  
  "_udpport":37779,  
  "resourcedefault":local  
}
```

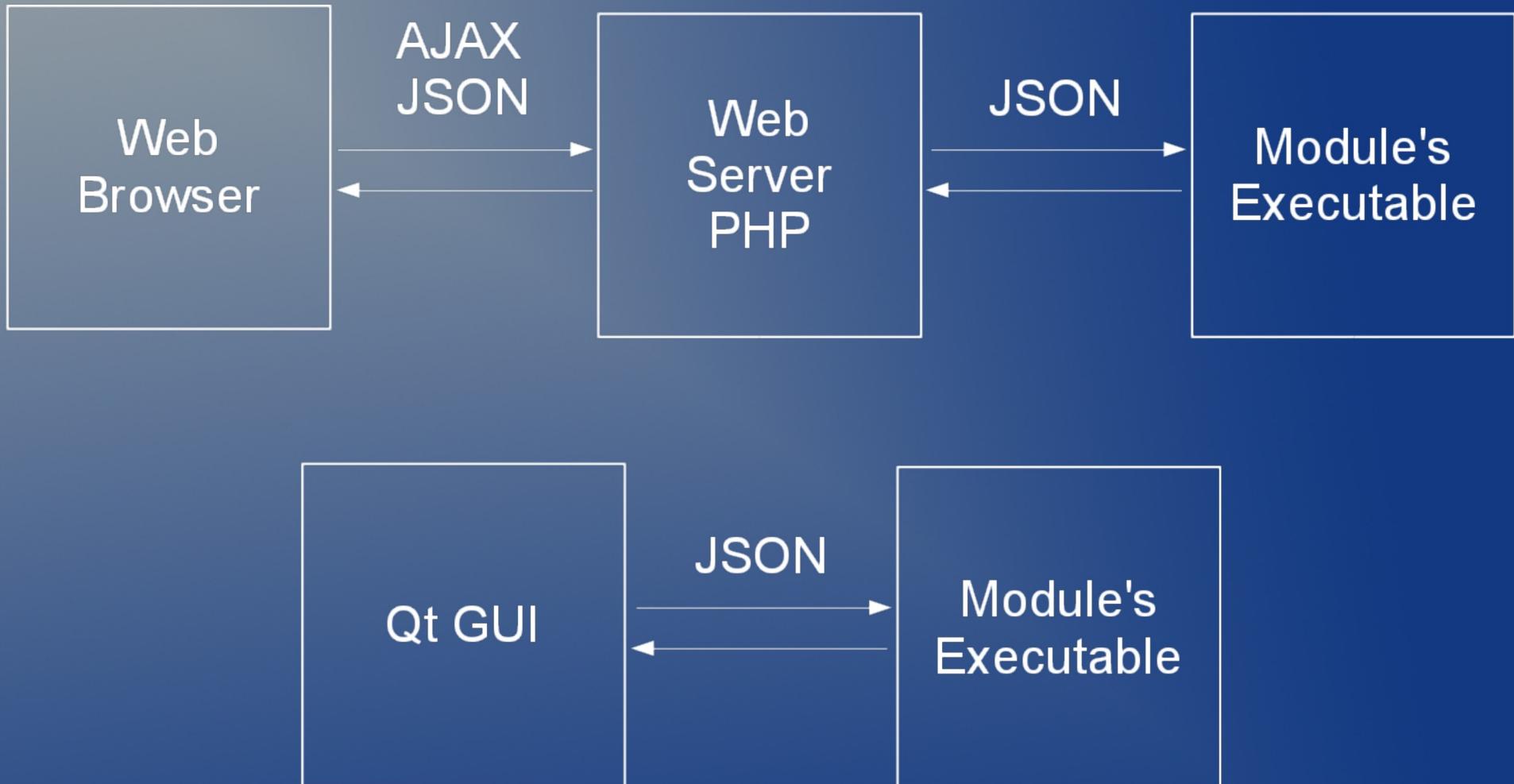
Executable Output

```
{  
  "udphost":"160.36.200.43",  
  "output1":"Fully completed 4 runs of sleep 15s",  
  "_message":{  
    "icon":"information.png",  
    "text":"test message"  
  },  
  "output3":1,  
  "hi":"center executable",  
  "udpport":37779  
}
```

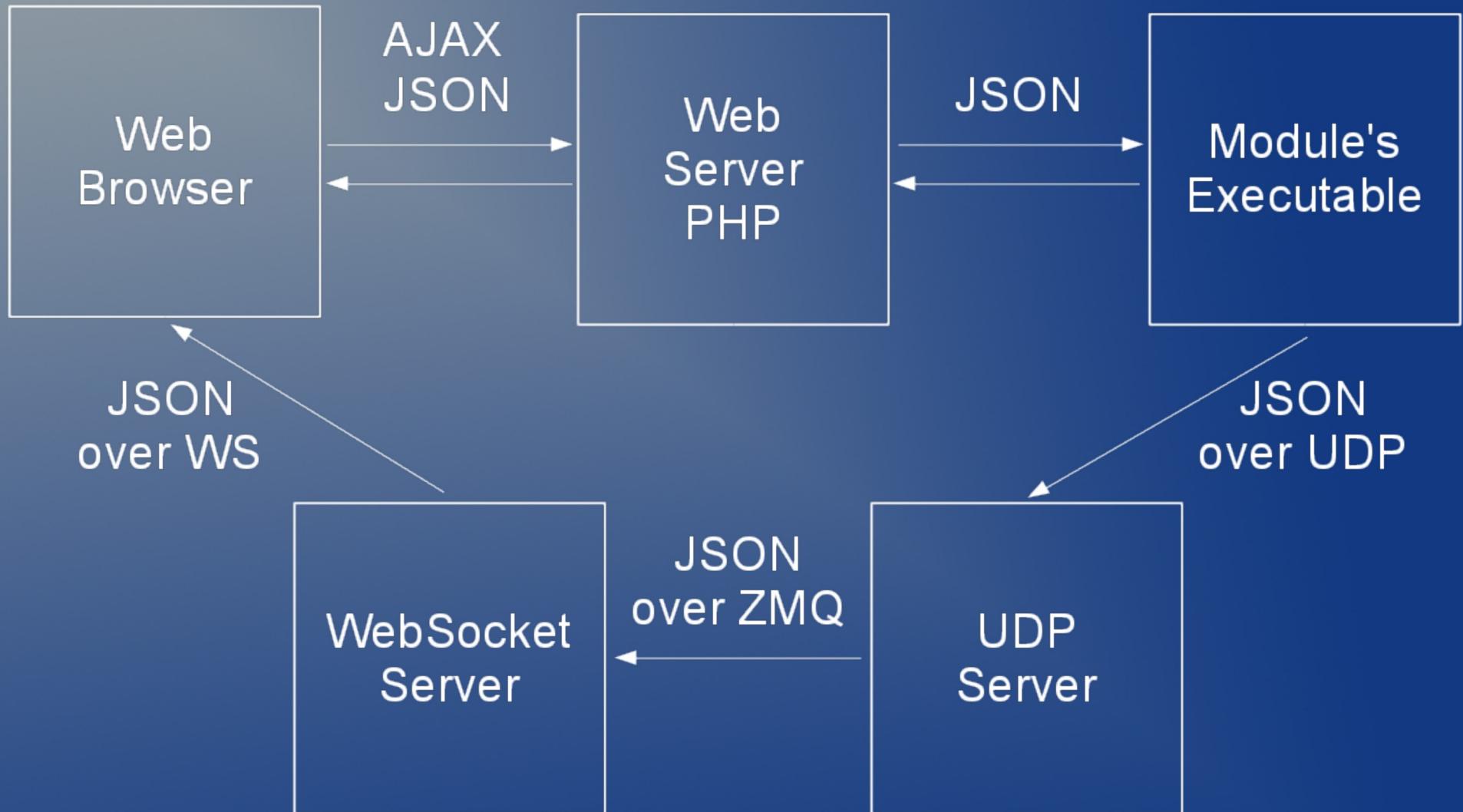
Module fields

name	input support	output support	append (output-only primarily for messaging)	default (input-only)	min (input-only)	max (input-only)	step (input-only)	required (input-only)	repeater (input-only)	help (example calculate:calculate_2, analysis:pm)	notes	genapptest example
integer	yes	yes	yes	no	yes	yes	html5-only	yes	html5-only	html5-only		tools:center repeater example interact:interact_1 step example tools:center
float	yes	yes	no	yes	yes	yes	html5-only	yes	no	html5-only		tools:center
text	yes	yes	no	yes	no	no	no	yes	no	html5-only		build:build_1
textarea	yes	yes	html5-only	yes	no	no	no	yes	no	html5-only	multiline default input not yet supported	build:build_2
checkbox	yes	yes	no	no	no	no	no	yes	html5-only	html5-only	additional 'checked' attribute	build:build_2 repeater example calculate:calculate_1
radio	yes	yes	no	no	no	no	no	yes	no	html5-only	one entry for each button name is the name of the radio button (common for all members)	build:build_2
listbox	yes	no	no	yes	no	no	no	no	html5-only	html5-only	no required attribute, but it will always have a value row spacing minor layout issue on repeater	build:build_2 repeater example calculate:calculate_2
plot2d	no	yes	no	no	no	no	no	no	no	html5-only	more info	tools:data_interpolation
atomicstructure	no	yes	no	no	no	no	no	no	no	html5-only	more info	tools:data_interpolation
file	yes	yes	no	no	no	no	no	no (always required in html5)	no	html5-only	additional multiple attribute(input-only)	simulate:simulate_1
label	html5-only	html5-only	no	yes*	no	no	no	no	no	html5-only	*can be header1,2,3,4 or blank	interact:interact_2
hidden	html5-only	no	no	yes	no	no	no	no	no	no	this is primarily to pass a value in json to the executable	analysis:pm
html	no	html5-only	no	yes	no	no	no	no	no	no	allows html output	none currently, but it's a typical field layout
progress	no	html5-only	no	no	no	yes (for output)	no	no	no	no	progress bar default ranges is 0 to 1.0, setting max to a positive value will make the range 0-max	tools:center

Execution model HTML5/PHP vs Qt/C++

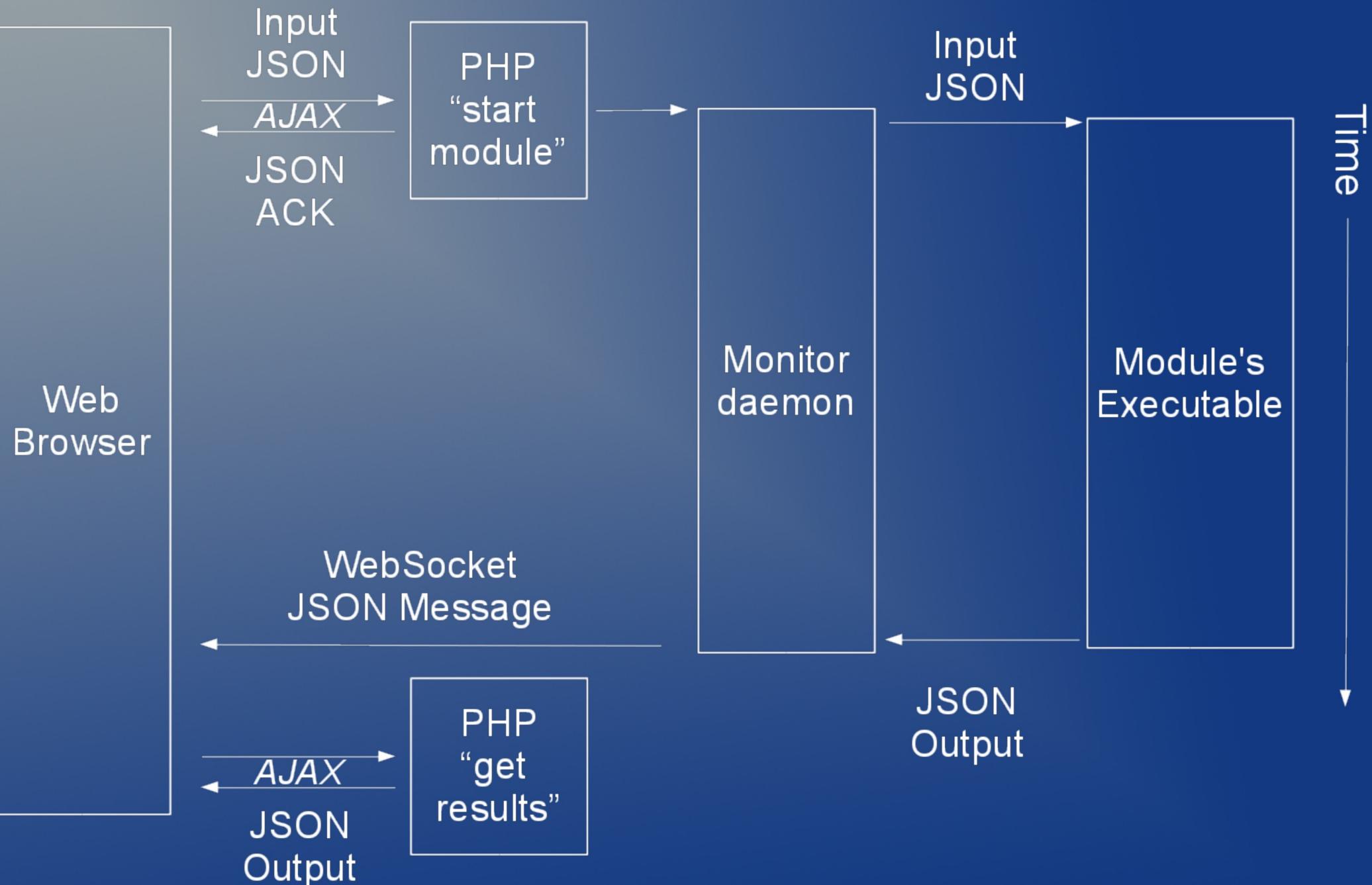


HTML5/PHP messaging



Examples on wiki “docs → udp messaging for html5/php”

Enhanced execution model HTML5/PHP

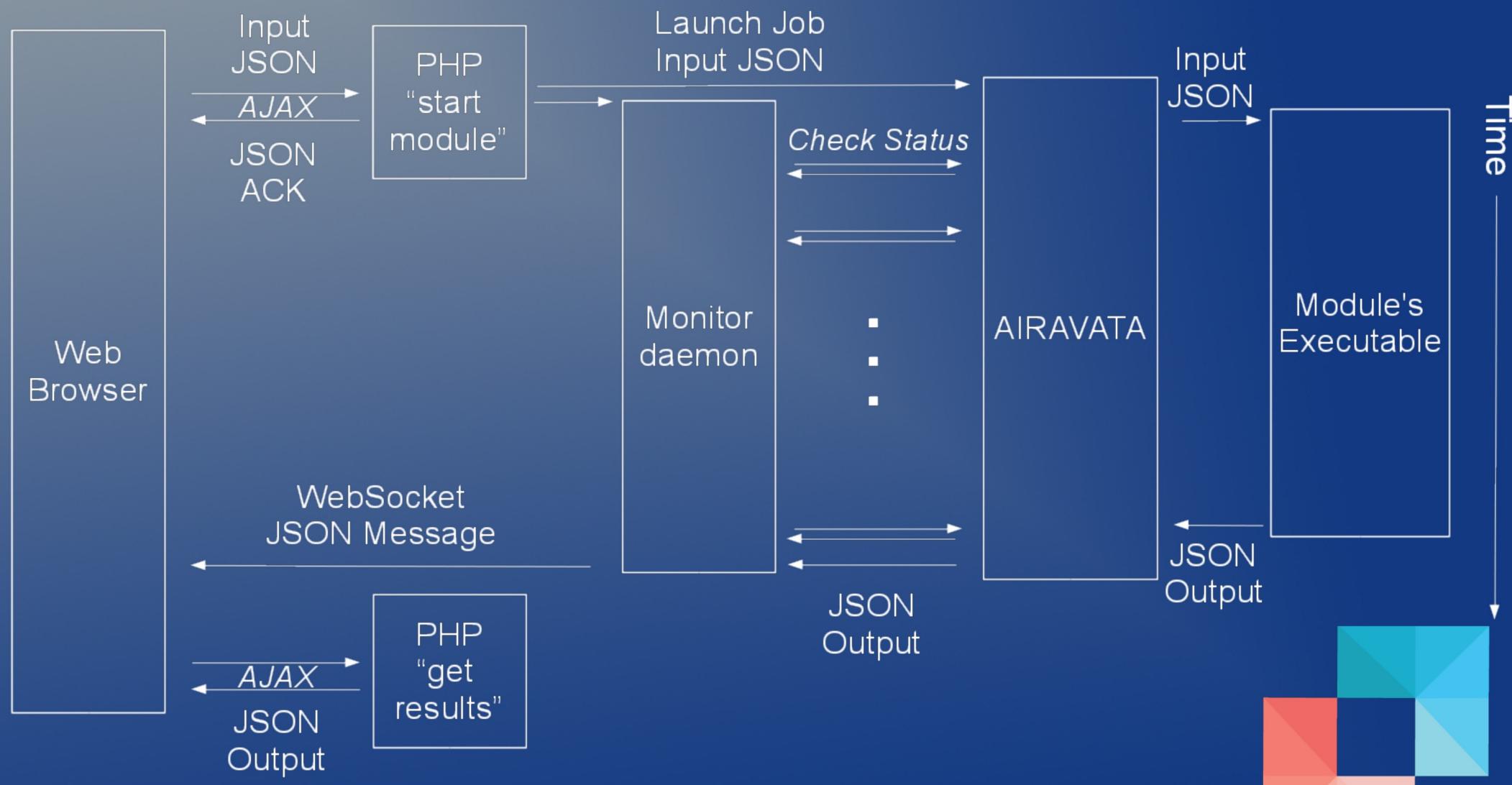




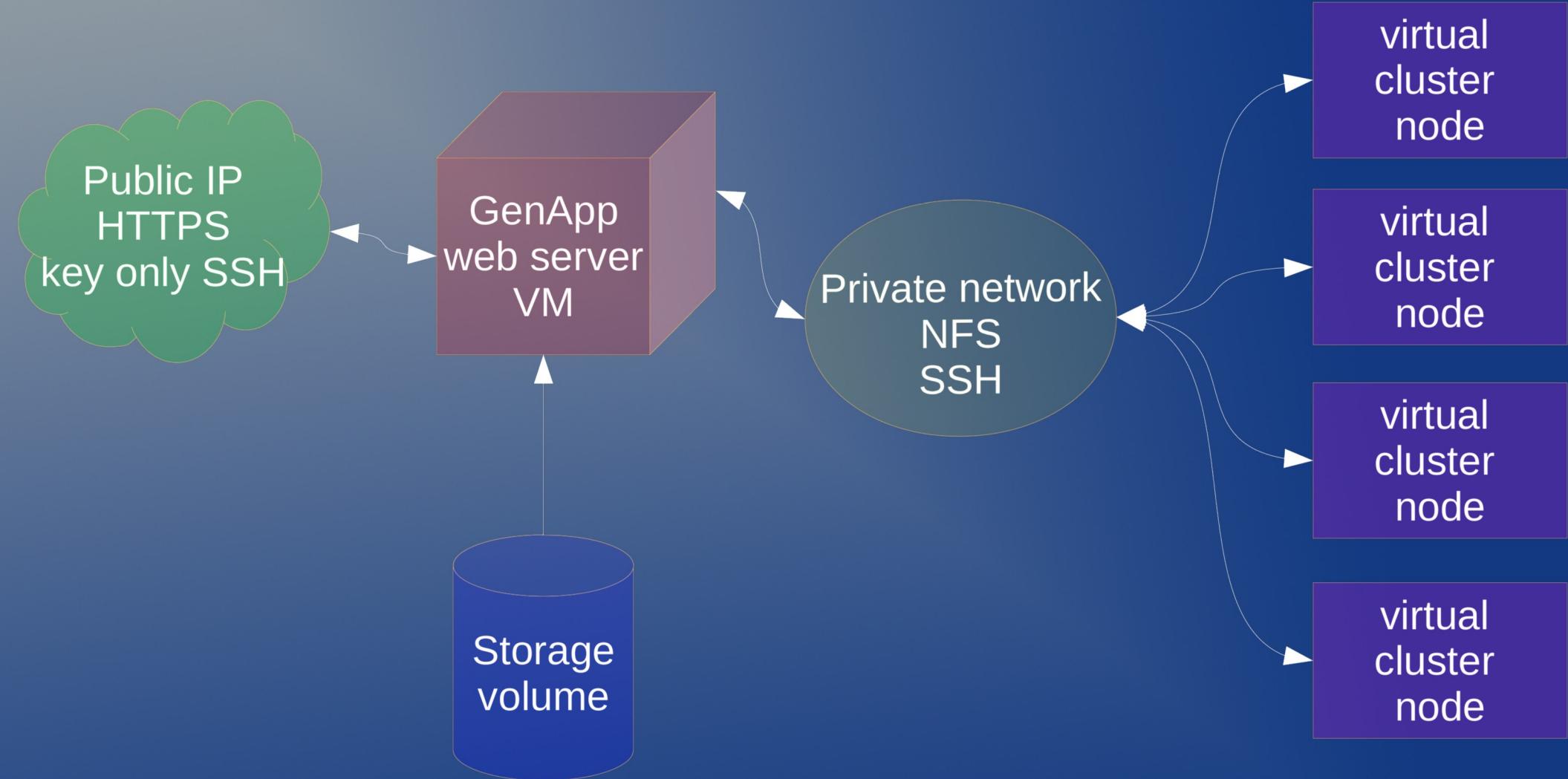
GSoC 2014

GenApp Enhanced execution model HTML5/PHP with Apache Airavata

Nadeem Anjum / Mentor: Suresh Marru

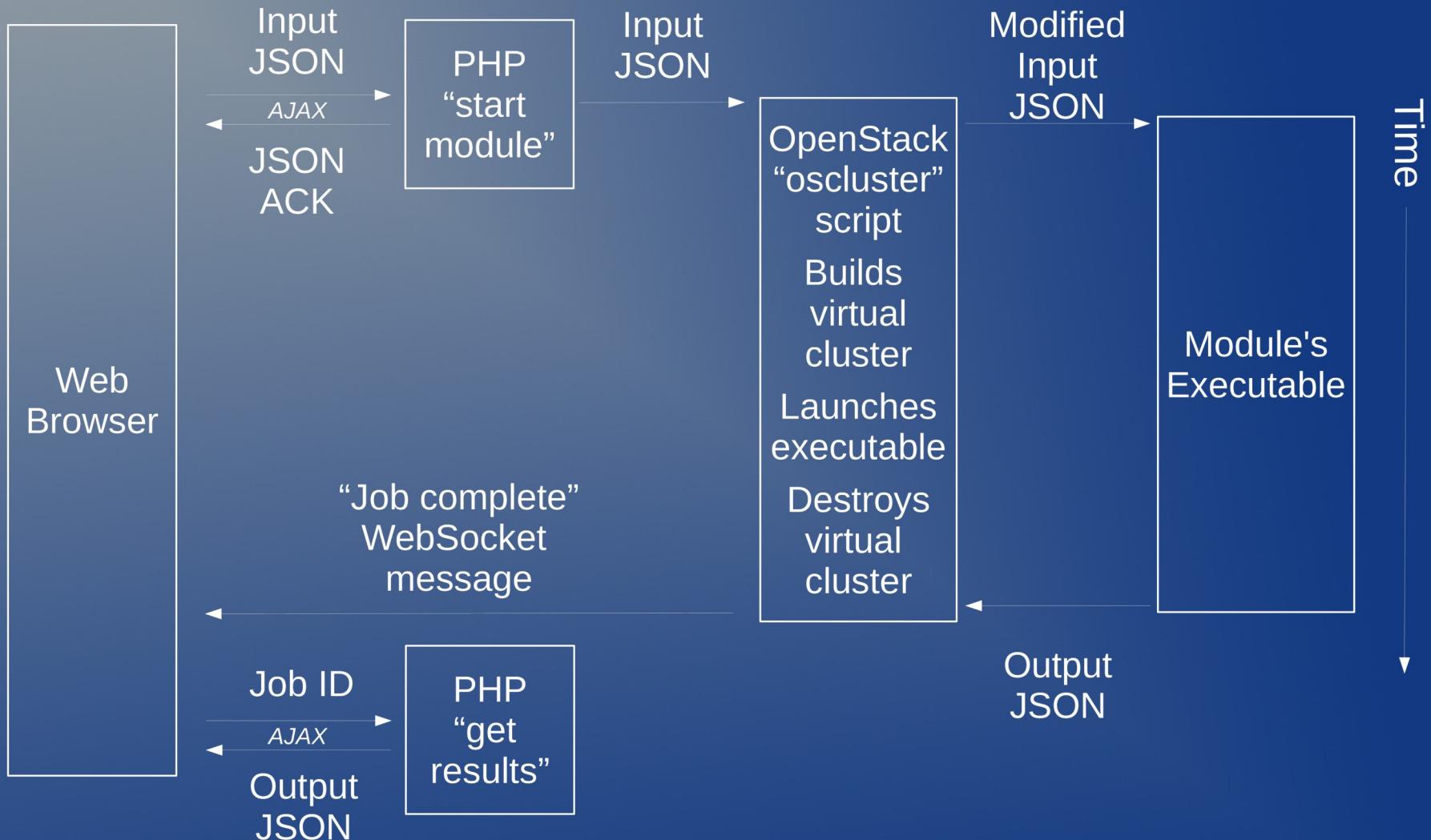


GenApp Integrated with OpenStack Supports Elastic Computing on Jetstream



Emre Brookes and Alexey Savelyev. (2017). PEARC17, ACM.

GenApp Integrated with OpenStack Supports Elastic Computing on Jetstream



Emre Brookes and Alexey Savelyev. (2017). PEARC17, ACM.

Section 1.10 resources

- genapp wiki
 - <http://genapp.rocks/wiki>

Section 2 – intermediate topics

- Dynamic content
 - Calculated fields
 - Repeaters
 - Messaging

Section 2.1 – calculated fields

- Convert energy module to calculate within the UI
- See “wiki → docs → calculated fields” for reference and try to implement

Section 2.2 – repeaters

- Checkbox
- Listbox
- Integer
- Show QuaFit & WillItFit examples
- See “wiki → advanced topics → repeaters and repeats” for reference and try to implement a repeater

Section 2.3 – messaging

- Send a udp JSON message
- Show SASSIE MMC example
- See “wiki → docs → udp messaging for html5/php” for reference and try to implement a message

Section 3 – Advanced topic – fragments to code

- Important for target language and framework developers
- see “wiki → Technical developer notes → Fragments to code”
- create a new target language
 - N.B. a GSoC student created the basic JAVA target language in 1 week

In closing

- GenApp produces working science gateways & local GUI apps
- Easily extensible
- Advancements are requirements driven
- So let us know your requirements!

Special Thanks

- The CCP-SAS Team ...

Thanks for attending

*Questions: ask now or
email me at emre@biochem.uthscsa.edu*

Acknowledgments

- Joseph Curtis
- To E. Brookes
 - NIH NIGMS K25GM090154 / PI
 - NSF CHE-1265817 / PI
- NSF XSEDE TG-MCB140255
- UTHSCSA BCF
- To N. Anjum, A. Kapoor, P. Patra
 - Google Summer of Code

Funded by a grant from the National Institute of General Medical Sciences of the National Institutes of Health



Pubs

- Brookes, E H, 2014, *An Open Extensible Multi-Target Application Generation Tool for Simple Rapid Deployment of Multi-Scale Scientific Codes*, ACM, 10.1145/2616498.2616560
- Brookes, E H, Anjum, N, Curtis, J E, Marru, S, Singh, R, Pierce, M, 2014, *GenApp module execution and airavata integration*, IEEE Press, 10.1109/GCE.2014.12
- Brookes, E H, Anjum, N, Curtis, J E, Marru, S, Singh, R, Pierce, M, 2015, *The GenApp framework integrated with Airavata for managed compute resource submissions*, Concurrency and Computation: Practice and Experience, Wiley & Sons, 10.1002/cpe.3519
- Wright DW, Nan R, Hui G, Curtis JE, Brookes EH, Perkins SJ. CCP-SAS - Novel Approaches for the Atomistic Modeling of Small Angle Scattering Data in Biology. Biophysical journal. 2015 January 27; 108(2):191a.
- Brookes EH, Kapoor A, Patra P, Marru S, Singh R, Pierce M. GSoC 2015 student contributions to GenApp and Airavata. Concurrency and Computation: Practice and Experience. 2015 October 31 DOI: 10.1002/cpe.3689
- Emre Brookes and Alexey Savelyev. (2017). GenApp Integrated with OpenStack Supports Elastic Computing on Jetstream. PEARC17, ACM.
- Alexey Savelyev and Emre Brookes (2017) GenApp: Extensible Tool for Rapid Generation of Web and Native GUI Applications, J. Future Generation Computer Systems, (invited, submitted)
- Emre Brookes and Alexey Savelyev. (2017). Architecture and Performance of a GenApp Generated NAMD Science Gateway on Jetstream. SuperComputing, ACM (submitted)

Supplementary Slides Follow

CCP-SAS

- SASSIE <http://www.smallangles.net/sassie>

- Joseph Curtis et al.



- PYTHON

- includes wrapped binary executables

- SCT/SCTPL/HYDRO <http://www.ucl.ac.uk/smb/perkins>

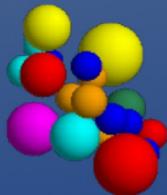
- Steve Perkins et al.



- Structural Immunology Group at University College London

- FORTRAN

- US-SOMO <http://somo.uthscsa.edu>



- Emre Brookes et al.

- C++/Qt

- includes wrapped binary executables

- attract others ...



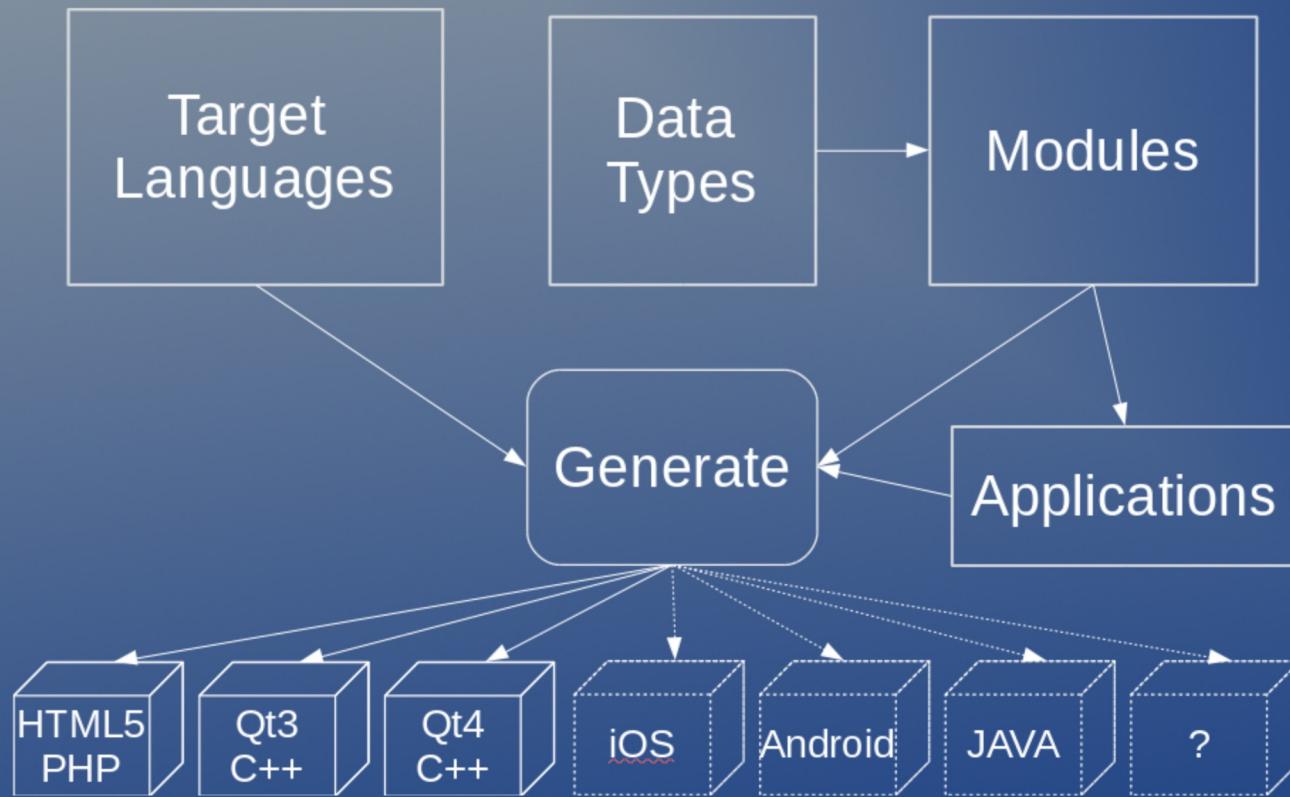
Considerations

- Ease of deployment in an ever-evolving software environment landscape
- Legacy and frequently specific lab developed codes
- Labs frequently can not afford a dedicated software team nor the cost in time and funding

“Dark code”

An Open Extensible Multi-Target Application Generation Tool for Simple Rapid Deployment of Multi-Scale Scientific Codes

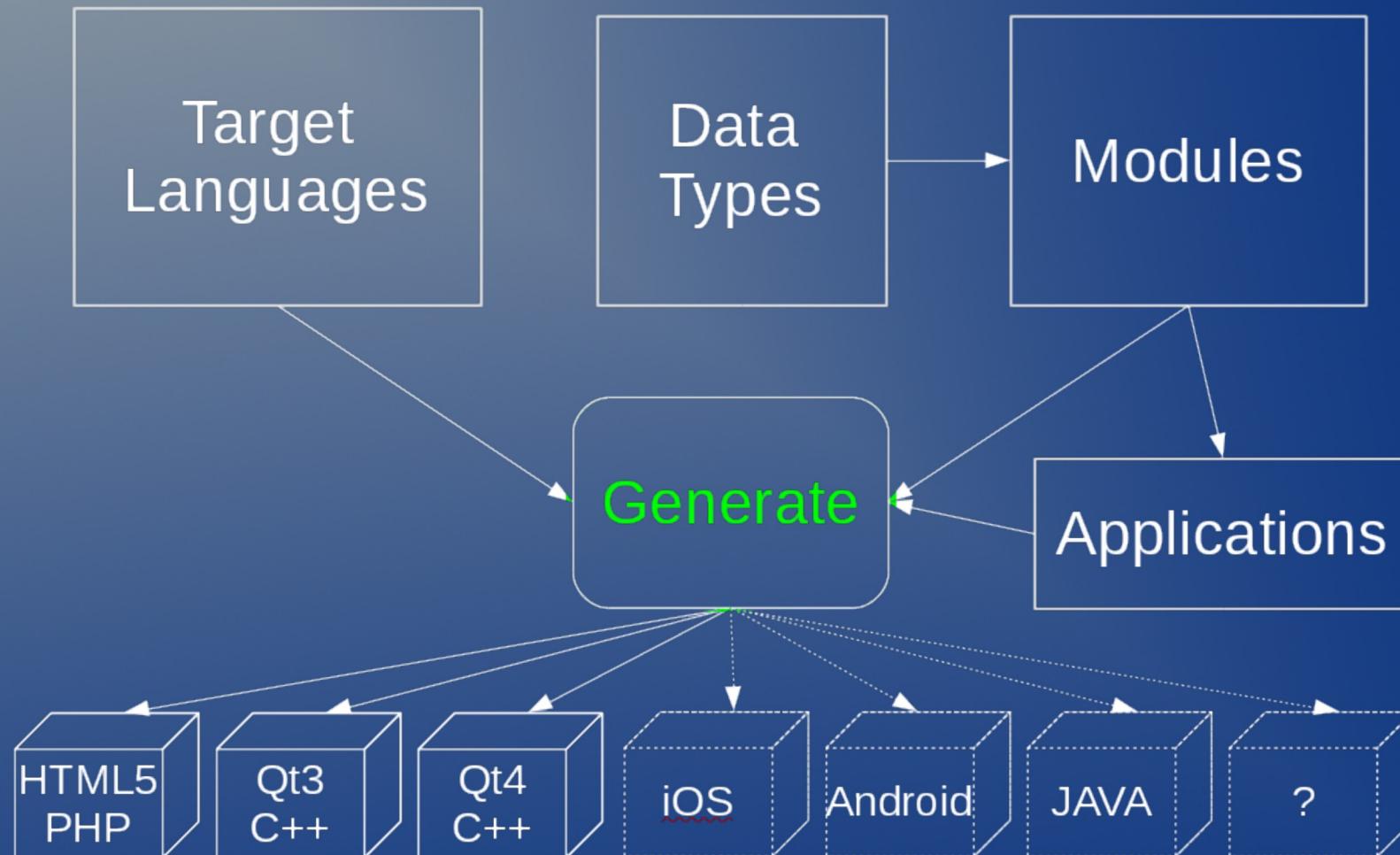
Brookes, E. H. XSEDE 14 Atlanta



- JSON
- Global directives & application menu and configuration
- Module I/O definitions
- Module messaging
- Language assembly descriptions

GenApp : What is it?

- An assembler of fragments of code directed by definition files



appconfig.json

```
{  
    "mail" : {  
        "admin"      : "admin@some-address"  
        , "feedback" : "feedbackuser@some-address"  
        , "from"      : "some-address"  
    }  
    , "hostip"     : "1.2.3.4"  
    , "hostname"   : "my.genapp.host"  
    , "messaging" : {  
        "wsport"      : 80  
        , "wssport"    : 443  
        , "zmqhostip" : "1.2.3.5"  
        , "zmqport"   : 12345  
        , "udphostip" : "1.2.3.5"  
        , "udpport"   : 12346  
    }  
    , "resources" : {  
        "local"       : ""  
        , "compute0"  : "ssh compute-0-0"  
        , "compute1"  : "ssh compute-0-1"  
    }  
    , "resourcedefault" : "local"  
    , "submitpolicy"   : "login"  
}
```

“language”.json : qt5.json

```
{  
    "id"          : "qt5",  
    "description" : "qt5 with direct calls",  
    "version"     : 1,  
    "assembly"   : [  
        {  
            "frequency"  : "once",  
            "output"      : "__application__/style.qss",  
            "inputs"       : [  
                { "style.qss" : "once" }  
            ]  
        },  
        {  
            "frequency"  : "once",  
            "output"      : "__application__/_application_.cpp",  
            "inputs"       : [  
                { "base_header.cpp" : "once" },  
                { "base_items.cpp" : "menu:id" },  
                { "base_footer.cpp" : "once" },  
                { "types/_fields:type_.extra.cpp" : "fields:id" },  
                { "base_additional.cpp" : "menu:id" },  
                { "base_additional2.cpp" : "menu:modules:id" }  
            ]  
        },  
        {  
            "frequency"  : "once",  
            "output"      : "__application__/_application_.h",  
            "inputs"       : [  
                { "base_header.h" : "once" },  
                { "base_items.h" : "menu:id" },  
                { "base_modules.h" : "menu:modules:id" },  
                { "types/_fields:type_.extra.h" : "fields:id" },  
                { "base_footer.h" : "once" }  
            ]  
        },  
        ...  
    ]  
}
```

“language”.json : qt5.json

```
{  
    "frequency" : "menu:id",  
    "output" : "__application__/_menu:id__.h",  
    "inputs" : [  
        { "menuelement_header.h" : "once" },  
        { "menuelement_items.h" : "menu:modules:id" },  
        { "menuelement_footer.h" : "once" }  
    ]  
},  
{  
    "frequency" : "menu:modules:id",  
    "output" : "__application__/_menu:id____menu:modules:id__.h",  
    "inputs" : [  
        { "types/input.header" : "once" },  
        { "types/_fields:type_.input" : "fields:id" },  
        { "types/input.buttons" : "once" },  
        { "types/_fields:type_.output" : "fields:id" },  
        { "types/input.footer" : "once" }  
    ]  
},  
{  
    "frequency" : "once",  
    "output" : "__application__/_utility_routines.cpp",  
    "inputs" : [ { "utility_routines.cpp" : "once" } ]  
},  
{  
    "frequency" : "once",  
    "output" : "__application__/_utility_routines.h",  
    "inputs" : [ { "utility_routines.h" : "once" } ]  
}  
]
```

GenApp Roles

Role	Responsibility	JSON	Expertise
Tool developer	Maintains the GenApp generator tool		Strong C.S. GenApp Framework
Framework developer	Implements and maintains target languages by building up code fragments and defining their assembly	<i>"language".json</i>	Language specific. GenApp Framework.
Application developer	Organizes modules in a menu definition file. Runs the GenApp generator to create working instances	directives.json menu.json appconfig.json	Using the GenApp Framework. Target environments. User community.
Module wrapper	Wraps executable modules by writing a module definition file and ensuring the wrapped executable accepts defined JSON input and produces defined JSON output	<i>"module".json</i>	The specific executable, where it can run and formatting inputs and outputs.

extremely basic instructions to wrap an application

- from your base directory
 - edit directives.json
 - edit menu.json
 - edit modules/*.json
 - create application executable json wrappers in bin/
 - NB: executable **must** only output valid json to stdout
 - run \$ genapp.pl
 - html output should be available via web interface
 - eg
 - ↗ <http://somo.chem.utk.edu/genpptest>
 - if you get forbidden errors in the web browser, make sure everything is group readable by user apache

Credit: Joseph Curtis – 1st User

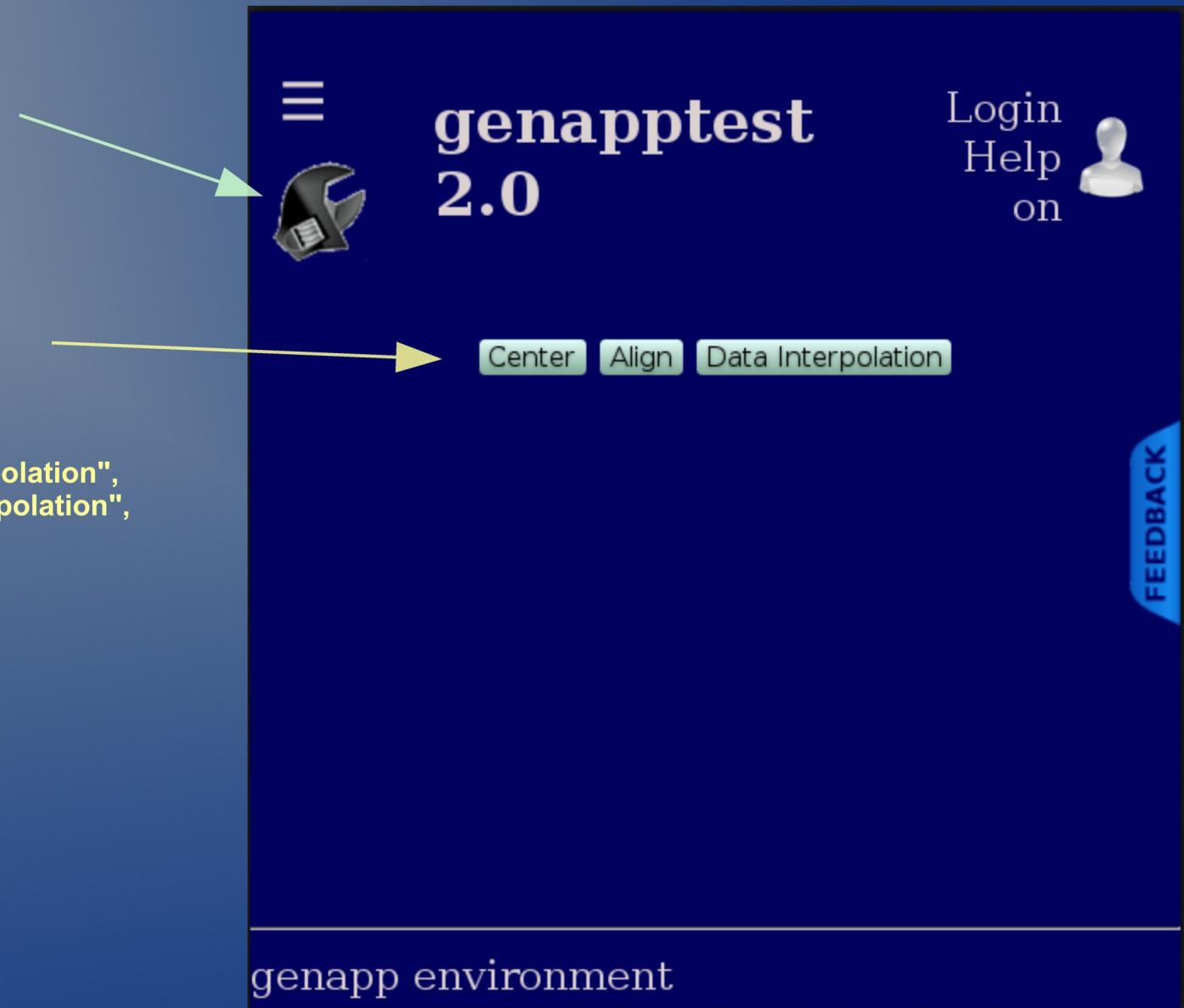
Menu Definition

```
{  
  "menu": [  
    {  
      "id" : "tools",  
      "label" : "Tools",  
      "icon" : "pngs/tools.png",  
      "modules": [  
        {  
          "id" : "center",  
          "label" : "Center",  
        },  
        {  
          "id" : "align",  
          "label" : "Align",  
        },  
        {  
          "id" : "data_interpolation",  
          "label" : "Data Interpolation",  
        }  
      ],  
      {  
        "id" : "build",  
        "label" : "Build",  
        "icon" : "pngs/build.png",  
        "modules": [  
          {  
            "id" : "build_1",  
            "label" : "Build 1",  
          },  
          {  
            "id" : "build_2",  
            "label" : "Build 2",  
          }  
        ]  
      },  
    ]  
  },  
}
```



Menu Definition

```
{  
  "menu": [  
    {  
      "id" : "tools",  
      "label" : "Tools",  
      "icon" : "pngs/tools.png",  
      "modules": [  
        {  
          "id" : "center",  
          "label" : "Center",  
        },  
        {  
          "id" : "align",  
          "label" : "Align",  
        },  
        {  
          "id" : "data_interpolation",  
          "label" : "Data Interpolation",  
        }  
      ]  
    },  
    {  
      "id" : "build",  
      "label" : "Build",  
      "icon" : "pngs/build.png",  
      "modules": [  
        {  
          "id" : "build_1",  
          "label" : "Build 1",  
        },  
        {  
          "id" : "build_2",  
          "label" : "Build 2",  
        }  
      ]  
    },  
  ],  
}
```





genapptest 2.0

Logoff emre Help on

Center Align Data Interpolation

Center value 1

Minutes to run

Message results

Results interval in seconds

FEEDBACK

Message text

Output value 1

Progress

genapp environment

Menu Access

Current
Top Level
Menu Icon

File Management

Register
User Management



genapptest 2.0

Logoff emre
Help on

Submenu Choices

Center Align Data Interpolation

Input Area

Center value 1

Minutes to run

Message results

Results interval in seconds

Submit

Output Area

Message text

Output value 1

Progress

Optional Footer

genapp environment

Job Management

FEEDBACK

Feedback
Tab

HTML5/PHP and Qt/C++ versions side-by-side

genapptest 2.0

Login Help off

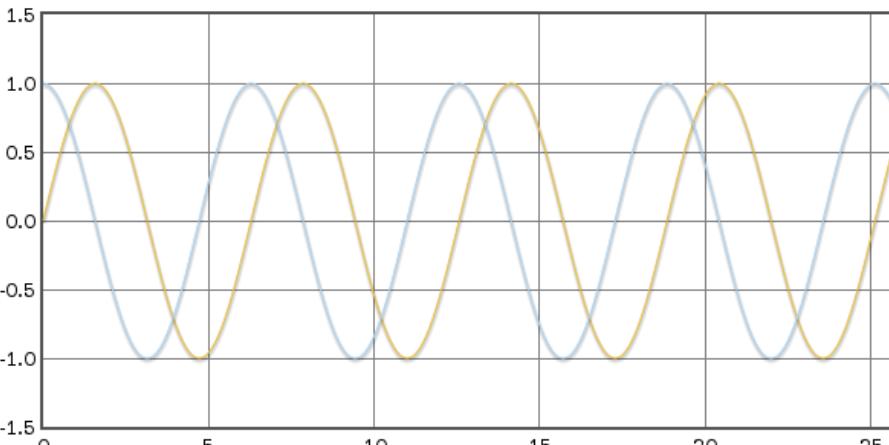
Center Align Data Interpolation

Start x
End x
Delta x
abs

Submit Reset to default values

output integer
output float
output text

plot 2d



input json reference [logref.txt](#)
results json reference [logres.txt](#)
test pdb output [testpdb.pdb](#)

genapptest 2.0

genapptest 2.0

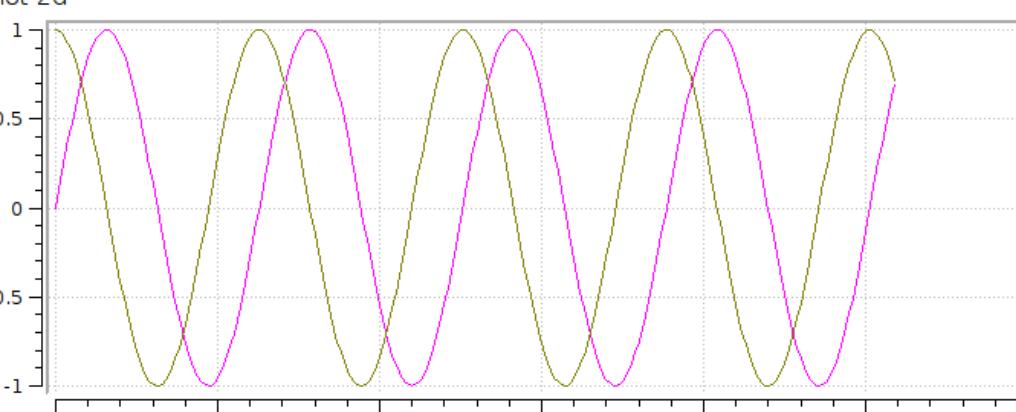
Center Align Data Interpolation

Start x
End x
Delta x
abs

Submit Reset to default values

output integer
output float
output text

plot 2d



input json reference [/tmp/genapptest/20140323021456264/logref.txt](#)
results json reference [/tmp/genapptest/20140323021456264/logres.txt](#)
test pdb output [/tmp/genapptest/20140323021456264/testpdb.pdb](#)



Center Align Data Interpolation

Start x
End x
Delta x
abs

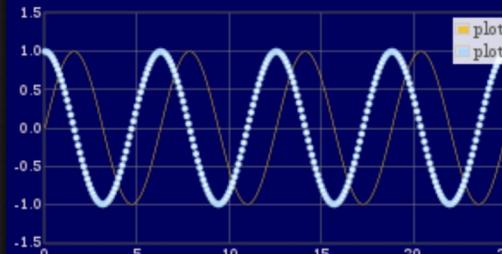
Unexpected results:

hi => data_interpolation executable

p => results/users/emre/no_project_specified

output integer
output float
output text

plot 2d



input json reference [logref.txt](#)

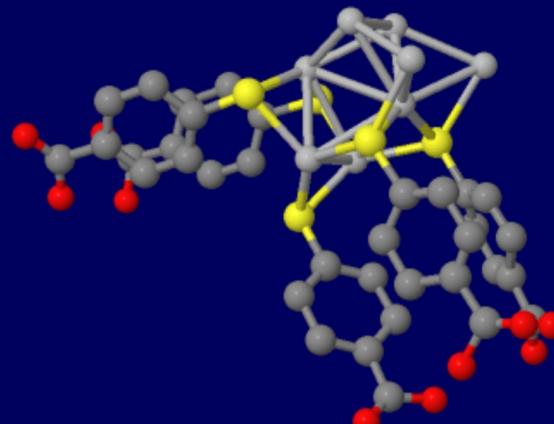
results json reference [logres.txt](#)

test pdb output

Molecular Viewer

HTML5:JSMol

Qt/C++: Multiple options (RASMOL, Jmol, others)



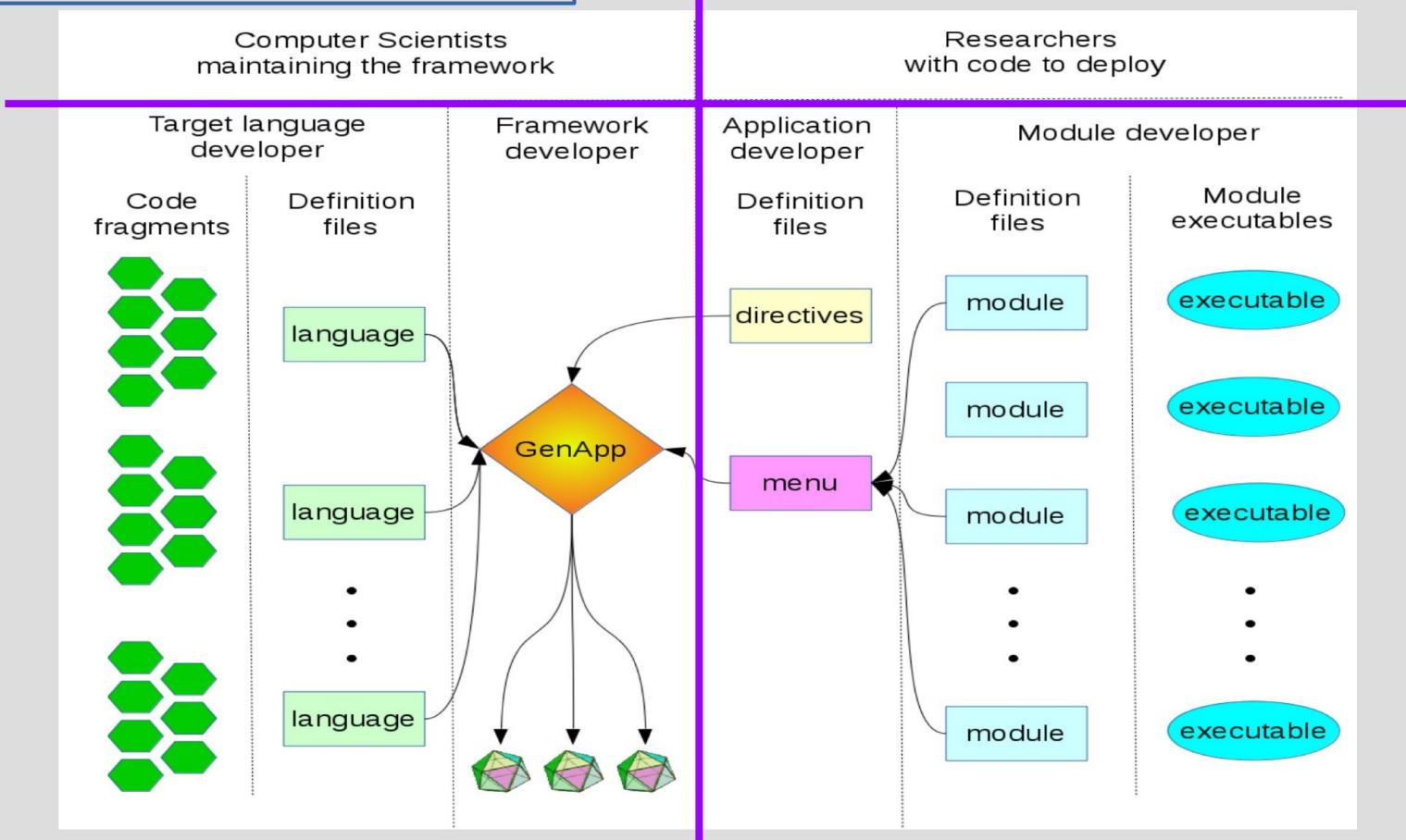
Encountering GenApp: Extending Horizon

Cool stuff:

- Modifying fragments of code
- Adding new /Extending types
- Conditional code generation
- Browser compatibility
- Admin utilities
-

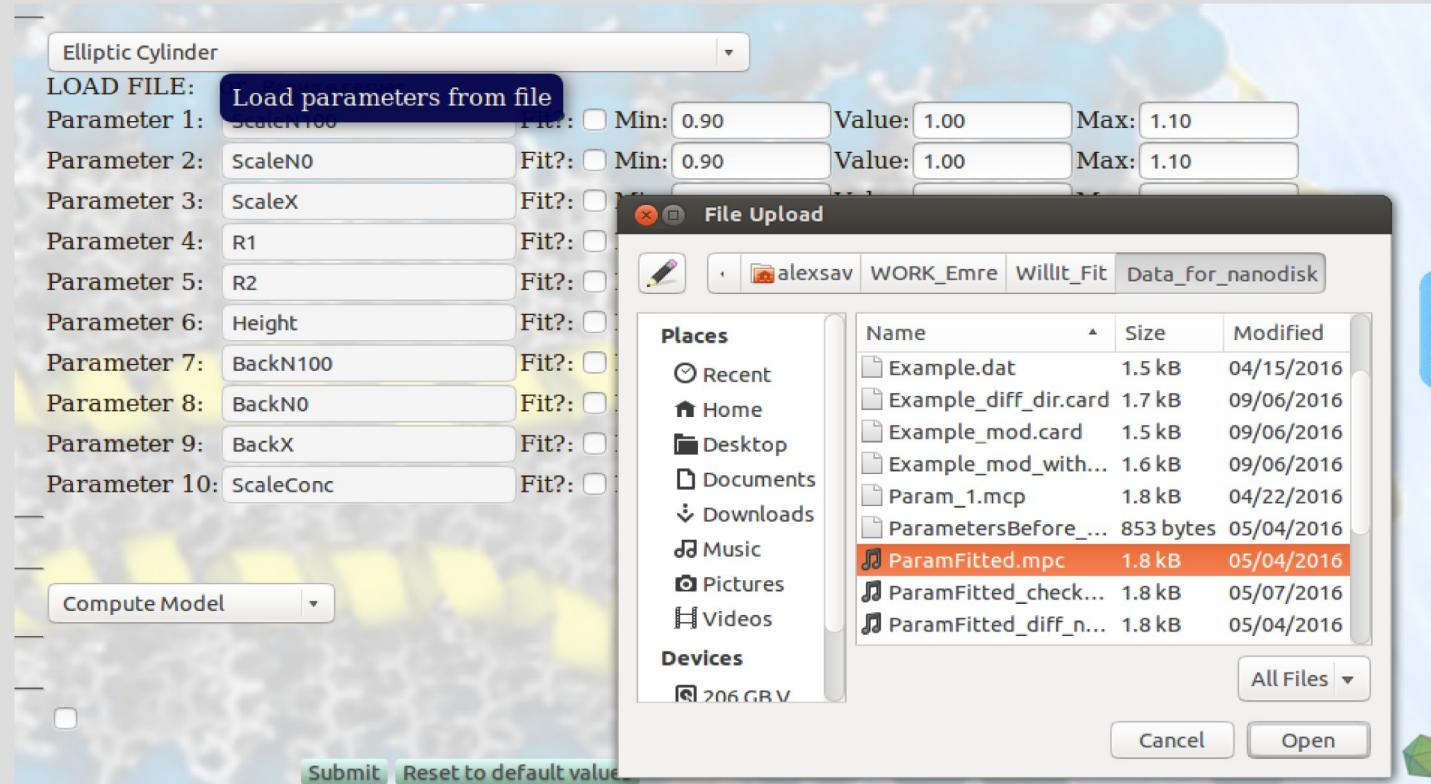


Web Development:
HTML5
CSS
JavaScript (jQuery)
PHP
MongoDB



Advanced “GenApping”: *WillItFit* (1)

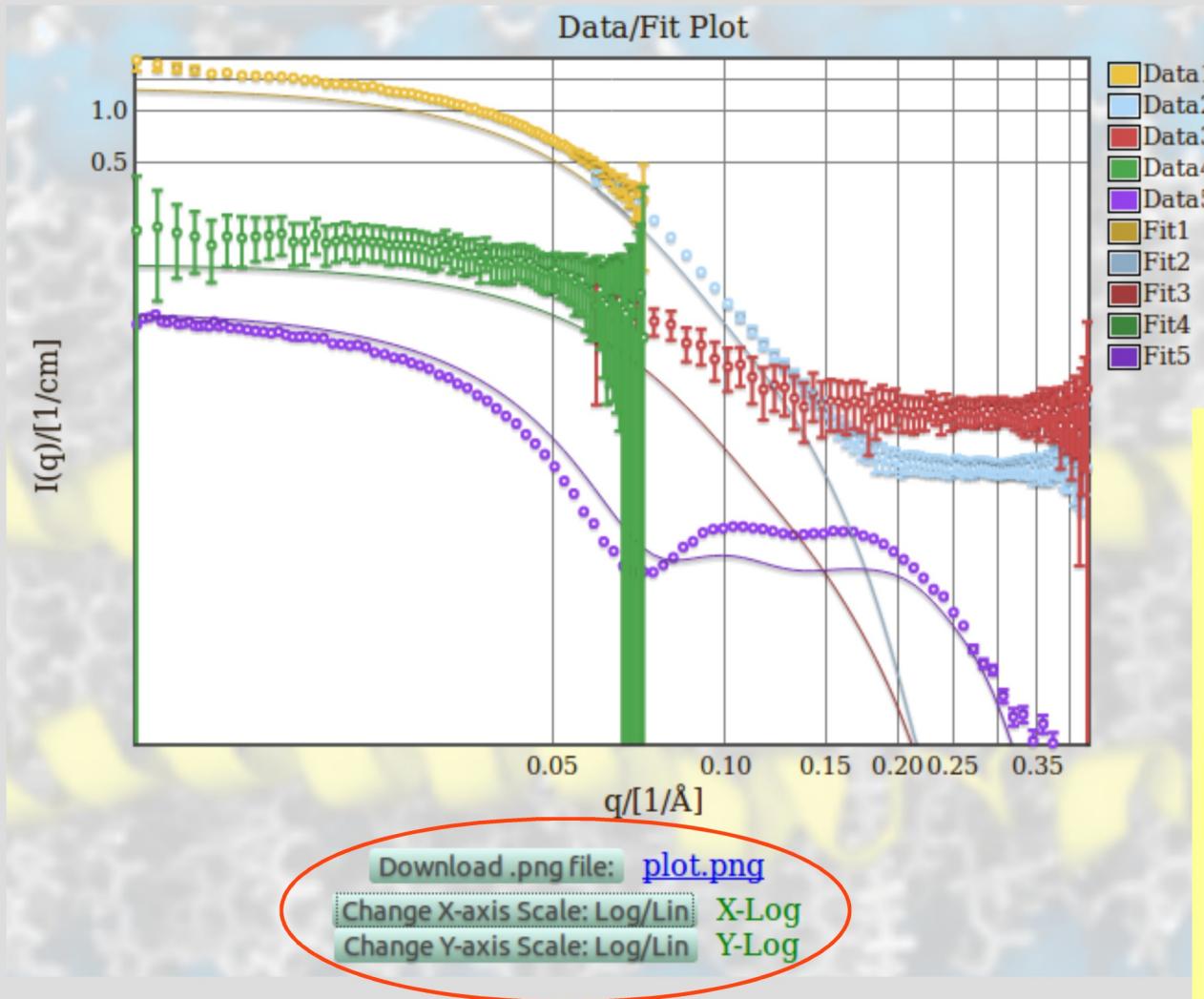
- Uploading parameter file, `lrffile.input` (numerical values, whitespace_separated etc.)



```
{  
  "role" : "input",  
  "id"   : "param_choice_file_anis_core_shell_micelles",  
  "label" : "LOAD FILE: ",  
  "type"  : "lrffile",  
  "setinputfromfile"  : "whitespace-separated_reverselogic",  
  "setinputfromfileids": "..list_of_ids..",  
  "repeat": "model_list_box:anis_core_shell_micelles",  
  "help"   : "Load parameters from file"  
}
```

Advanced “GenApping”: *WillItFit* (2)

- Advanced Plot2D.output options (save to file, change X-, Y-axis scales independently)



jQuery FLOT package
<http://www.flotchart.org>

```
{  
    "role" : "output",  
    "id" : "fit_plot",  
    "label" : "Fit Plot:",  
    "type" : "plot2d",  
    "height": "400px",  
    "width" : "700px",  
    "pan" : "false",  
    "zoom" : "false",  
    "backgroundcolor" : "white",  
    "selzoom" : "true",  
    "changescalex" : "true",  
    "changescaley" : "true",  
    "savetofile" : "true",  
    "rotatedylabel" : "true",  
    "hover" : "true"  
}
```

Advanced “GenApping”: MULCh (2)

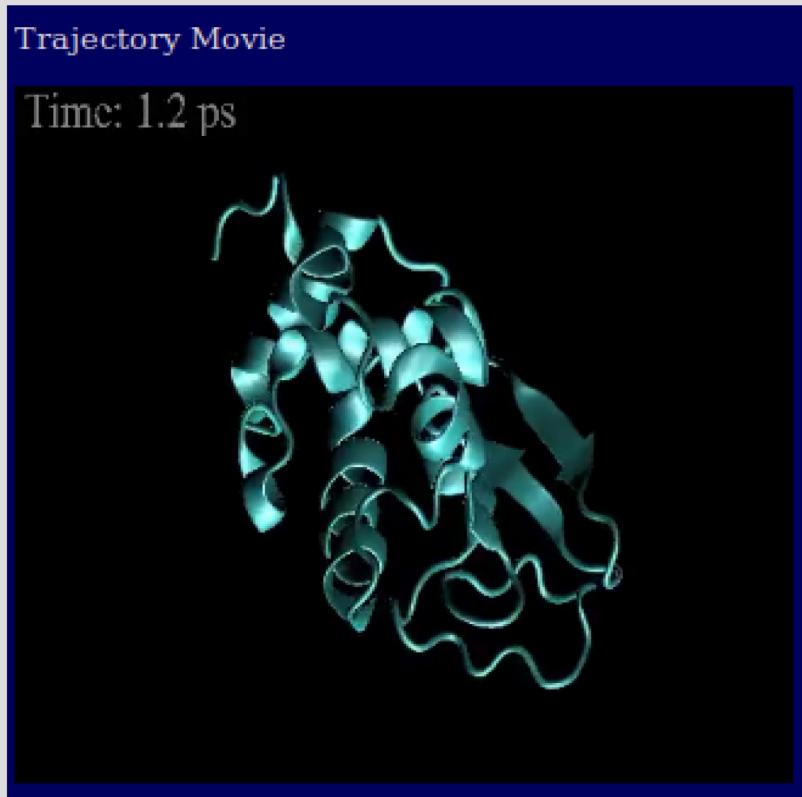
- Calculated fields (inter-dependent numerical fields)

CONTRAST VALUES (opt)			
Vol ₁ (Å)	65	$\Delta\rho_1 =$	155 f_{D_2O+} 23
Vol ₂ (Å)	40	$\Delta\rho_2 =$	435 f_{D_2O+} 234
CONTRAST POINTS			
Upper qR_g limit	1.3		
Number of Contrast Points:	4		
$f_{D_2O}(0-1)$	Refine Scale Scale Value Start Point	Contrast Data	
0.1	✓	1.0 1 Browse... No file selected.	$\Delta\rho_1$ 38.5 $\Delta\rho_2$ 277.5 Vol ₁ /(Vol ₁ +Vol ₂) 0.6190476190
0.2	✓	1.0 1 Browse... No file selected.	$\Delta\rho_1$ 54 $\Delta\rho_2$ 321 Vol ₁ /(Vol ₁ +Vol ₂) 0.6190476190
0.3	✓	1.0 1 Browse... No file selected.	$\Delta\rho_1$ 69.5 $\Delta\rho_2$ 364.5 Vol ₁ /(Vol ₁ +Vol ₂) 0.6190476190
0.4	✓	1.0 1 Browse... No file selected.	$\Delta\rho_1$ 85 $\Delta\rho_2$ 408 Vol ₁ /(Vol ₁ +Vol ₂) 0.6190476190

```
{  
    "role"      : "input",  
    "id"        : "delta_rho_1",  
    "label"     : "Delta rho",  
    "type"      : "text",  
    "repeat"    : "contrast_points",  
    "required"  : "true",  
    "calc"      : "d_rho1*fdo2 + fdo2_1",  
    "pattern"   : "^(-)?(([1-9][0-9]*)|(0))?([.][0-9]+)?([eE][-+]?[0-9]+)?$"  
}
```

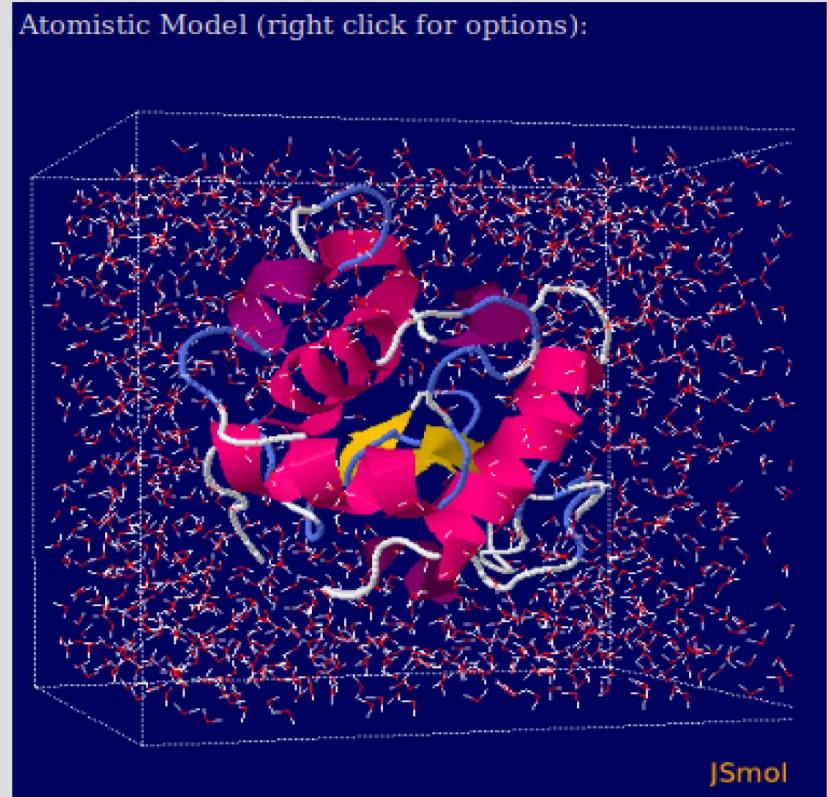
Advanced “GenApping”: ParamMD (1,2)

- Trajectory Movie generation [.mp4, .webm] video.output



```
{  
  "role"      : "output",  
  "id"        : "vid",  
  "type"      : "video",  
  "label"     : "Trajectory Movie",  
  "width"     : 400,  
  "height"    : 450  
}
```

- Structure visualization, JSmol atomicstructure.output



```
{  
  "role"      : "output",  
  "id"        : "outputpdb_view",  
  "label"     : "Atomistic Model",  
  "type"      : "atomicstructure",  
  "jsmoladd"  : "hide HOH;spin on",  
  "width"     : 450,  
  "height"    : 450  
}
```

Advanced “GenApping”: Admin Utilities

- Job History, jobs information within specified time frame (PHP, MongoDB)

NAMDrun - Beta 0.1

Logoff alexey Help on   

Job monitor Integrity check Users User management Job history

Start Date 2016-10-10

End Date 2016-11-03

Submit Reset to default values

FEEDBACK 

name	email	duration (h)	running	finished	cancelled
Totals		102.411	1	221	3
alexey	alexsav.science@gmail.com	1.547	1	99	1
amirayuyue	amirayuyue@gmail.com	0	0	0	0
cpayne	christy.payne@uky.edu	0	0	0	0
danielma	danmart_us@yahoo.com	0	0	0	0
emre	emre@biochem.uthscsa.edu	0.292	0	49	0
graceb	gracebrannigan@gmail.com	0	0	0	0
gumbart	gumbart@physics.gatech.edu	0	0	0	0
hwang	hhwang8@gatech.edu	56.094	0	65	2
jvermaas	joshua.vermaas@nrel.gov	44.476	0	8	0
mocohen	mocohen@uchicago.edu	0	0	0	0
pcardena	pacl3@gatech.edu	0	0	0	0
ttjoseph	tomas.joseph@uphs.upenn.edu	0	0	0	0
genapp.rocks					