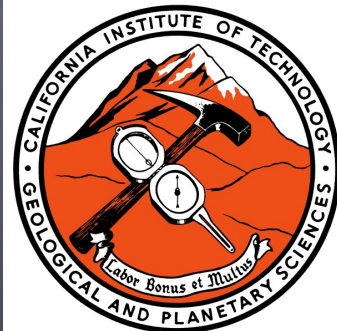


Paula Antoshechkina, Caltech
Paul D. Asimow, Caltech

alphaMELTS for MATLAB/Python

alphaMELTS2 workshop
Rotorua, NZ, January 28-29, 2023

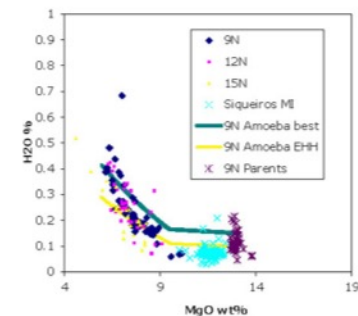
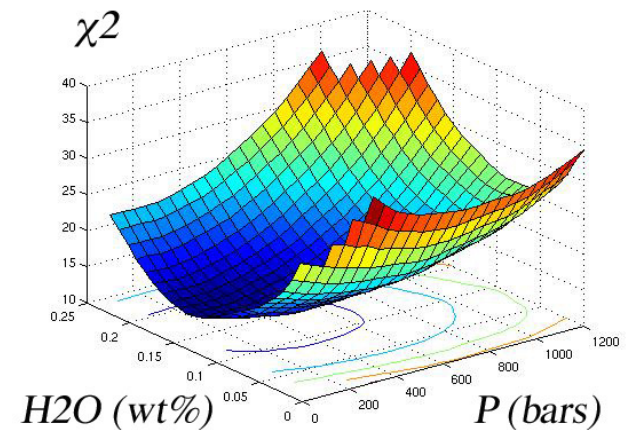
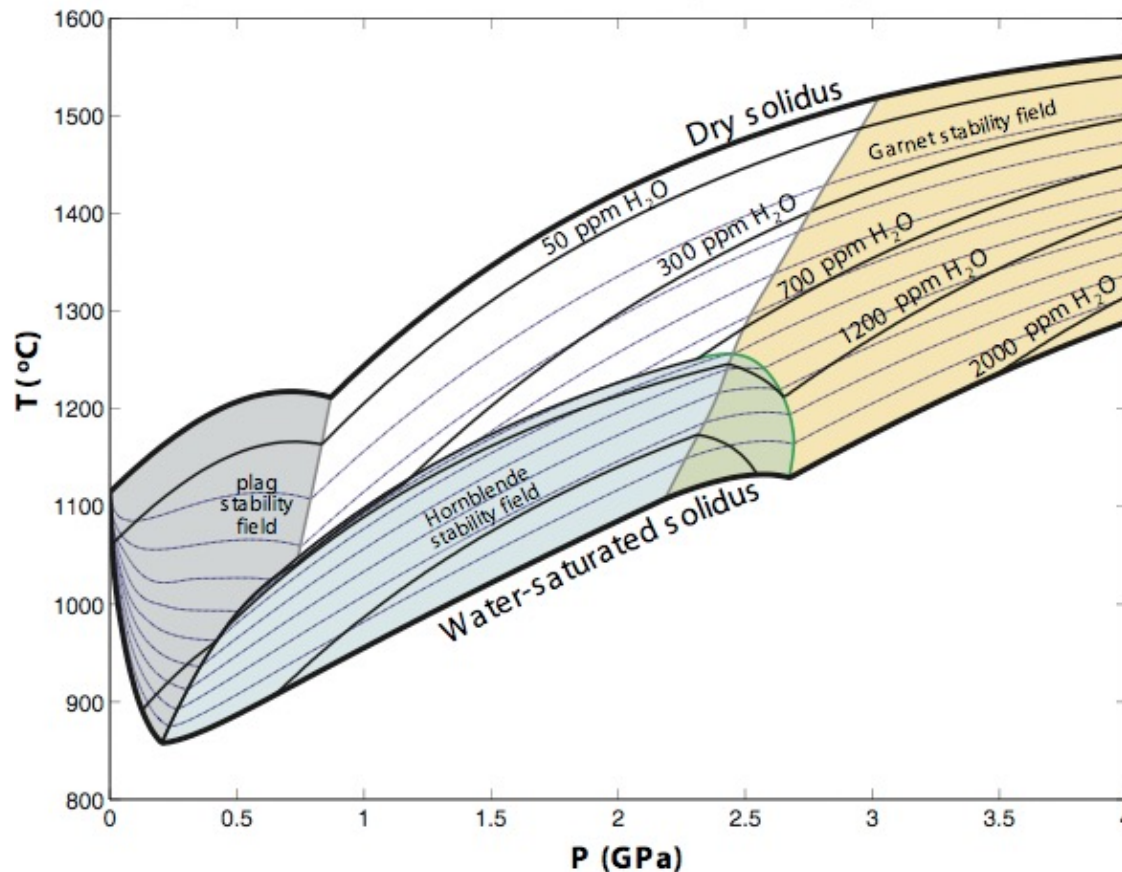
Caltech



Early run-alpha melts.command

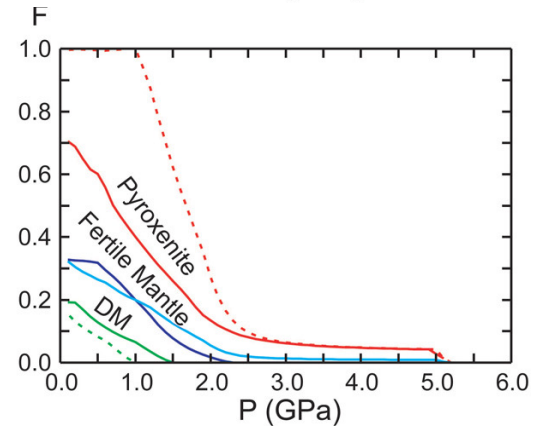
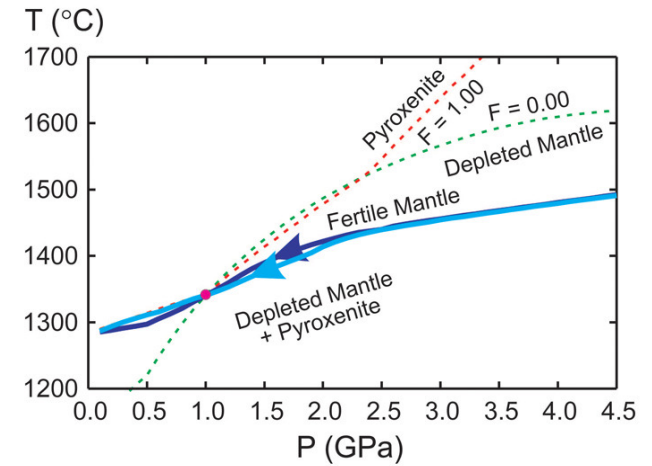
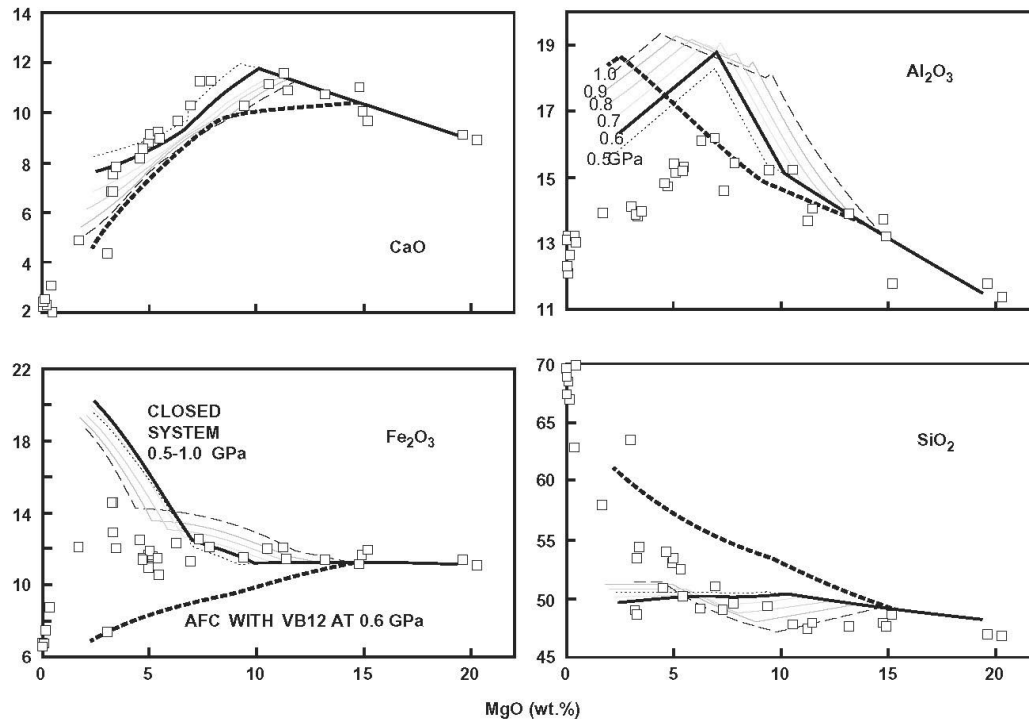
CAN RUN FROM MATLAB, PYTHON, R
- USE TEXT FILES FOR PLOTTING ETC.

pHMEELTS Solidus surface for DMM peridotite plus water

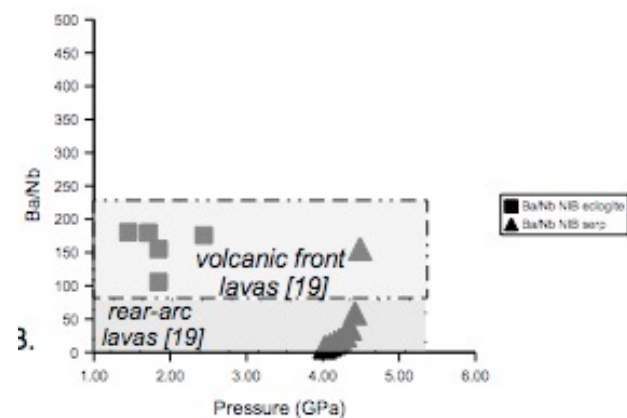
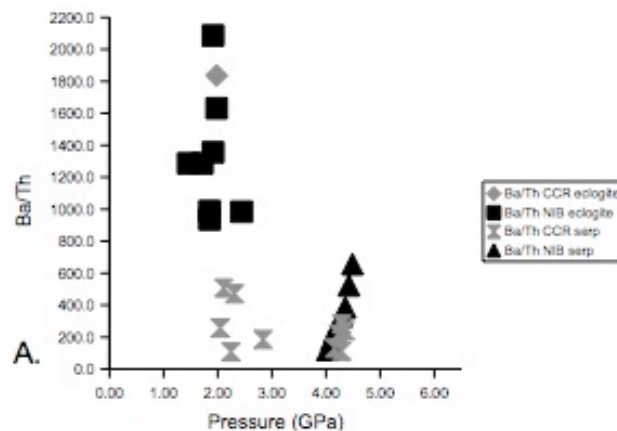
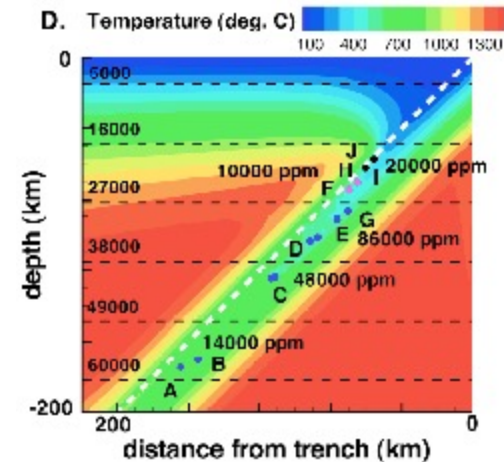
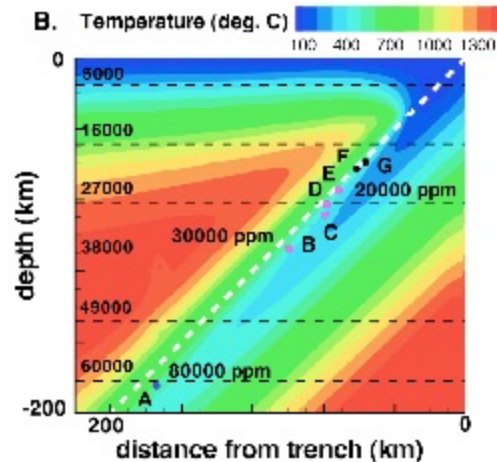


Assimilation / dual_adibat

GUI REQUIRES "SUPPLEMENTAL CALCULATOR"
R.I.P. MELTS SUPPLEMENTAL CALCULATOR ~1995 - 2021




pHMELTS + Conman (text-based)



LEPR/traceDs: lepr.earthchem.org


Dev server at traceds.caltech.edu

[LEPR+ Home](#) [OFM Research Site](#) [ENKI Portal Site](#) [MELTS Site](#) 

LEPR 3.0 / TraceDs 1.0 / DCO/EPC 1.0

Logged in to LEPR+ database as: paulamay. [\[logout\]](#) [\[update user profile\]](#) [\[admin page\]](#) [\[preferences\]](#)

Library of Experimental Phase Relations / TraceDs



LEPR is a database of results of published experimental studies involving liquid-solid phase equilibria relevant to natural magmatic systems. TraceDs is a database of experimental studies involving trace element distribution between liquid, solid and fluid phases.

The servers are physically located at OFM Research Inc. The database engine is MySQL. The web interface is written in PHP.

This web site requires the use of a browser that supports javascript and Adobe™ Flash.

How to cite data downloaded from this database?

- Most importantly: **Cite the original experimental data source.** Secondly, cite the appropriate synthesis publication:
- for **LEPR 3.0**: Hirshmann M.M., Ghiorso M.S., Davis F.A., Gordon S.M., Mukerjee S., Grove T.L., Krawczynski M., Medard E., Till C.B., 2008, Library of Experimental Phase Relations (LEPR): A database and web portal for experimental magmatic phase equilibria. *Geochemistry, Geophysics, Geosystems*, 9, Q03011, doi:10.1029/2007GC001894.
- for **TraceDs**: Nielsen R., Ghiorso M.S., Trischman T., 2015, V33C-3119: What We Have Learned About the Existing Trace Element Partitioning data During the Population Phase of traceDs. AGU Fall Meeting abstracts, San Francisco.

There are three ways to search the database:

Quick search by reference

Quick search by laboratory


Configure complex search ...

Configure

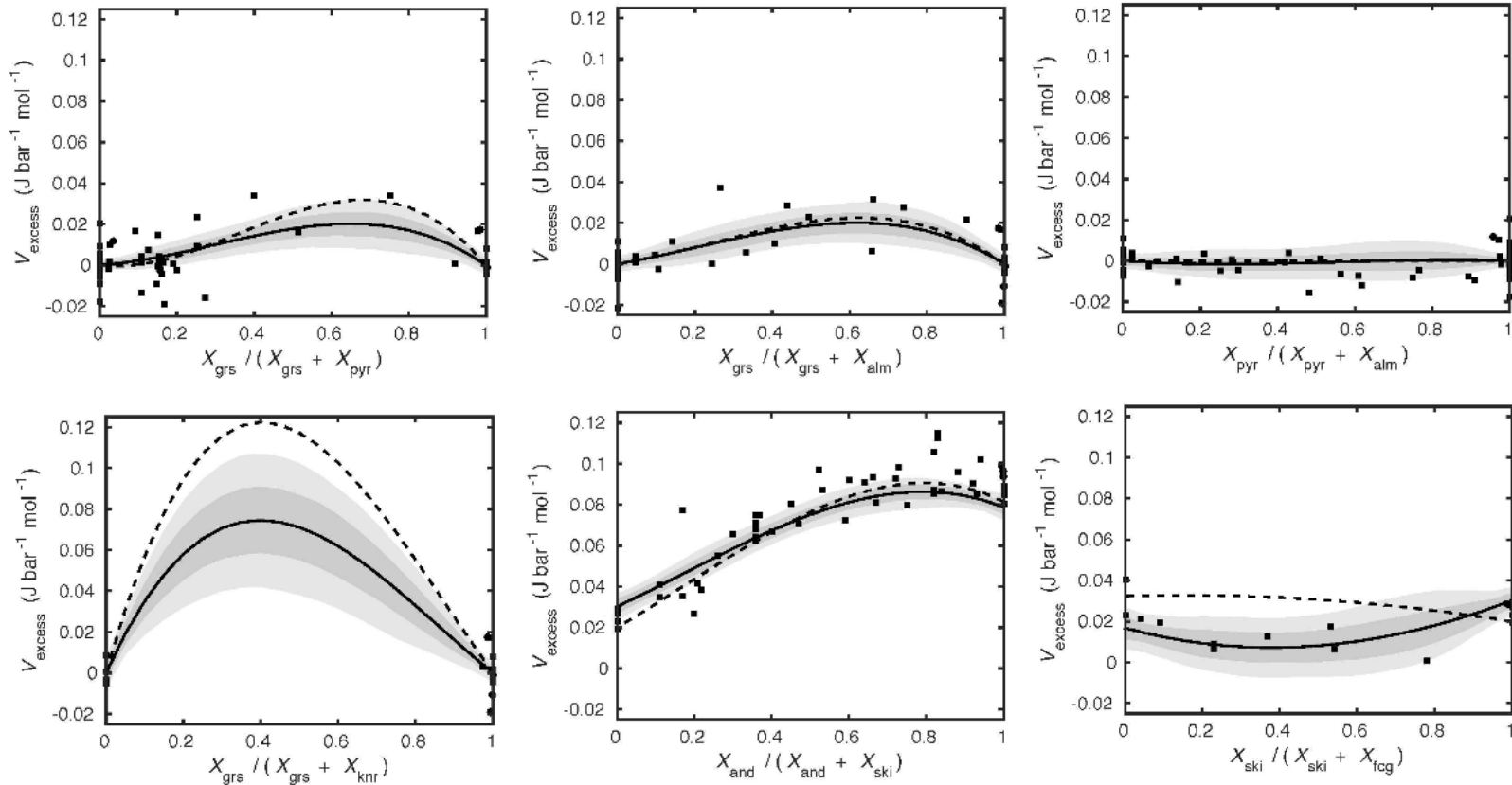
Adam, J. and Green, T.H. [1994]
Agee, C.B. [1990]
Agee, C.B., and Walker, D. [1990]
Agee, C.B., Draper, D.S. [2004]

for compositional data on phases.

Phase names

 [EAR/OCE Funded Research](#) | [Contact Us](#) | ©2005-2018 OFM Research Inc.

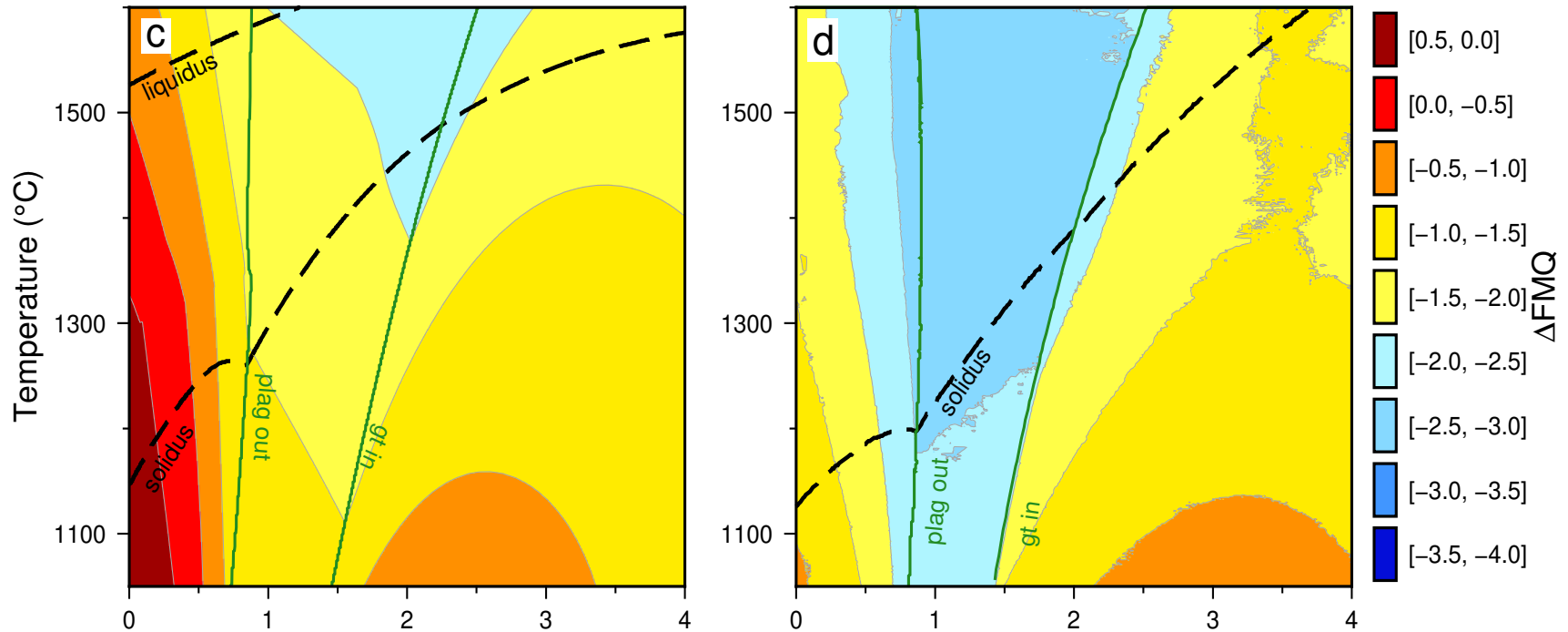
Leprechaun + MELTS functions



MATLAB/Python Outline

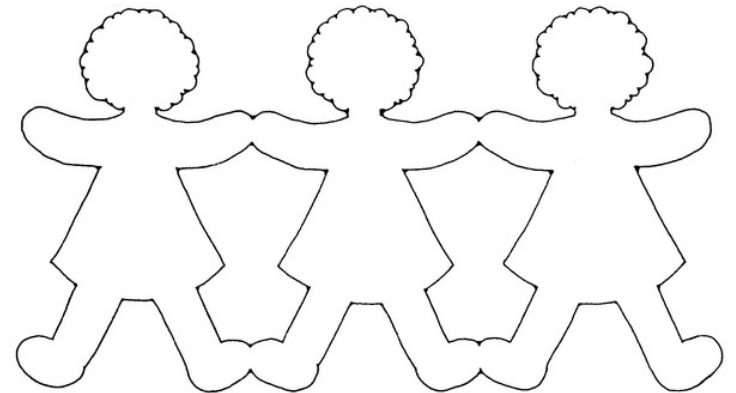
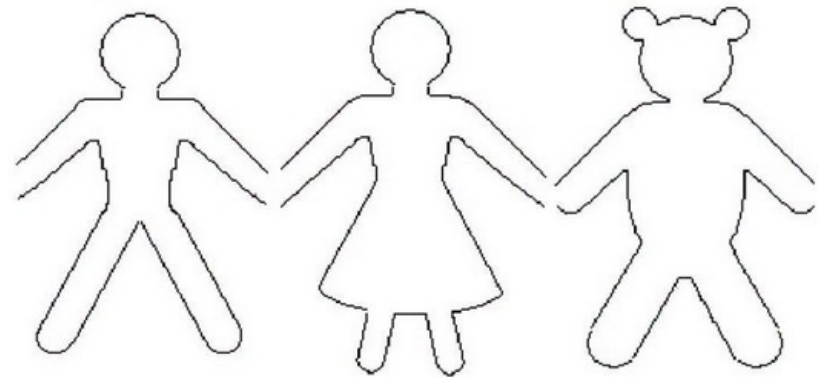
- Library of C functions, libalphaMELTS, that also underlies the alphaMELTS 2 executable
 - Some functions will mimic alphaMELTS 2 menu options
 - Also lower level access so it is very flexible and moderately scalable
- MATLAB/Python wrappers
 - https://gitlab.com/magmasource/MELTS_Matlab
- Was originally called MELTS for MATLAB
 - Starting to integrate more alphaMELTS-specific features
 - <https://agu2018fallmeeting-agu.ipostersessions.com/default.aspx?s=89-81-04-Eg-EA-64-38-13-B4-11-17-CF-90-80-32-85&guestview=true>

MELTS for MATLAB versus Perple_X



MATLAB/Python OOP

- Linked list of MELTS dynamic nodes
 - Each node can have GUI and 'Supplemental calculator'–like functionality
- Thermodynamic engine
 - `ptpath = MELTSdynamic(4)`
 - `ptpath.engine`
- Properties e.g.
 - `ptpath.engine.temperature`
- Methods include
 - `ptpath.engine.findLiquidus`
 - `ptpath.engine.calcEquilibriumState`



The paper doll analogy

- MATLAB is geared towards matrix operations, which makes for compact code for many applications but less so for MELTS calculations
 - The analogy we use in in-person workshops is how describing a group of people using separate lists for each of eye color, hair style etc. would be awkward; what we want is all the information about a particular person in one place
 - We do that by defining a “class”; properties of a particular instance of that class are accessed using the dot (.) operator (or '\$' on R), which you can think of as a bit like “s”:
 paula.eyes.color = 'blue' paul.eyes.color = 'brown'
 - We can also add “methods” to the class, which are the things each instance of the class can do. The example we use in the in-person workshops is waving:
 paula.wave() paul.wave() thequeen.wave()
- In alphaMELTS for MATLAB/Python we take the analogy one step further
 - Each thermodynamic state (MELTSengine) is contained in a MELTSDynamic ‘node’ and the nodes are connected in a linked list
 - Each MELTSDynamic node contains information about which node is next, which node is first, which is the current node... sort of like a chain of paper dolls
 - When we add or delete nodes, join lists, or sort the list then the information about each node’s place in the list is automatically updated... no scissors or tape required!
 - There are also tools to extract vectors and matrices of values from the MELTSDynamic list, as that’s what we need for plotting

.melts file settings

- easyMelts / GUI
 - **Initial** Composition:
 - Log fO₂ Path
 - Mode: fractionate solids
 - **Assimilant**: ...
- alphaMELTS 2
 - The above plus ...
 - Mode: fractionate olivine
 - Log fO₂ Offset
 - **Set** min F:
- alphaMELTS for MATLAB/Python
 - **Initial** lines are all set another way
 - **Assimilant** lines will be set another way
 - Not yet implemented
 - Other lines as strings
 - .engine.set SystemProperties
 - **Set** lines (after workshop)
 - (Set) Output: ...
 - Other **Set** lines

MATLAB/Python classes

- MELTSDynamic handles linked list and system wide properties associated with choice of model (e.g. rhyolite-MELTS 1.2.0; pMELTS...)
 - E.g. what phases are available?
- All the hard work is done in MELTSEngine
 - E.g. what phases are actually present?
- MELTSstatus (and MELTSevent) track interaction with the C library of MELTS functions
 - How many calculations have we done (line number in any output files)? Did something go wrong, can it clean up?
 - Console on Windows version

More tools

- MELTSmap makes it easier to extract vectors and matrices of data from the 'paper doll'-like structures of properties needed for thermodynamic calculations
- Tools to join lists, add or remove nodes, sort lists of nodes
- Start a new MELTSdynamic list to switch between pMELTS and rhyolite-MELTS models
- Handles errors gracefully

Getting help

- MATLAB type e.g:
"help MELTSengine"
 - Click on "Reference page for MELTSengine"
 - Explore the properties and methods
 - Will show how to call 'methods' (like functions), what output is like etc.
- Python (limited) type e.g:
"help(MELTSdynamic())"
 - Also check .py source
- Coming soon: Doxygen docs for all platforms.

MELTSengine is a class.
`obj = MELTSengine(cMode, title)`

Class Details
Superclasses: [matlab.mixin.SetGet](#), [matlab.mixin.Copyable](#)
Sealed: false
Construct on load: false

Constructor Summary
[MELTSengine](#)

Property Summary

X	For end-member or oxide properties or results of evaluate saturation
activity	For end-member or oxide properties (using solid/liquid or phase compositions)
affinity	output map for evaluate saturation (can be called before or after equilibration)
bulkComposition	For regular MELTS calculations (composition in grams by default)
calculationMode	
cp	
d2vdp2	
d2vdt2	
d2vdt2dp	
dcpdt	
dispComposition	output map in wt% (can be input for supplemental calculator)
dvdp	
dvdt	
g	output maps for "bulk" and phases

Method Summary

	addlistener	Add listener for event.
	calcEndMemberProperties	
	calcEquilibriumState	
	calcMolarProperties	
	calcPhaseProperties	
	calcSaturationState	
Sealed	copy	Copy MATLAB array of handle objects.
	copyAndKeepOutput	
protected	copyElement	Handles point to the same entities, unless reset
	delete	Delete a handle object.
	eq	== (EQ) Test handle equality.
	findLiquidus	

Examining alphaMELTS classes

- Can examine particular instance on MATLAB command line
- Tab completion is your friend!

```
>> ptpath
```

```
ptpath =
```

[MELTSdynamic](#) with properties:

```
calculationMode: 4
modeString: 'MELTSandCO2_H2O'
systemNames: [1x46 string]
endMemberFormulas: [46x1 MELTMap]
endMemberWeights: [46x1 MELTMap]
failureHandle: [1x1 event.listener]
nodeIndex: 276
engine: [1x1 MELTSEngine]
Next: [0x0 MELTSDynamic]
Prev: [1x1 MELTSDynamic]
First: [1x1 MELTSDynamic]
Last: [1x1 MELTSDynamic]
```

```
>> ptpath.engine
```

```
ans =
```

[MELTSengine](#) with properties:

```
status: [1x1 MELTSstatus]
calculationMode: 4
title: []
meltsIndex: 275
runMode: 1
pressure: 250
temperature: 1.1892e+03
reference: -1.4584e+05
logfO2: -13.2238
systemProperties: [2x1 string]
bulkComposition: [19x1 double]
liquidNames: "liquid1"
solidNames: ["olivine1" "plagioclase1" "fluid1"]
phaseNames: [1x1052 string]
phaseComposition: [19x1 double]
molarComposition: []
g: [0x1 MELTMap]
h: [0x1 MELTMap]
s: [0x1 MELTMap]
v: [0x1 MELTMap]
cp: [0x1 MELTMap]
dcpdt: [0x1 MELTMap]
dvdt: [0x1 MELTMap]
dvdp: [0x1 MELTMap]
d2vdt2: [0x1 MELTMap]
d2vdt dp: [0x1 MELTMap]
d2vdp2: [0x1 MELTMap]
molwt: [0x1 MELTMap]
rho: [0x1 MELTMap]
mass: [0x1 MELTMap]
viscosity: [0x1 MELTMap]
dispComposition: [0x1 MELTMap]
affinity: [0x1 MELTMap]
X: [0x1 MELTMap]
activity: [0x1 MELTMap]
mu0: [0x1 MELTMap]
mu: [0x1 MELTMap]
```

Features

- MATLAB
 - Able to change MELTS model during calculation
 - Also should scale
- Python
 - Not able to change MELTS model during run
 - Complements ENKI tools
- Both
 - Good for desktop applications
 - Can change system in between call to C library of functions
 - Can in principle do (simple) MCS-like calculations and dual_adiabat-like calculations.
 - Can also output _tbl.txt and .tbl text files and process with "run_alphamelts.command -x"

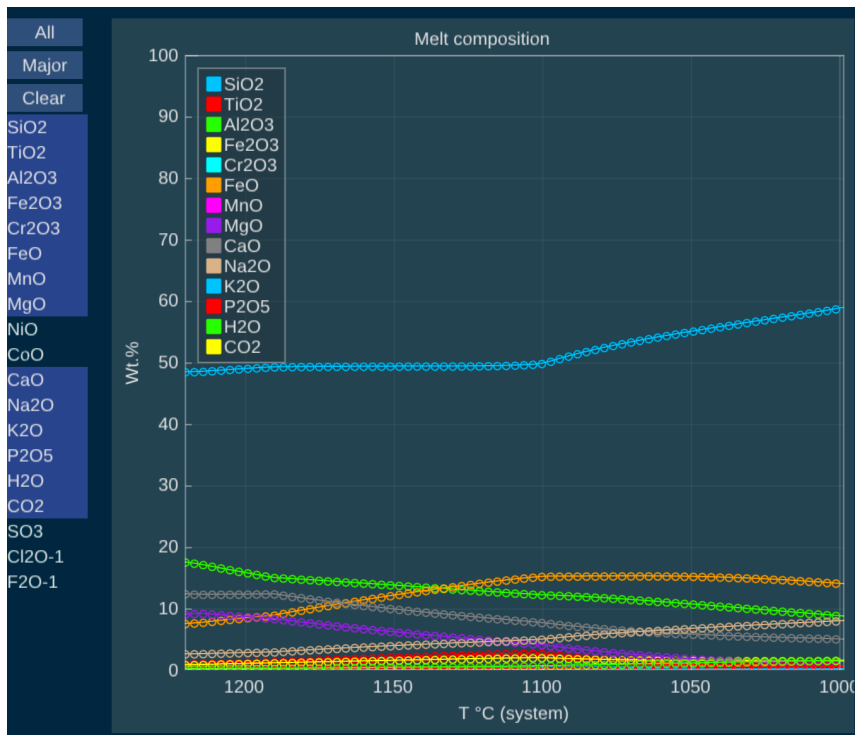
Compare ENKI (enki-portal.org)

- alphaMELTS for Python
 - Based on “xMelts” C code (gitlab.com/ENKI-portal)
 - Codebase for rhyolite-MELTS GUI and Magma Chamber Simulator’s MELTS-batch
 - Adapted for alphaMELTS 2 and alphaMELTS for MATLAB/Python
 - Relatively small and easy local installation
 - Classes and methods make automating typical tasks and plotting results easy
 - Will have higher level alphaMELTS menu-like functions
 - Will have trace elements and pHMELTS eventually
- ENKI Thermoengine
 - Uses “Thermoengine” ObjectiveC and Python code
 - Run online or locally via Docker
 - Low level methods allow for more flexibility than alphaMELTS for Python, but with a steeper learning curve
 - Best way to get started is with Workshop Jupyter Notebooks
 - Will have highly optimised “coder” module-generated C code eventually
 - Will scale efficiently and is recommended for larger coupled thermodynamic and fluid-dynamic simulations

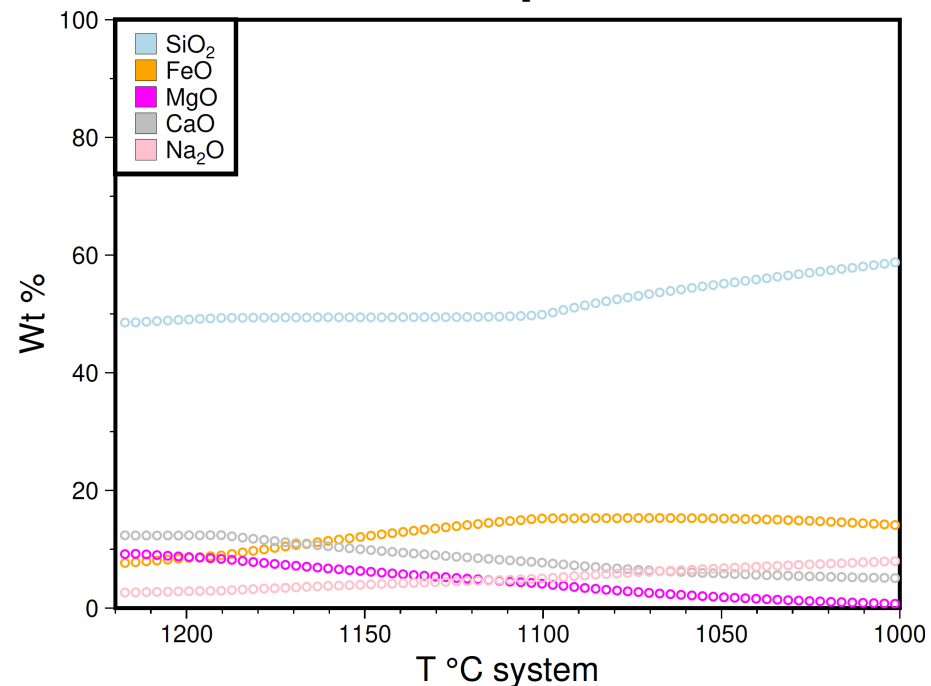
Comparisons of the 'tutorial'

- Compare standalone alphaMELTS to easyMelts
 - Also very useful for debugging (+console)

- Also Compare MATLAB and Python with GMT
- Compare R (next slide)



Melt composition



More 'tutorial' comparisons

- Compare ENKI's Thermoengine
 - Steeper learning curve (e.g. no "Find Liquidus" unless you code it yourself)
 - Current code is fast but takes time to get files from ENKI server in the cloud or from the docker image
 - Future Thermoengine code will be ~10x faster so more scalable than alphaMELTS for Python

Initialize tools and packages that are required

```
[12]: from thermoengine import equilibrate
import matplotlib.pyplot as plt
import numpy as np
%matplotlib inline
```

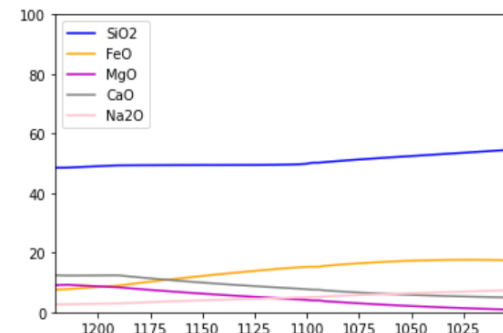
...

Create a MELTS v 1.0.2 instance.

Rhyolite-MELTS version 1.0.2 is the default model.

```
[13]: melts = equilibrate.MELTSmodel()
```

```
success, Minimal energy computed. 1022.30 50.00
success, Optimal residual norm. 1019.30 50.00
success, Optimal residual norm. 1016.30 50.00
success, Optimal residual norm. 1013.30 50.00
success, Optimal residual norm. 1010.30 50.00
success, Optimal residual norm. 1007.30 50.00
success, Optimal residual norm. 1004.30 50.00
```



R with alphaMELTS for Python

You need to put an 'L' after any integer

```
1 library(reticulate)
2 library(reshape)
3 library(ggplot2)
4
5 use_python("~/usr/local/bin/python3")
6 rmelts <- import("meltsdynamic")
7
8 liquidus = rmelts$MELTSdynamic(1L)
9
10 pressure = 500.0
11 temperature = 1200.0
12
13 bulk = c(48.68, 1.01, 17.64, 0.89, 0.03, 7.59, 0.0, 9.10, 0.0, 0.0, 12.45, 2.65)
14
15 liquidus$engine$setBulkComposition(bulk)
16 liquidus$engine$pressure = pressure
17
18 # Note that as of Nov 18th, 2020 temperature is in Celcius
19 liquidus$engine$temperature = temperature
20
21 liquidus$engine$findLiquidus()
22 temperature = liquidus$engine$temperature
23
24 liquidus$engine$setSystemProperties("Mode", "Fractionate Solids")
```

'\'' in Python version replaced by '\$' in R

```
reverse() + scale_y_continuous(limits=c(0,100)) +
+   labs(x='T °C (system)', y='Wt %', title='Melt composition') + theme_
+   bw() + theme(legend.position = c(0.1, 0.75))
>
> # Can extend the dataframe with rbind e.g:
> df_test_long <- rbind(df_test_long, data.frame(temps = df_test$temps, variable='mas
+   s', value = ptpath$getListProperty('mass', 'liquid1')))
>
> ggplot(subset(df_test_long, df_test_long$variable %in% c("sio2", "feo", "mgo", "ca
+   o", "na2o"))) +
+   geom_point(aes(x=temps, y=value, color=variable)) + scale_x_reverse() + scale_y_c
+   ontinuous(limits=c(0,100)) +
+   scale_color_manual(values=c("#56B4E9", "#E69F00", "#660099", "#999999", "#FFCC9
+   9")) +
+   labs(x=expression("T ~-degree~"C (system)"), y='Wt %', title='Melt composition')
+   + theme_bw() + theme(legend.position = c(0.1, 0.75))
```

Environment

Variable	Value
liq_comp	num [1:19, 1:75] 48.5102 1.0065 17.5785 0.8869 ...
liquidus	<meltsdynamic.MELTSdynamic object at 0x11840bfd...>
melts	<Null pointer>
meltsdynamic	<Null pointer>
ptpath	<meltsdynamic.MELTSdynamic object at 0x11846391...>

Values

Variable	Value
bulk	num [1:19] 48.68 1.01 17.64 0.89 0.03 ...
pressure	500
rmelts	<module 'meltsdynamic' from '/Users/psmith/Develo...>
temperature	1220.3125
temps	num [1:75] 1220 1217 1214 1211 1208 ...

Melt composition

variable

- sio2
- feo
- mgo
- cao
- na2o

Wt %

T °C (system)

Python / R version

- Python version requires tinynumpy
 - `pip install tinynumpy`
 - `pip3 install tinynumpy`
- Running in Rstudio
 - `reticulate::install_miniconda()`
 - `reticulate::py_install("tinynumpy", method="pip")`
 - `reticulate::repl_python()`

Future work and links

- alphaMELTS@CIT
 - alphamelts.caltech.edu
 - 'Wiki' or similar documentation
- MAGMA@Caltech
 - magmasource.caltech.edu
 - For alphaMELTS 1.9 and all other MELTS interfaces
- DACITE
 - Database of AlphaMELTS at CIT Examples
 - Standalone, MATLAB, Python, Jupyter, R
- Trace Elements
- Code will eventually be Open Source on Github

alphaMELTS temp home on GitLab

The screenshot shows the GitLab web interface for the project **MELTS_Matlab**. The top navigation bar includes links for Projects, Groups, Activity, Milestones, Snippets, and a search bar. The left sidebar contains a list of project features: Project, Details, Activity, Releases, Cycle Analytics, Repository, Issues (0), Merge Requests (0), CI / CD, Operations, Registry, Wiki, Snippets, and Settings. The main content area displays the project details for **MELTS_Matlab**, including the project ID 8231666, 58 commits, 1 branch, 0 tags, and 32.3 MB of files. A progress bar is shown below the project statistics. The **Auto DevOps** section is highlighted, indicating that it will automatically build, test, and deploy the application based on a predefined CI/CD configuration. A button labeled "Enable in settings" is available. Below this, the current branch is **master**, and the repository path is **MELTS_Matlab / +**. A recent commit by Paula Antoshechkina is shown, titled "Bug fix for when calculation fails to converge in MATLAB." with the commit hash **caf5b960**. At the bottom, there are buttons to add a README, CHANGELOG, CONTRIBUTING file, Kubernetes cluster, and set up CI/CD. A table lists the project's files and their last commit details.

Name	Last commit	Last update
examples	Bug fix for when calculation fails to converge in MATLAB.	1 day ago
mextests	Last few routines added to python. More debugging and a li...	2 months ago
package	Bug fix for when calculation fails to converge in MATLAB.	1 day ago
.gitignore	More mex testing.	3 months ago

alphaMELTS temp home on GitLab

- Sign up at gitlab.com
 - Send your username
 - Sign up for notifications
- Watch this space!
 - XML output that can be fed into GCDkit
 - fO_2 alternative
 - Different ways to do assimilation
 - Phase diagram mode
 - Integrated output file
 - Integration with pyrolite
 - Trace elements and pHMELTS