

Summer 2020

Mapping the most extremely anomalous features within the Earth
 Open OnDemand web portal for teaching
 Python Tips
 New CHPC Class Account Provisioning
 CHPC Fall 2020 Presentation Schedule

Mapping the most extremely anomalous features within the Earth

Research Highlight · Michael Thorne, Dept. of Geology & Geophysics

At a depth of 2891 km the boundary between the Earth's solid rocky mantle and liquid outer core, the core-mantle boundary (CMB) provides the largest contrast in physical properties anywhere in the Earth (Earth layering and naming conventions are shown in Fig. 1).

within the D'' layer with whole mantle plumes rising to the Earth's surface to generate hot spot volcanoes such as Hawaii, Iceland, or Samoa. To complete the circuit, subducted slabs of Earth tectonic plates drip down from the surface, with a final resting place on top of the CMB, making D'' a slab graveyard. Also embedded within D'', and possibly extending several hundreds of km above it, are regions referred to as Large Low Velocity Provinces (LLVPs). LLVPs are continent sized regions of the lower-most mantle that show reduced seismic wave speeds relative to the average mantle (wave speeds are reduced by as much as 5% with respect to the average). Two LLVPs exist in antipodal regions of the Earth, one beneath Africa and the other beneath the Pacific Ocean, but their origin is still unknown. Because the boundaries between LLVP and normal mantle appear sharp (on the order of 10's of km) LLVPs have been argued to have a compositional origin and may be remnants from the initial differentiation of the planet. Yet, an even more enigmatic feature exists sitting atop the CMB, referred to as ultralow-velocity zones (ULVZs).

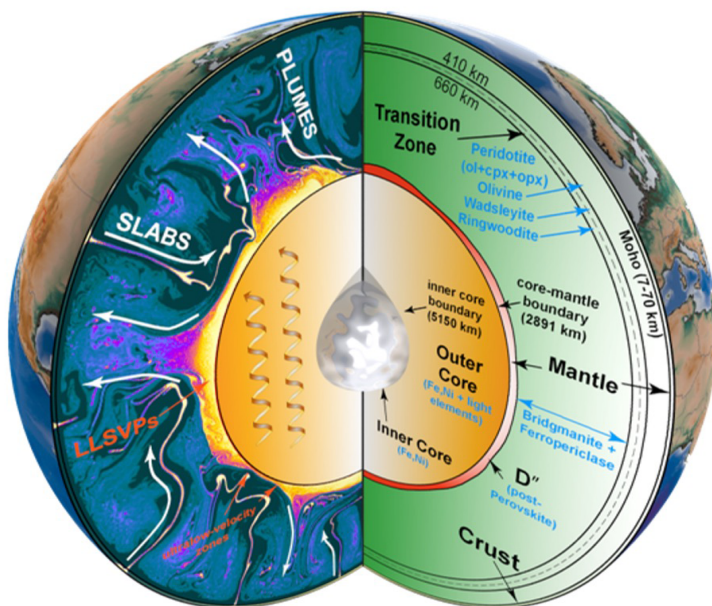


Figure 1. *Cartoon image of the Earth's interior structure.* On the right-hand side the standard 1-D Earth layering is shown based on composition. But, 3D structure exists in addition to the 1D layering as shown on the left-hand side. Here we show a snapshot of a thermochemical convection simulation (by Mingming Li, Arizona State University). Locations of possible ULVZs near the CMB are indicated

The change in density at the CMB ($\Delta\rho$ from mantle to outer-core is $\sim 4.3 \text{ g/cm}^3$) is larger than at Earth's surface going from rock to air ($\Delta\rho$ between granite and air is $\sim 2.6 \text{ g/cm}^3$). In addition to being both a compositional and mechanical layer boundary, the CMB also marks the start of a thermal boundary layer, associated with the bottom-most layer of the mantle referred to as D'' (pronounced D double prime). Convective processes are thought to start

No other feature within the Earth has properties as extreme as ULVZs. ULVZs appear to be relatively thin features (roughly 10 to 20 km thick) with a modest increase in density ($\sim 10\%$), but with exceptionally reduced seismic wave speeds. S-wave speed reductions have been modeled to be as large as 45% with respect to average mantle properties. Nothing else in the Earth has such dramatic variations in properties, yet we know almost nothing about these features. ULVZs were first discovered in 1993 [Garnero et al., 1993], and their existence has now been confirmed in over 50 seismic studies [see recent review by Yu and Garnero, 2018]. Despite this, as of a recent review conducted in 2018 less than 20% of the CMB had been probed for ULVZ existence. As a result, we barely know where they do and do not

exist, which is key to deciphering what they may physically represent. Furthermore, strong modeling tradeoffs exist between ULVZ seismic velocities and morphology. As a result, we don't have strong constraints on the ULVZ elastic parameters (P-, and S-wave velocity and density) which are necessary in order to determine their possible compositions.

Despite how little we know about ULVZs, it appears that some of them may be tied to the genesis of hot spot volcanism. Some of the Earth's largest hot spot volcanoes show continuous low seismic velocity conduits, evidence for whole mantle plumes, all of the way from the volcanoes at the surface down to their roots in large ULVZs at the CMB [Cottaar and Romanowicz, 2012; French and Romanowicz, 2015; Thorne et al., 2013; Yuan and Romanowicz, 2017]. Several past efforts have linked ULVZs to hot spot volcanism [Williams et al., 1998], a link that may in part be fueled by speculation that ULVZs require some amount of partial melt to explain the massive velocity reductions [Williams and Garnero, 1996]. Roughly 20% of the lower mantle is comprised of the mineral ferropericlasite, (Fe,Mg)O, which has a low melting temperature relative to other lower mantle constituents. Thus, it is thought that the existence of partial melt likely involves melting of this Fe-bearing mineral. However, some ULVZs appear near the boundaries of LLVPs and may have purely compositional origins. One such possibility is that the ULVZs are accumulations of heavily Fe-enriched ferropericlasite, which can also explain the dramatic velocity reductions [Brown et al., 2015; Wicks et al., 2010]. Furthermore, some large ULVZs appear to be located on the boundaries of old subducted slabs, such as beneath Northern Mexico and Florida, regions that are far removed from hot spot volcanoes and LLVPs [Thorne et al., 2019]. In this case it is possible that the ULVZ is created by partially melting Mid-Ocean Ridge Basalt (MORB) in the subducted slab. A scenario in which the melt is also expected to entrain Fe.

Many of the scenarios discussed above for ULVZ creation involve Fe enrichment, which could significantly impact the Earth's dynamic processes. For example, because FeO metallizes at CMB pressures it should have both a higher thermal and electrical conductivity and density than ambient mantle.

As a result, ULVZs could act to stabilize and anchor whole mantle plumes [Manga and Jeanloz, 1996], and may also influence the Earth's magnetic field affecting magnetic field reversals [Glatzmaier et al., 1999], field reversal paths [Kutzner and Christensen, 2004] and even nutation of the core which can result in periodic variations in the Earth's rotation [Buffett et al., 2000].

In this project, we are working to produce a global map of ULVZ locations and to determine their physical properties. In order to do this, we have created a new global catalog of ULVZ sensitive seismic waveforms by gathering data from available seismic repositories around the world. Here we have collected data focusing on the SPdKS seismic phase. SPdKS starts as an S-wave leaving the earthquake source, and strikes the CMB at the critical angle for P-wave diffraction. These short legs of P-wave diffraction along the CMB are highly sensitive to ULVZ structure. Some of this energy leaks into the lower mantle as a P-wave (K) and is converted back to an S-wave as it exits the core on its return trip through the mantle. A complimentary phase also exists, called SKPdS in which the Pd leg occurs on the receiver-side of the ray path (ray paths are shown in Fig 2).

In our new data collection, we collected a total 606,770 seismograms from deep earthquakes (≥ 75 km) occurring between 1990 and 2017. We utilized machine learning approaches to assist us in narrowing down our data set to 271,602 high quality (low noise) seismic traces. For our first data analysis step, we used 58,155 seismograms in a narrow earthquake to seismometer distance range (epicentral distance). This range was chosen first as it displays the most characteristic waveforms that can be associated with ULVZs. Data coverage from this reduced dataset, covers 56.9% of the CMB by surface area, which is more than 30% greater than all other previous studies combined, and represents the most comprehensive coverage of the CMB to date. From this dataset we detected 2,222 waveforms that provide evidence for ULVZ existence [Thorne et al., 2020]. Because the seismic energy follows both the SPdKS and SKPdS paths, we don't know exactly where the ULVZs are located. Thus, we inverted for ULVZ location using the principle of parsimony. That is, we invert for the minimum number of ULVZ

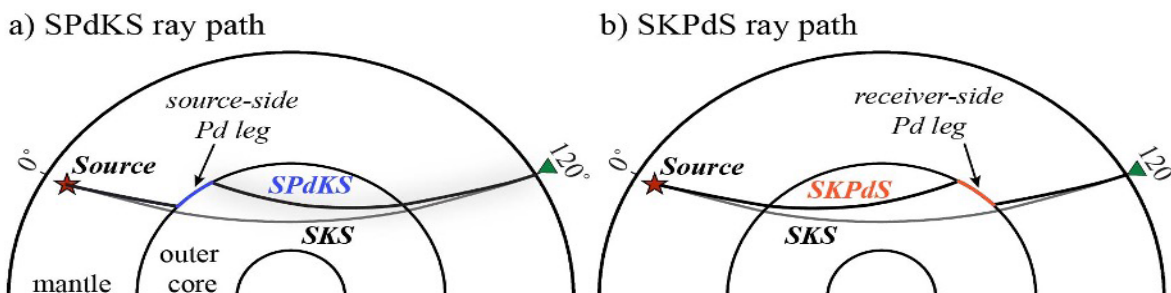


Figure 2. (a) Ray path for SKS (black) and SPdKS (blue) for a receiver (green triangle) located at an epicentral distance of 120°. (b) Ray path for SKS (black) and SKPdS (orange).

locations that can explain the highly anomalous observations. Here we used a genetic algorithm to search for solutions using CHPC resources for the computation. But, because the answer is non-unique, we solved the problem several thousands of times using different starting random seeds to initiate the genetic algorithm. In this way, we can build a probabilistic map of ULVZ location. The first results of this analysis are shown in Fig. 3.

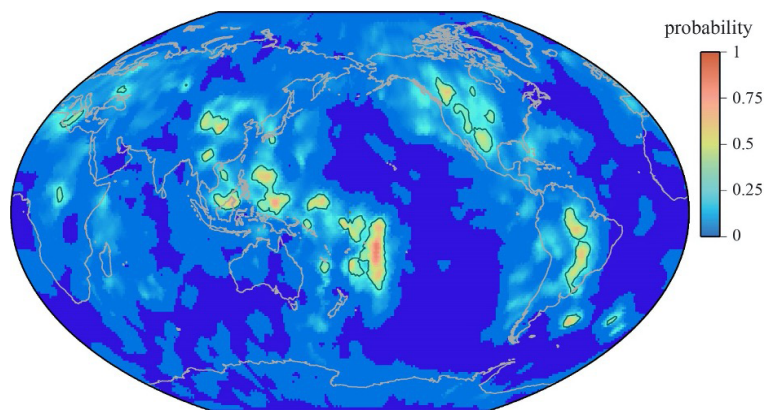


Figure 3. Probability of ULVZ existence on the core-mantle boundary. The calculated probabilities are shown with a contour drawn around a probability of 0.3 to highlight some of the regions with the highest probabilities. Dark blue areas are the regions with no data coverage.

Here we see a distribution of approximately 20 ULVZs, half of which were unknown previous to this study. Some of the new ULVZs (e.g., directly beneath the Caroline hot spot) are located directly above hot spots providing additional links to hot spot volcanism. But the majority of the new ULVZs appear to be located near the boundaries of past subduction (e.g., beneath S. America and East Asia), providing evidence for ULVZs in a region of the lower mantle previously thought to be mostly devoid of ULVZ presence. Our results also show that the ULVZ underlying the Samoa hot spot is approximately twice as large as previously known, making it the largest known ULVZ. The project is now focusing on detecting ULVZs using the rest of the data at all source-receiver distances. These additional data are sensitive to ULVZ structure, but show more subtle waveform effects. However, by including these data we may be able to obtain up to 90% coverage of the CMB area. In addition, we are currently beginning to apply waveform modeling analysis within the discovered ULVZs in order to determine their elastic parameters and thus put constraints on their composition and origin

References:

- Brown, S. P., M. S. Thorne, L. Miyagi, and S. Rost (2015), A compositional origin to ultralow-velocity zones, *Geophys. Res. Lett.*, 42, 1-7, doi: 10.1002/2014GL02097.
- Buffett, B. A., E. J. Garnero, and R. Jeanloz (2000), Sediments at the Top of Earth's Core, *Science*, 290, 1338-1342.
- Cottaar, S., and B. Romanowicz (2012), An unusually large ULVZ at the base of the mantle near Hawaii, *Earth Planet. Sci. Lett.*, 355-356, 213-222, doi: 10.1016/j.epsl.2012.09.005.
- French, S. W., and B. Romanowicz (2015), Broad plumes rooted at the base of the Earth's mantle beneath major hotspots, *Nature*, 525, 95-99, doi: 10.1038/nature14876.
- Garnero, E. J., S. P. Grand, and D. V. Helmberger (1993), Low P-wave velocity at the base of the mantle, *Geophys. Res. Lett.*, 20(17), 1843-1846, doi: 10.1029/93GL02009.

Glatzmaier, G. A., R. S. Coe, L. Hongre, and P. H. Roberts (1999), The role of the Earth's mantle in controlling the frequency of geomagnetic reversals, *Nature*, 401, 885-890.

Kutzner, C., and U. R. Christensen (2004), Simulated geomagnetic reversals and preferred virtual geomagnetic pole paths, *Geophys. J. Int.*, 157, 1105-1118, doi: 10.1111/j.1365-246X.2004.02309.x.

Manga, M., and R. Jeanloz (1996), Implications of a metal-bearing chemical boundary layer in D'' for mantle dynamics, *Geophys. Res. Lett.*, 23(22), 3091-3094.

Thorne, M. S., N. Takeuchi, and K. Shiomi (2019), Melting at the edge of a slab in the deepest mantle, *Geophys. Res. Lett.*, 46, 8000-8008, doi: 10.1029/2019GL082493.

Thorne, M. S., E. J. Garnero, G. Jahnke, H. Igel, and A. K. McNamara (2013), Mega Ultra Low Velocity Zone and Mantle Flow, *Earth Planet. Sci. Lett.*, 364, 59-67, doi: 10.1016/j.epsl.2012.12.034.

Thorne, M. S., S. Pachhai, K. Leng, J. K. Wicks, and T. Nissen-Meyer (2020), New candidate ultralow-velocity zone locations from highly anomalous SPdKS waveforms, *Minerals*, 10, 211, 1-26, doi: 10.3390/min10030211.

Wicks, J. K., J. M. Jackson, and W. Sturhahn (2010), Very low sound velocities in iron-rich (Mg,Fe)O: Implications for the core-mantle boundary region, *Geophys. Res. Lett.*, 37(L15304), doi: 10.1029/2010GL043689.

Williams, Q., and E. J. Garnero (1996), Seismic Evidence for Partial Melt at the Base of Earth's Mantle, *Science*, 273, 1528-1530.

Williams, Q., J. Revenaugh, and E. J. Garnero (1998), A Correlation Between Ultra-Low Basal Velocities in the Mantle and Hot Spots, *Science*, 281, 546-549, doi: 10.1126/science.281.5376.546.

Yu, S., and E. J. Garnero (2018), Ultralow velocity zone locations: A global assessment, *G-cubed*, 19, 1-19, doi: 10.1002/2017GC007281.

Yuan, K., and B. Romanowicz (2017), Seismic evidence for partial melting at the root of major hot spot plumes, *Science*, 357, 393-397, doi: 10.1126/science.aan0760

Open OnDemand web portal for teaching

Martin Cuma, CHPC Scientific Consultant

CHPC has offered a web based access to its Linux compute and storage resources through the Open OnDemand (OOD) portal for a few years. Over time, in collaboration with the OOD developers, we have improved the interface, expanded the number of applications accessible through OOD and hardened the OOD web server.

Any CHPC user can log in to ondemand.chpc.utah.edu with their University credentials and access the portal that provides an interface to storage (home, group, scratch), cluster jobs and select interactive applications that run as cluster jobs. The interactive applications include engineering programs such as Abaqus, Ansys, or Comsol, as well as code development platforms including Matlab, RStudio and Jupyter notebook. Each interactive application runs as an independent job with dedicated resources, which is different and advantageous as compared to running these programs on cluster interactive nodes. The ease of setting up such an interactive job with OOD also contrasts with submitting an interactive job using SLURM scheduler commands in a Linux terminal.

With this in mind, we feel that the Open OnDemand portal is an attractive option for University classes and workshops, as it reduces the time the instructor needs to spend explaining how to access CHPC resources. The web browser access also eliminates user specific problems related the operating system and access tools that the student may use.

During the summer semester, in cooperation with the Department of Biomedical Informatics, we piloted a class that used OOD exclusively for its hands-on computational portion. About 30 class members used the RStudio Server OOD application to run computations up to several hours long. Apart from a few initial glitches, fixed by tuning the web server performance and adding additional hardware,

the pilot class was very successful. The students were able to quickly learn the OOD interface and the instructor was happy about not having to troubleshoot individual access issues.

With the pilot class a success, we hope to support more classes using Open OnDemand. Two OOD applications that we see as particularly good fits for class support are the above mentioned RStudio Server for R and Jupyter notebook server for Python.

The RStudio Server application in our OOD instance offers three different R installations which differ in the packages installed: Base R, Bioinformatics R and Geospatial R. We recommend basing class on one of the latter two, as they include domain specific software stacks. Students can install any missing R library in their own data space, their CHPC home directory.

The Jupyter notebook OOD application is set up to use any Python distribution installed in any user's home directory. For example, an instructor can install their own Anaconda Python, and share it with the students through the OOD Jupyter notebook app launch window option. This is a major improvement over our past Jupyter notebook offering which did not allow this as it did not tie to CHPC storage systems.

Furthermore, because the OOD application starts a cluster job, there are dedicated resources available to this job, allowing us to scale to large job/student counts. By default, we recommend using the notchpeak-shared-short, <https://www.chpc.utah.edu/documentation/guides/notchpeak.php>, account and partition although if the class is large or requires more compute resources, we can create a reservation on one of our unallocated clusters (lonepeak,kingspeak), or provide an allocation and reservation on notchpeak.

One limitation of the use of OOD is in its use with applications with graphical visualization, as most of our cluster compute nodes do not have GPUs, and even for the cluster compute nodes that do have GPUs we do not run a X server on these nodes for performance reasons. This makes visualization programs like Visit or Paraview slow to respond. If there is a demand for better visualization using OOD, we are willing to work with users/instructors on an acceptable solution.

In closing, we feel that Open OnDemand provides a great access tool for University classes and encourage instructors to give it a try. For more details look at our OOD documentation at <https://www.chpc.utah.edu/documentation/software/ondemand.php> and consult with us at helpdesk@chpc.utah.edu.

Instructors interested in using CHPC resources for a class are also encouraged to look at the article "New CHPC Class Account Provisioning" located on page 6 of this newsletter.

Python Tips

Wim R. M. Cardoen, CHPC Scientific Consultant

In this article we discuss some Python tricks. All code snippets below require Python ≥ 3.8 , and can be downloaded from:

<https://github.com/wcardoen/python-reflections>.

F-strings:

The format method for the string class has been around since Python 2.7. This is a familiar pattern for languages of which the syntax has been modelled on the C language.

```
# Example 1: format method since Python 2.7
artist = {"von Beethoven": "Bonn",
          "de Balzac": "Tours",
          "van Eyck": "Maaseik",
          "Alighieri": "Firenze"}
for person in artist.keys():
    print("{0:>13} was born in {1}").\
        format(person, artist[person])
```

The aforementioned lines results in the following output:

```
von Beethoven was born in Bonn
de Balzac was born in Tours
van Eyck was born in Maaseik
Alighieri was born in Firenze
```

In Python 3.6 the formatted string literal (**f-string**) was introduced. In an f-string the following modifications have been applied to a regular string (and its `format` method):

1. the `format` method applied to the regular string is omitted.
2. the regular string must be prexed by the 'f' or 'F' character.
3. the indices referring to variables in the format method are replaced by the variables as such.

Therefore, the introduction of the f-string makes codes shorter and more readable.

```
# Example 1: use of f-strings (iter 1)
for person in artist.keys():
    print(f"{person:>13} was born in "+
          f"{artist[person]}")
```

The format specifiers can also contain evaluated expressions:

```
# Example 1: use of f-strings (iter 2)
WIDTH = max((len(item) for item in \
            artist.keys()))
for person in artist.keys():
    print(f"{person:>{WIDTH}} was born "+
          f"in {artist[person]}")
```

Since *Python 3.8* f-strings support the '=' character for self-documenting expressions and debugging. The f-string `f"{expr=}"` will print the string `'expr='` and suffix it with the evaluated value of the expression `'expr'`.

```
# Example 2: Self-documenting expression
# (iter 1)
from math import cos, pi
print(f" {cos(pi/4.0)=}")
```

This results into the following output:

```
cos(pi/4.0)=0.7071067811865476
```

Applying a format specifier to the previous example:

```
# Example 2: Self-documenting expression using
# a format specifier (iter 2)
WIDTH=10
PRECISION=4
from math import cos, pi
print(f" {cos(pi/4.0)=: {WIDTH} . {PRECISION}}")
```

We now get:

```
cos(pi/4.0)=:  0.7071
```

For more information see <https://www.python.org/dev/peps/pep-0498/>.

Chaining of comparison operators:

Let $X := \{x_1, x_2, x_3, \dots, x_n\}$ be the set of Python expressions and $P := \{\widehat{op}_1, \widehat{op}_2, \dots, \widehat{op}_{(n-1)}\}$ be the set of Python comparison operators applied to X. Note, the Python language has the following comparison operators: `<`, `>`, `==`, `>=`, `<=`, `!=`, `is (not)`, `(not)in`.

The compound logical expression:

$x_1 \widehat{op}_1 x_2 \text{ and } x_2 \widehat{op}_2 x_3 \text{ and } \dots \text{ and } x_{(n-1)} \widehat{op}_{(n-1)} x_n$
and
 $x_1 \widehat{op}_1 x_2 \widehat{op}_2 x_3 \widehat{op}_3 \dots \widehat{op}_{(n-1)} x_n$

are equivalent in the Python language.

Example 1::

```
a, b, c=5, 3, 7
print(f" {a<b<c=} ")
# Equivalent to: (a<b) and (b<c)
# FALSE and TRUE => FALSE
```

which returns:

```
a<b<c=False
```

Example 2::

```
print(f" {15%4==3>2=} ")
# Equivalent to: (15%4==3) and (3 >2)
# or: TRUE and TRUE => TRUE
```

which returns:

```
15%4==3>2=True
```

Example 3::

```
lstA=[[1,2],["hello","world"]]
lstB=lstA
lstB[1][0]="HELLO"
lstC=lstB[:]
print(f" {lstA is lstB is lstC=} ")
print(f" {lstA == lstB == lstC=} ")
# Equivalent to: (lstA is lstB) and (lstB is
# lstC)
# or: TRUE and FALSE => FALSE
# Equivalent to: (lstA == lstB) and (lstB ==
# lstC)
# or: TRUE and TRUE => TRUE
```

which returns:

```
lstA is lstB is lstC=False
lstA == lstB == lstC=True
```

The *itertools* module:

The `enumerate` method generates the indices and values of an iterable's elements.

```
a= ['a', 'b', 'c']
b= ['x', 'y', 'z']
c= [10, 100]
# enumerate: returns indices and el.
for index, item in enumerate(b):
    print(f" El. with {index=} is "+
          f" {item=} ")
```

```
El. with index=0 is item='x'
El. with index=1 is item='y'
El. with index=2 is item='z'
```

Python's `zip` function generates an iterable of tuples which is derived from its constituent iterables. The newly created iterable ends as soon as one of its constituent iterables is exhausted.

```
item=('a', 'y', 100)
item=('b', 'y', 100)
```

The `zip` method allows the extension of `enumerate` to several iterables:

```
for item in zip(range(len(b)),b,c):
    print(f" {item=}")
```

```
item=(0, 'x', 10)
item=(1, 'y', 100)
```

Python's `itertools` module covers a wide range of functionality. The following two examples use functions which are described in <https://docs.python.org/3/library/itertools.html#module-itertools>. The function `zip_longest` generates (like `zip`) an iterable of tuples derived from its constituent iterables. The new created iterable is exhausted when its **longest** constituent iterable is exhausted. The "empty" elements are filled up with `None`.

```
import itertools as it
for item in it.zip_longest(a,b,c):
    print(f" {item=}")
```

```
item=('a', 'x', 10)
item=('b', 'y', 100)
item=('c', 'z', None)
```

The product function generates a Cartesian product iterable based on its constituent iterables.

```
s = it.product(a,b,c)
for item in s:
    print(f" {item=}")
```

```
item=('a', 'x', 10)
item=('a', 'x', 100)
item=('a', 'y', 10)
item=('a', 'y', 100)
item=('a', 'z', 10)
item=('a', 'z', 100)
item=('b', 'x', 10)
item=('b', 'x', 100)
item=('b', 'y', 10)
item=('b', 'y', 100)
item=('b', 'z', 10)
item=('b', 'z', 100)
item=('c', 'x', 10)
item=('c', 'x', 100)
item=('c', 'y', 10)
item=('c', 'y', 100)
item=('c', 'z', 10)
item=('c', 'z', 100)
```

Note that the result of the last block code can also be generated using the following tuple comprehension:

```
s= ((x,y,z) for x in a for y in b for z in c)
for item in s:
    print(f" {item=}")
```

New CHPC Class Account Provisioning

Anita Orendt, CHPC Scientific Consultant

Recently, CHPC improved the process of provisioning CHPC accounts for students in classes that need access to CHPC resources. Previously the instructor of any class needing access to CHPC resources needed to send in a request with the class name (for example CHEM 1234) along with the class roll. Using this information, both a CHPC group for the class with the instructor as the CHPC PI and the individual user accounts for the students in the class were manually created. The new method replaces the manual process of creating the user accounts with a token-based authorization process. We tested this process with a class over the summer semester.

Under the new method, class instructor still needs to request the creation of the class account (preferably via a request sent to helpdesk@chpc.utah.edu), along with an estimation of the number of students expected to enroll in the class. In response, CHPC will then set up the class group with the instructor as the CHPC PI for the account as before. We will then provide the instructor, via email, the class token along with an account creation link. A token expiration date is also set, which is the last date the token can be used to generate new class accounts. The instructor should share this information with the students in the class. With the new provisioning process instructors no longer need to contact CHPC if the class roster changes (such as new students being added to the class).

Each student must then use the token and link provided by the instructor to obtain a CHPC account as part of the class group. The use of the link and the token will automatically create the student's CHPC account or, in the case of a student who already has an existing CHPC account, add the student to the class CHPC group.

When the student account is created, the instructor is sent a notification. In addition, to reduce chances of misuse of the token (e.g., a student in a class may give the token to a friend who creates CHPC account, despite not being in the class), the instructor is also provided with a link that will show them a list of class students accounts. The instructor should periodically check the names in this e-mail against the class roster until the token expiration date is reached, and notify CHPC if there is a person not in the class that has been added to the class group.

A few weeks after the class is over, CHPC will reach out to the instructor about removing the CHPC accounts created for the class.

As always if you would like additional information or have any questions please contact us at helpdesk@chpc.utah.edu.

Date	Time	Presentation	Presenters
Fri, August 28	1-2PM	Overview of CHPC	Anita Orendt
Tues & Wed, Sept 1-2	9AM-3PM	XSEDE HPC Workshops: MPI	XSEDE Webcast
Weds, Sept 9	1-3PM	Hands on Introduction to Linux, Part 1	Brett Milash Wim Cardoen
Fri, Sept 11	1-3PM	Hands on Introduction to Linux, Part 2	Brett Milash Wim Cardoen
Mon, Sept 14	1-3PM	Hands on Introduction to Linux, Part 3	Brett Milash Wim Cardoen
Wed, Sept 16	1-3PM	Hands on Introduction to Linux, Part 4	Wim Cardoen Brett Milash
Fri, Sept 18	1-2PM	Modules Basics	Anita Orendt
Mon, Sept 21	1-2PM	Slurm and Slurm Batch Scripts	Anita Orendt
Wed, Sept 23	1-3PM	Hands on Introduction to Open OnDemand	Martin Cuma
Fri, Sep 25	1-3PM	Hands on Introduction to Python, Part 1	Brett Milash Wim Cardoen
Mon, Sept 28	1-3PM	Hands on Introduction to Python, Part 2	Brett Milash Wim Cardoen
Wed, Sept 30	1-3PM	Hands on Introduction to Python, Part 3	Brett Milash Wim Cardoen
Fri, Oct 2	1-3PM	Numpy, Part 1 (Hands on Introduction to Python, Part 4)	Brett Milash Wim Cardoen
Mon, Oct 5	1-3PM	Numpy, Part 2 (Hands on Introduction to Python, Part 5)	Brett Milash Wim Cardoen
Wed, Oct 7	9AM-3PM	XSEDE HPC Workshops: Big Data and Machine Learning	XSEDE Webcast
Fri, Oct 9	1-2PM	National and Regional Compute Resources	Anita Orendt
Mon, Oct 12	1-2PM	OSG w/ information about the newchpc osg submit	Wim Cardoen
Wed, Oct 14	1-3PM	Workflows Using Snakemake	Brett Milash
Fri, Oct 16	1-3PM	Nextflow – NEW!	David Chen Brett Milash
Mon, Oct 19	1-3PM	Introduction to Parallel Computing	Martin Cuma
Weds, Oct 21	1-2PM	Introduction to Programming with MPI	Martin Cuma
Fri, Oct 23	1-2PM	Introduction to Programming with OpenMP	Martin Cuma
Mon, Oct 26	1-2PM	Hybrid MPI-OpenMP Programming	Martin Cuma
Weds, Oct 28	1-2PM	GPU Programming	Wim Cardoen
Fri, Oct 30	1-2PM	Introduction to I/O at CHPC	Martin Cuma
Tues, Nov 3	9AM-3PM	XSEDE HPC Workshop : OpenMP	XSEDE Webcast
Weds, Nov 4	1-2PM	Introduction to Profiling	Martin Cuma
Fri, Nov 6	1-2PM	Introduction to Debugging	Martin Cuma
Mon, Nov 9	1-3PM	Introduction to Containers	Martin Cuma
Weds, Nov 11	1-3PM	Introduction to R	Wim Cardoen Brett Milash
Fri, Nov 13	1-2PM	Data Management Tips	Anita Orendt Brett Milash
Mon, Nov 23	1-3PM	Using Git for Version Control	Martin Cuma
Mon, Nov 30	1-2PM	Overview of Protected Environment	Anita Orendt Wayne Bradford

The University of Utah
University Information Technology
Center for High Performance Computing
155 South 1452 East, Room 405
SALT LAKE CITY, UT 84112-0190

Thank you for using CHPC resources!

Welcome to CHPC News!

If you would like to be added to our mailing list, please provide the following information and via the contact methods described below.

Name:

Phone:

Email:

Department
or Affiliation:

Address:
(campus
or U.S. mail)

Please acknowledge the use of CHPC resources!

If you use CHPC computer time or staff resources, we request that you acknowledge this in technical reports, publications, and dissertations. An example of what we ask you to include in your acknowledgments is:

“A grant of computer time from the Center for High Performance Computing is gratefully acknowledged.”

If you make use of the CHPC Protected Environment, please also acknowledge the NIH shared instrumentation grant:

“The computational resources used were partially funded by the NIH Shared Instrumentation Grant 1S10OD021644-01A1.”

Please submit copies or citations of dissertations, reports, pre-prints, and reprints in which CHPC is acknowledged in one of the following ways:

Electronic responses

By email: helpdesk@chpc.utah.edu

By fax: (801) 585-5366

Paper responses

By U.S. mail: 155 South 1452 East, Rm 405
Salt Lake City, UT 84112-0190

By campus mail: INSCC 405