Bits & Bytes

New Director of the MPCDF

Hermann Lederer, Erwin Laure

As of March 2020, the MPCDF has a new director. Erwin Laure took over from Hermann Lederer, who occupied the role of Acting Director since Stefan Heinzel passed away in August 2018. Laure joins the MPCDF from the Royal Institute of Technology (KTH) in Stockholm, Sweden, where he led the national supercomputing center, PDC, for the past 12 years. After his studies in Vienna, Austria and a short postdoc period there, he joined the European Organization for Nuclear Research (CERN) in Geneva, Switzerland, where he was contributing for 8 years in leading positions to the development of the distributed computing infrastructure for LHC and other scientific domains.

"I'm very excited joining the Max Planck Society and especially the MPCDF, one of the leading institutions in high-performance computing and academic data management in Europe", Laure said. "High-performance computing and data analysis are today indispensable tools for a growing number of scientific domains. Demands on the HPC and storage infrastructure as well as for application support are ever growing. New developments in hardware, algorithms, scientific workflows, and the ever growing data deluge make today a very interesting, yet challenging time. I'm looking forward working with the excellent team at the MPCDF and the scientists from the various institutes on innovative solutions especially ensuring that the MPCDF remains a reliable and flexible partner in tackling the computing, storage, and data management challenges the institutes are facing."

Erwin Laure

High-performance Computing

Hermann Lederer, Renate Dohmen, Markus Rampp, Sebastian Ohlmann,

Extension of MPG HPC resources: Raven

The newest Max Planck supercomputing system Cobra was installed in 2018 with some GPU extensions added last year. The further expansion of the system has now been contracted after another competitive European procurement process. The system called Raven will comprise two different compute architectures to best address future HPC needs: a pure CPU segment based on compute nodes with future Intel Xeon CPUs, and a complementary GPU segment with next generation Nvidia Tesla GPUs. The comparatively large GPU segment will have 192 compute nodes with four GPUs each, expected to be installed by end of 2020. Requests for application support for porting to GPUs are welcome (contact: Markus Rampp). The CPU segment will start as an interim system in mid 2020, followed by an upgrade to a new processor type in early 2021.

GPU development partition on Cobra

A GPU development partition "gpudev" with a short turnaround time has been created on the set of V100-GPU nodes of Cobra to allow users test and optimize their GPU codes in an easy and time-saving manner. In order to use the "gpudev" partition you have to specify

```
#SBATCH --partition=gpudev
```

in your submit script. The maximum number of nodes available with "gpudev" is 1, the usage is always exclusive, the maximum execution time is 15 minutes, and you can choose to use one or two GPUs like on the normal gpu_v100 partitions by specifying

```
#SBATCH --gres=gpu:v100:2
or
#SBATCH --gres=gpu:v100:1
```

Intel MKL: using only part of the functionality

The Intel Math Kernel Library (MKL, https://software.intel.com/en-us/mkl) is a comprehensive numerical library that offers lots of functionality optimized for x86 CPUs – BLAS, LAPACK, SCALAPACK, FFT and much more. MKL is generally available on our systems as a module. When linking with MKL, all of its functionality is available to the calling program. Sometimes, however, it is desirable to use only part of the functionality: suppose one wants to use the fftw3 library instead of MKL for computing FFTs, e. g., for comparing the performance or because MKL does not provide the full fftw3 functionality. When using the standard MKL, this is not possible: it depends on the run-time behavior, but usually all functionality such as fftw3 will be used from MKL.

The solution to this problem is to build custom shared object files (i. e. a subset of all MKL routines) which can be linked to the application. This can be achieved by using the makefiles provided in '$MKLROOT/tools/builder' which will build shared object files that contain only functions specified in a text file. Predefined lists for BLAS, LAPACK and SCALAPACK exist that make it easy to extract only this functionality (for more information, see the Intel MKL developer guide).

As an example, the BLAS and LAPACK functionality can be extracted with:

```
$ cp -r $MKLROOT/tools/builder .
$ cd builder
$ make libintel64 export=blas_example_list \
    name=libmkl_blas
$ make libintel64 export=lapack_example_list \
    name=libmkl_lapack
```

This creates the shared object files 'libmkl_blas.so' and 'libmkl_lapack.so' that can be linked to an application using '-lmkl_blas -lmkl_lapack'.

Jupyter Notebook as a service

Michele Compostella, Klaus Reuter

Introduction

The visualization team at the Max Planck Computing and Data Facility has developed a novel service that will allow effortless and secure connection to Jupyter Notebooks running on various HPC systems (currently Draco, Cobra and Talos), with direct access to all software and data stored on the HPC systems. This new service integrates and expands the remote visualization service recently deployed on the Draco and Cobra clusters, offering more web-based solutions to develop, compute and analyse results on HPC systems.

Usage

In order to configure, start and connect to the service, users should login to https://rvs.mpcdf.mpg.de from a web browser that supports HTML5 (we recommend to use Mozilla Firefox or a Chrome-based browser) and authenticate with their Kerberos user name and password.

Three options are available:

1. **Initialize Remote Visualization**:
This step is only required the first time a user wants to access the remote visualization or the Jupyter Notebook services on a specific HPC system. Note that **users are required to re-initialize** their remote visualization password on the machines in order to start using the new Jupyter Notebook services. The selected password will then be used for both remote visualization and Jupyter Notebook sessions.
– Select the desired machine in the drop-down menu corresponding to *Machine*.
– Filling the web form, the user can set the VNC and Jupyter Notebook password which is required to connect to running sessions later. We encourage users to select a password different from their Kerberos' one and with a minimum of 8 characters, following common password policies.
– In the section *Default modules for JN session* the user

can specify the name of any module that should be loaded before starting all the Jupyter Notebook sessions. This list is stored in a file located in the home folder of the user (`$HOME/.jupyter/modules.conf`). For example, in order to have some machine learning modules pre-loaded to each Jupyter Notebook session, a user can write

```
gcc/6
cuda/10.1
cudnn/7.6.2
nccl/2.4.8
impi/2019.4
tensorflow/gpu/2.1.0
```

in the text box to have the respective modules immediately available at the start of each session. Note that additional modules can be loaded from a running Jupyter notebook, if needed (see the **Troubleshooting** section).

2. **Submit new session**:
This page provides the form to request a remote desktop, remote visualization or Jupyter notebook session.
– Select the desired machine in the drop-down menu corresponding to *Machine*.
– Select the desired *Session Type* among the following:

- *Remote desktop*: suitable for efficient use of GUI tools (e. g. VTune, Matlab) and for activities that do not require intensive calculations.

- *Remote visualization*: suitable for GPU-accelerated calculations and rendering using tools like VisIt or Paraview.

- *Jupyter*: suitable for standard data analysis and computation using Jupyter Notebooks.

- *Jupyter for machine learning*: suitable for GPU-accelerated machine-learning applications using Jupyter Notebooks.

Note that some types of sessions are not available on all clusters (for example, the *Remote desktop* is currently available only on the Draco cluster). Depending on the *Machine* and on the selected *Session Type*, different resources are available for each session. The information icon next to the session type describes the exact resources allocated to each session.
– Select a *Session Length*.
– Depending on the *Session Type*, select the screen *Resolution* of the remote session (for *Remote desktop* and *Remote visualization* sessions) or the *Interface* and *Software* (for *Jupyter* and *Jupyter for machine learning*). Two values for the *Interface* are available: a *Classic* interface for standard Jupyter Notebook sessions and a *Lab* interface for users that prefer the Jupyter Lab layout. In the *Software* menu, it is possible to specify if the Anaconda3 package used for the Jupyter sessions should be based on Python only or should include the R language.

3. **Connect to session**:
This page provides information about the status of submitted remote visualization and Jupyter notebook sessions, including the possibility to cancel a session or connect to a running one.
– Clicking the *Show Sessions* button, all the submitted and running sessions will be displayed.
– Select *Cancel* to cancel running sessions on the clusters and *Connect* to access the requested session. Note that sessions are opened in a new browser tab and the password that has been specified during step 1 should be entered to finally have access to the session directly from the web browser.

Manual submission

Experienced users may want to submit their Jupyter Notebook sessions directly from the login node of the cluster. In this case, after the login on the machine, the user should first initialize just once the password for the remote visualization and Jupyter Notebook services using

```
$ module load rvs
$ module load anaconda/3
$ setup_rvs
$ setup_jn
```

and following the instructions prompted. This will create a password for both the remote visualization and the Jupyter Notebook services. Also, the commands create the folders `$HOME/rvs` and `$HOME/.jupyter` in the home directory of the user (with an automatic backup of previously existing folders with the same name). As an example, to start a 4-hour long interactive Jupyter Notebook session, the user can then simply run the following commands:

```
$ module load rvs
$ cd $HOME/rvs
$ sbatch --time=04:00:00 $RVS_HOME/bin/jn.cmd
```

Please, substitute `jn.cmd` with `jlab.cmd` for the Jupyter Lab interface. As soon as the Jupyter session starts on a compute node, the user will receive a notification e-mail with instructions on how to connect via a standard web link, or via the new web interface.

Troubleshooting

In the following we present a list of the most common questions and problems users may encounter, with a suggestion on how to solve them.

1. *The cluster where I want to submit my job does not appear in the list of available machines.*
The remote visualization and Jupyter Notebook services

are currently supported only on Draco, Cobra and Talos. If a cluster name does not appear in the drop-down menus, you may not be registered as an official user of that cluster. Please check that you can login directly via ssh to the cluster. If this is the case, but you still don't see the cluster entry, please contact the MPCDF support at support@mpcdf.mpg.de.

2. *I don't remember my password for connecting to a session.*
If you forgot your password or if you want to change the list of default modules available in your Jupyter Notebook session, you can re-initialize your remote visualization and Jupyter notebook sessions as described in step 1 of the **Usage** section. The new initialization will create a backup of your `$HOME/rvs` (containing the session logs) and `$HOME/.jupyter` folders before setting the new password for you.

3. *I want to load a module from my Jupyter Notebook session.*
You can list, load and unload modules available on the cluster directly from your Jupyter Notebook session using:

```
module('list')
module('load','module_name')
module('unload','module_name')
```

Note, however, that inside a Jupyter Notebook session the command

```
module('purge')
```

will not unload all modules, but will restore the Jupyter Notebook to the initial state instead (with all the modules that were loaded when the notebook was first launched). Please note that for technical reasons it is not possible to module-load some Python modules that set the `LD_LIBRARY_PATH` variable from within a Jupyter session. Please put such environment modules into the list of *Default modules for JN session* to module-load them before the session starts.

4. *The State of my session is PENDING for a long time.*
Usually, your session should start within a few seconds from the submission. If your session shows the PENDING status for a long time, all the available hardware resources may already be in use. You can try submitting a new session to a different cluster, or select a session that requires less resources. For example, selecting a *Remote desktop* instead of a *Remote visualization* session or a *Jupyter* session instead of a *Jupyter for machine learning* one increases the chances that a slot for your session is more readily available.

5. *I have a "503 Service Unavailable" error when I try to connect to the session.*
When a Jupyter Notebook session is started on a cluster, it may take a few seconds for the Jupyter Notebook server to be ready and accept your connection. In case you see this error message, please wait a few seconds and then try again connecting to your session. If the problem persists minutes after the start of the session, please contact the MPCDF support at support@mpcdf.mpg.de.

FTP decommissioning

John Alan Kennedy

After over 25 years of continuous and very fruitful service the anonymous FTP server at the MPCDF is about to be decommissioned. The FTP service itself has been kept operational as a legacy service for the past few years. However the functionality it offers, and much more, is provided by the ownCloud-based datashare service at the MPCDF. Numerous FTP users/projects have already been migrated to the Datashare service. As such the anonymous FTP service is about to be decommissioned in favor of the datashare service.

The datashare service is an ownCloud-based file sharing service which is very rich in functionality and has an intuitive web interface. The service currently provides users with a 200 GB quota and offers numerous options for sharing and publishing data.

In this article we will highlight the Anonymous FTP use-cases which can now be covered with the datashare service. We will focus on:

- Anonymous data sharing

- Anonymous upload

Both data upload and sharing may be enabled by using the "Public Link" functionality in the Datashare service. This allows you to create a link to a folder which you may then share with external collaborators. There are three options available:

1. Download/View

2. Download/View/Upload

3. Upload Only (File Drop)

Options 1 and 3 most directly mimic the Anonymous FTP service.

Anonymous Data Sharing may be enabled by creating a folder and enabling a public link with Download/View.
Anonymous Data Upload may be enabled by creating a folder and enabling a public link with Upload Only.

Figure 1 shows the creation of a public link for an upload folder. On link creation a data owner simply needs to click the fitting option for the folder to enable anonymous upload or download.
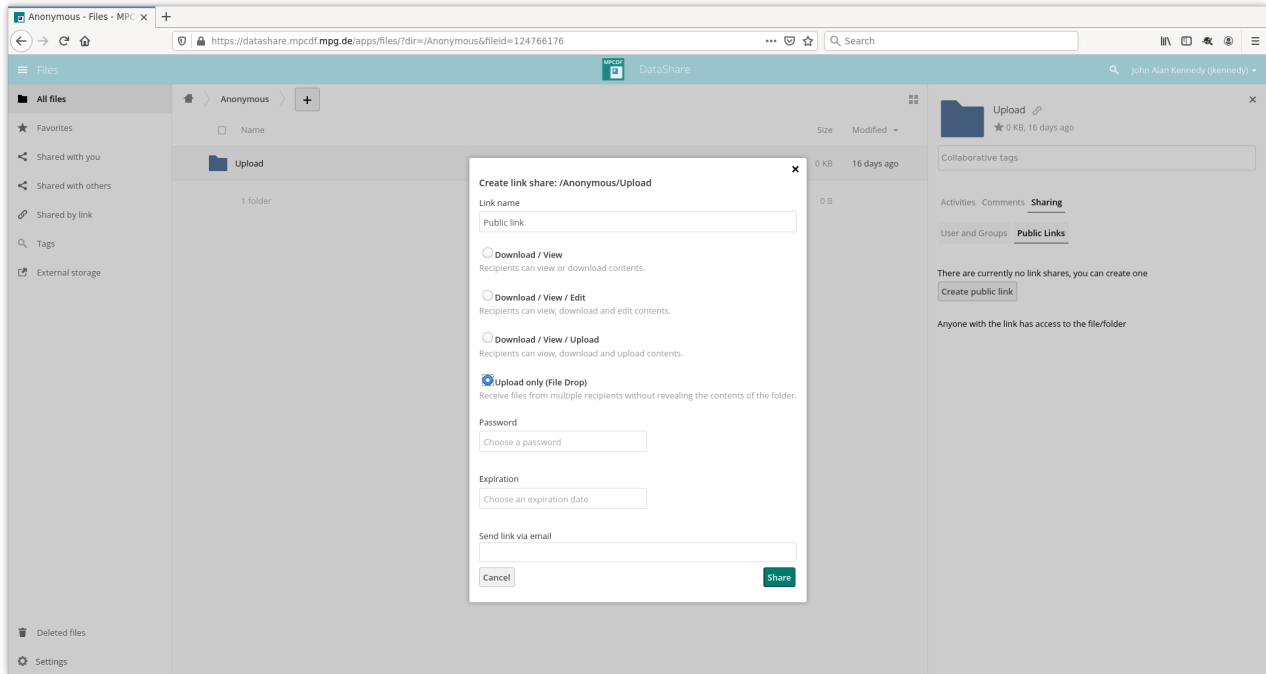

Fig. 1: Creation of an anonymous upload folder via the File Drop public link option

Data which is anonymously uploaded will be accounted for as part of your own datashare quota.

Note that you can also create a password and/or add an expiration date for the link. Once the link has been created it can easily be shared with collaborators via e-mail.

Sharing data with users who have an MPCDF account can be achieved either by following the above "Public Link" sharing method or alternatively by directly sharing the data with the user.

More information about the datashare service can be found on the MPCDF web pages: `https://www.mpcdf.mpg.de/services/data/share`.

Specific information about public links can be found in the ownCloud User Manual.

Tips & Tricks

Florian Kaiser, Andreas Schott

Changing the password, take care against lockout

Before setting a new password, please make sure that you have access to all devices such as mobile phones where the old password might be saved for wifi, mail access, or on batch systems where it may be stored with "save-password". Assure such access attempts to be inactive and update the password in all affected places to avoid your account becoming temporarily locked by too many authentication attempts using still the old password.

Missed the password change, no problem

In case you did not change your password before it had expired, there is an easy way to help yourself: the password-recovery page, which you can find in the Userspace on our homepage. Give it a try. `https://www.mpcdf.mpg.de/userspace/recovery`.

Ecosystem Data Management

Raphael Ritz

How can the steadily growing amounts of digital data from different areas be used sustainably in a systemtical manner? Since October 1, 2019, the project "Ecosystem Data Management: Analyzes - Recommendations - FAIRification" (Ecosystem Data Management - EcoDM), funded by the Federal Ministry of Education and Research (BMBF) with 1.7 million Euros, has been dealing with this question.

For a funding period of 30 months, the Max Planck Society represented by the Max Planck Computing and Data Facility (MPCDF), the German Archaeological Institute (DAI), the Helmholtz Open Science Office located at the Helmholtz Center Potsdam - GFZ German Research Center for Geosciences and the Potsdam University of Applied Sciences address this. What is special: EcoDM wants to find out how the potential of rapid digital data growth can be systematically made available. Starting point is the question how to realize and put into practice the FAIR principles (Findable, Accessible, Interoperable, Reusable). Science is a global undertaking. The current establishment of a European Open Science Cloud and the national research data infrastructure in Germany will have far-reaching consequences for scientific practice in the digital age as a whole, which go well beyond the necessary standardizations in research data management. The Max Planck Society is contributing to these developments in the interests of the excellence of basic research in Germany in an international context.

The handling of research data in science is only one of four areas that the four project partners will examine in the next two years. Also on the agenda is how to handle data in the areas of public sector, economy and qualification in a sustainable way. The MPCDF coordinates the project and leads the activities in the area of science.

News & Events

Tilman Dannert, Hermann Lederer, Raphael Ritz

Third Advanced HPC Workshop

After all the positive feedback from the last Advanced HPC Workshop, the MPCDF will organize a similar event from November 23-26, 2020 in Garching at the MPCDF. Compared to the issue in 2019 the 2020 workshop will have a stronger focus on GPU programming and will specifically address the new HPC system Raven to be deployed at the MPCDF later this year, and therefore the schedule is extended by another day. The main topics are

- Software engineering for scientific software
- Debugging serial and parallel HPC codes
- Serial and parallel profiling of HPC codes
- Porting codes to GPUs and usage of the relevant tools.

The workshop will start around noon on Monday, November 23, with two and a half days of lectures given by HPC application experts of the MPCDF, Intel, and Nvidia, followed by a day dedicated to hands-on sessions for a number of selected projects on Thursday. The registration and further details of the workshop will be announced in the next issue of Bits&Bytes.

HPC Summer School 2020

The eleventh International HPC Summer School planned to take place in Toronto, Canada from July 12-17, 2020 has been cancelled as an in-person event due to the corona virus pandemic. There are preliminary plans to host the school next year in Toronto, in July 2021. Eventually a subset of the planned materials for IHPCSS 2020 might be offered as a virtual event in July 2020.

RDA-Deutschland Conference 2020

From February 25-27, 2020 almost 200 people met for the German conference of the Research Data Alliance in Potsdam at the GeoForschungsZentrum. Like last year one of the focus areas was the forthcoming National Research Data Infrastructure (NFDI). In addition, the FAIR principles (Findable, Accessible, Interoperable, Reusable) and activities towards their realization received a lot of attention. This year's event saw for the first time a series of satellite events including a Data Carpentry course. Once more this event – organized under the leadership of the MPCDF – has turned out to be an attractive and well-received forum for the German research data community.