

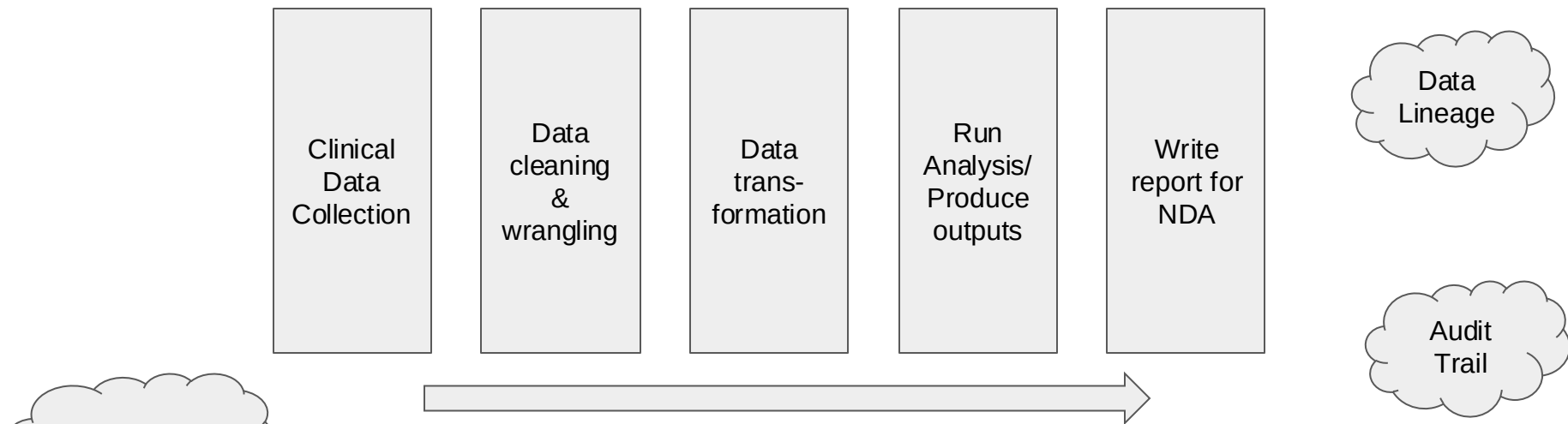


SLURM + Open Source Clinical Workflows

Current State

Michael Mayer & Phil Bowsher

Simplified Clinical Development Workflow



Any output used in the final report for the NDA must be run in a way that it can be traced back all the way to the initial clinical data collected as part of the clinical trial

What is SLURM ?

SLURM stands for “Simple Linux Utility for Resource Management” and is a modern & popular Resource Manager and Scheduler used on High Performance Computing(HPC) systems.

A user that wants to run some computations can request resources (such as CPU, memory, GPUs etc..). SLURM will take care of that request and allocate the resources once they become available.



Open source Workflow software



Selected use cases for leveraging HPC / SLURM

- Clinical trial design
- General Parameter studies
- Bootstrap
- PK/PD Analysis
- Genomic data analysis
- ...

All of the above are integrated in the overall clinical workflow leading up to a NDA.

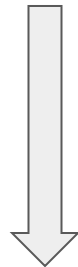
For the following slides, we will exclusively focus on the use of SLURM with R

Tool of choice to develop open source code

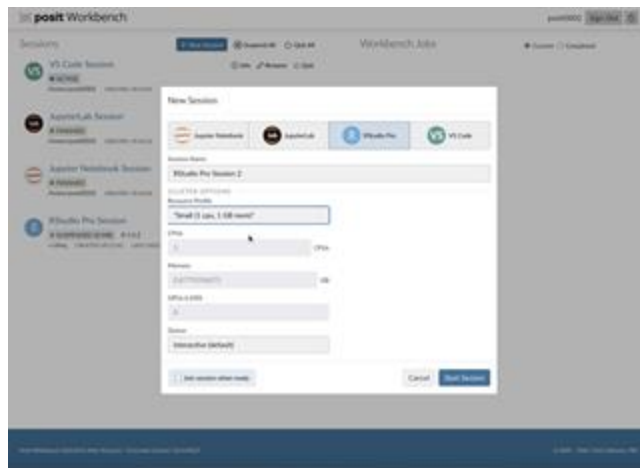
Posit Workbench can be configured to run interactive sessions (e.g. RStudio IDE) in a SLURM job.

The full compute power of a SLURM cluster can be leveraged by

- **Workbench jobs:** Offload long-running compute tasks in a separate SLURM job, all launched from within the IDE session (RStudio IDE and VS Code)
- **Interactive computing:** Write code that uses R packages that submit additional jobs/tasks to the SLURM cluster from within the IDE
- **Batch computing:** Write standalone R code and submit them using the SLURM CLI



Technical complexity



posit™ Workbench

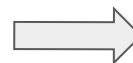
High-Level overview



conversion



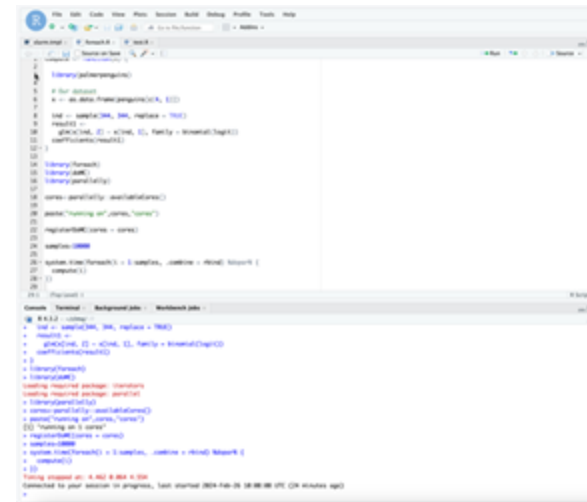
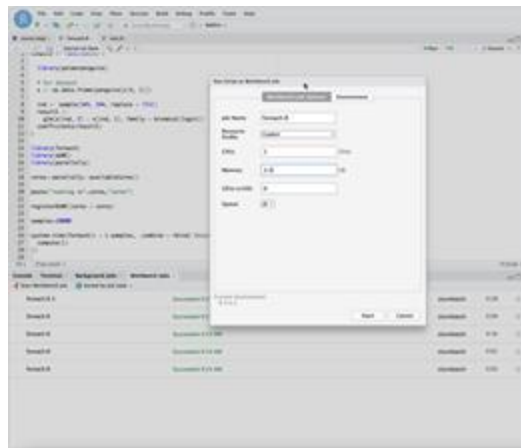
interactive Sessions

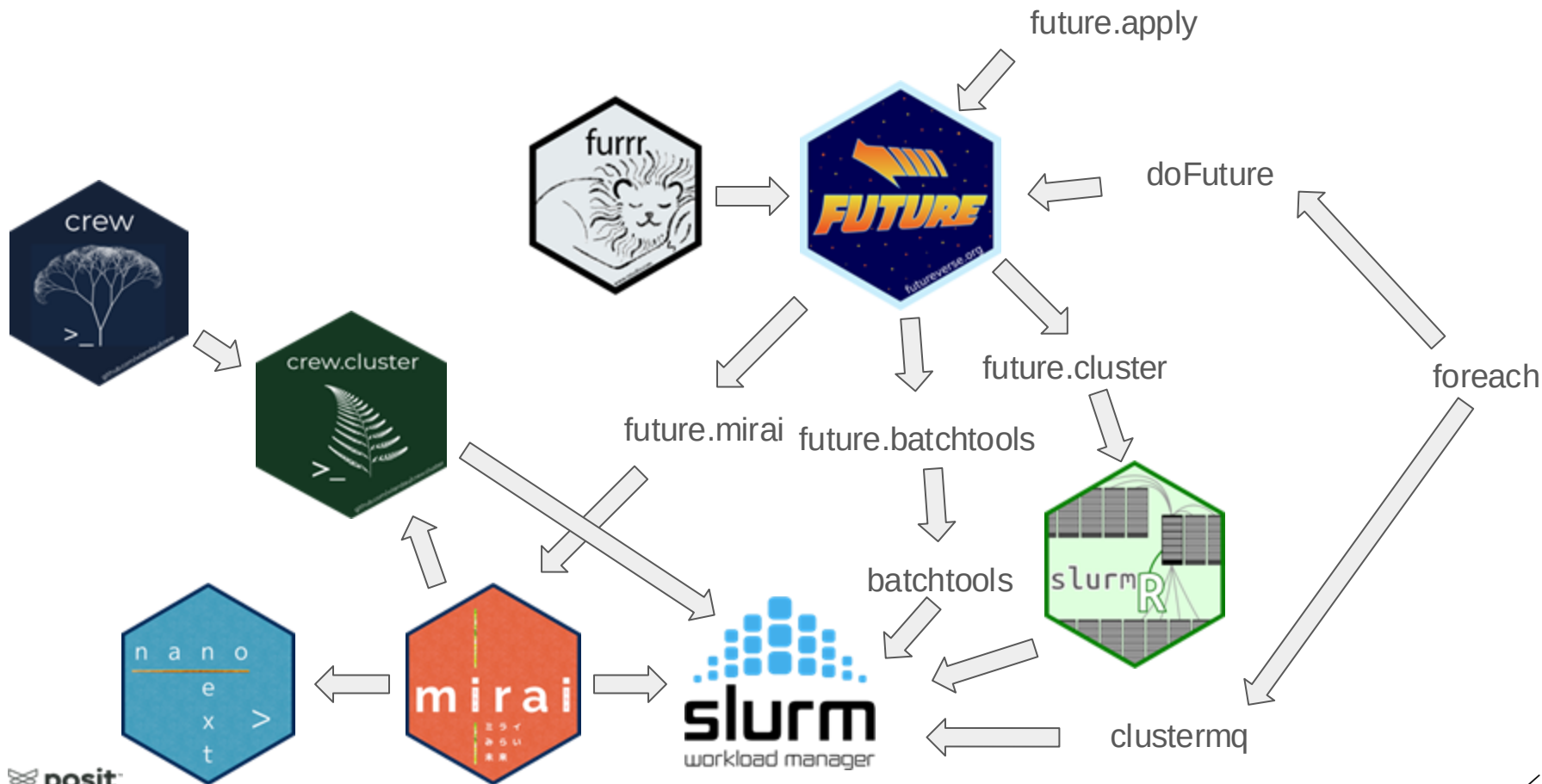


Compute backend(s)



Workbench jobs





Containers on HPC

On HPC clusters, alternative technologies to docker are being used



- Apptainer/Singularity are most popular HPC container technologies
- Integration into SLURM is possible via SLURM's SPANK architecture
- Building containers can be very time consuming
 - In order to benefit from docker layers, some users build docker containers first and then convert them to singularity/apptainer images
 - Building directly from singularity definition files works equally well, in particular in environments where the container builder does not have elevated permissions

Efficient building of containers - package installation



`install.packages()` is sequential (slow)

Use [pak](#):



- Downloads, compiles and installs packages in parallel
- Automatically detects and installs OS dependencies
- Creates a lock file for reproducibility



Pip install is sequential (slow)

Use [uv](#)



- 10-100x faster than pip
- Uses requirements.txt
- Has many more features (e.g. can install python versions)

Efficient building of containers



`install.packages()` is sequential (slow)

Use [pak](#):



- Downloads, compiles and installs packages in parallel
- Automatically handles OS dependencies
- Creates a local cache for reproducibility



Pip install is sequential (slow)

Use [uv](#)



- 10-100x faster than pip

A customer wanted to install 1500 R packages in a container.

With `install.packages()` this took 24 hours to build.

When moving to pak it finished the build in less than an hour.

(e.g. can

Opportunities & Conclusions

- There is exciting, modern and efficient R packages available to use SLURM based HPC's in the context of clinical workflows.
- Posit Workbench and its use of Workbench jobs can help to reduce the entry barrier for users unfamiliar/uneasy with linux command line tools
- Containers are becoming popular on HPC clusters as well due to the increased need for reproducibility.
- Container technologies on HPC clusters are different due to the different execution model.
- With the use of the right tools, container build times can be optimized.