

# Running R scripts on CSC's Taito supercluster

## Kylli Ek, CSC

CSC, Espoo, 26.2.2018



*CSC – Suomalainen tutkimuksen, koulutuksen, kulttuurin ja julkishallinnon ICT-osaamiskeskus*

# Reasons for using CSC computing resources

- Computing something takes more than 2-4 hours
- Need for more memory
- Very big datasets
- Keep your desktop computer for normal usage, do computation elsewhere
- Need for a server computer
- Need for a lot of computers with the same set-up (courses)
- Free for Finnish university users / will be free for state research institutes



# CSC HPC resources

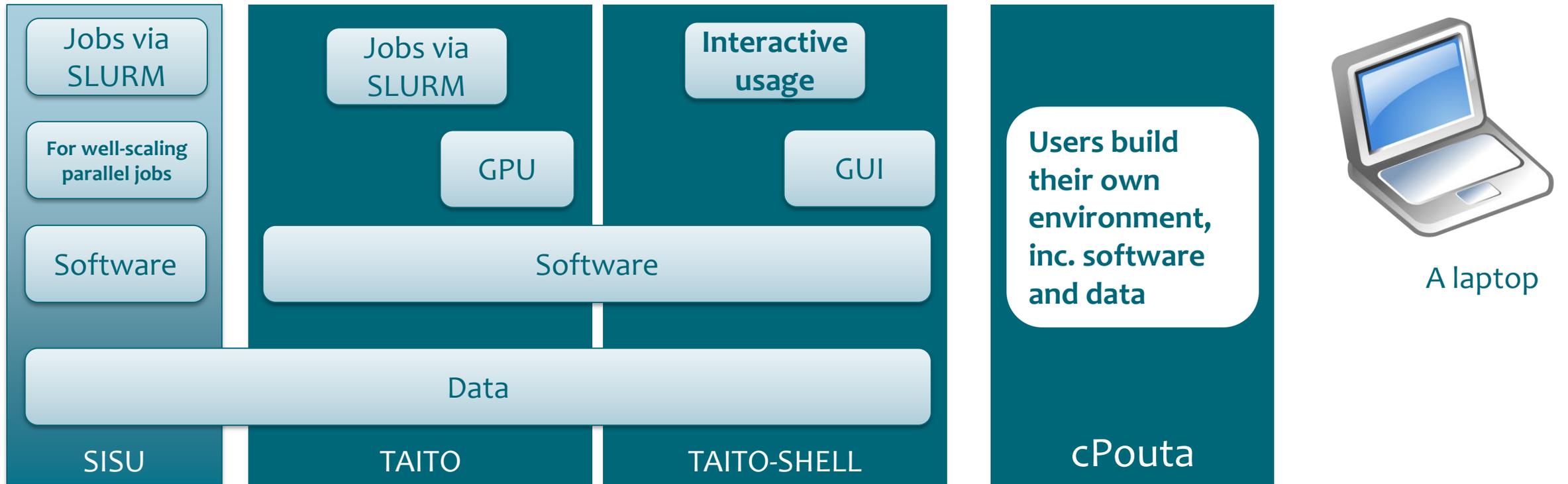
Max  
9600 cores

Max 672 cores /  
1536 GB memory

Max 4 cores /  
128 GB memory

Max 48 cores /  
240 GB memory

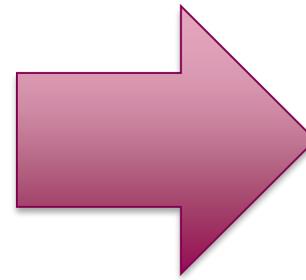
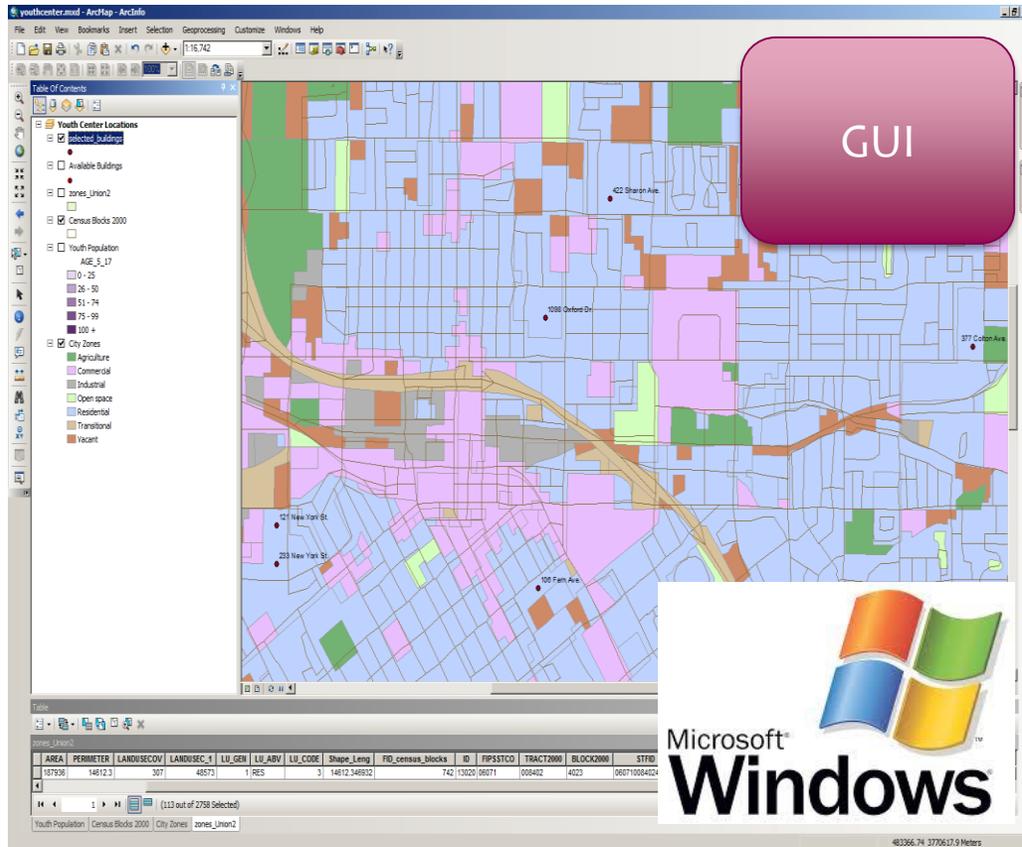
2-4 cores /  
8 GB memory



# Realistic expectations

- A single core of a CSC machine is about as fast as one of a basic laptop.
- It has just a lot of them.
- .. and more memory and faster input-output.
  - Just running your single core script at CSC does not make it much faster.
  - For clear speed-ups you have to use several cores.
  - ... or optimize your script.

# The keys to geocomputing: Change in working style & Linux



ArcGIS, QGIS, ...

R, Python, shell scripts, Matlab, ...

# Using different GIS-software in Taito

	Bash	R	Python	QGIS
GDAL	x	x	x	x
GRASS	x	x	(x)	x
LasTools	x	(x)	(x)	x
SagaGIS	x	x	(x)	x
Taudem	x	(x)	(x)	?
R spatial packages	-	x	-	-
Python geo packages	-	-	x	-

- Full list of GIS software: <https://research.csc.fi/software> -> Geosciences
- You can install software also yourself, inc. Python packages

# Rspatial

- Documentation: <https://research.csc.fi/-/rspatial-env>
- Installed packages:
  - **R** (3.4.1) with spatial packages: geoR, geoRglm, geosphere, ggmap, grid, gstat, GWmodel, mapproj, maptools, ncdf4, RandomFields, raster, rgdal, rgeos, rgrass7, RSAGA, sf, sp, spacetime, spatial, spatial.tools, spatstat, spdep, strucchange.
  - rgdal - [GDAL/OGR](#) (2.2.1), [Proj4](#) (4.9.3)
  - rgeos - GEOS (3.6.1)
  - RSAGA - [Saga GIS](#) (2.1.4, there is also a new version of SagaGIS on Taito, but currently RSAGA supports only this older version, therefore that version is loaded here)
  - rgrass7 - [GRASS GIS](#) (7.2.0, without GUI)

# RStudio

- RStudio is available in Taito-shell
- If using NoMachine connection, use from right menu RStudio for GIS
- Otherwise:

```
module load rspatial-env  
module load rstudio  
rstudio
```

# Shared data area in Taito

- Hosts large commonly used datasets
- Reduces the need to transfer data to Taito
- Located at /proj/ogiir-csc/
- All Taito users have read access.
- Only CSC personnel have write access.
- For data with open license
  
- If you think some other dataset should be included here, ask from servicedesk@csc.fi

[https://research.csc.fi/gis\\_data\\_in\\_taito](https://research.csc.fi/gis_data_in_taito)

## MML:

Lidar point cloud data

Dem 2m (see virtual rasters section below)

Dem 10m (see virtual rasters section below)

## FMI

10km avg relative humidity

10km avg sea level pressure

10km daily max temperature

10km daily mean temperature

10km daily min temperature

10km daily precipitation

10km daily radiation

10km daily snow

10km monthly mean temperature

10km monthly precipitation

## LUKE

Multi-source national forest inventory

# Module system

- Tool to set up your environment
  - Load libraries, adjust path, set environment variables
  - Needed on a server with hundreds of applications and several compilers etc.
- Example: initialize R with GIS packages (and some other GIS software)

```
$ module load rspatial-env
```

# Batch system

- Has to be used on Taito (not in Taito-shell)
- Optimizes resource usage by filling the server with jobs
- You have to reserve time, cores and memory for your job
- You have to write a batch job script
- You can use array jobs for pseudo-parallel analysis (multiple files, different scenarios, different time steps etc)
- <https://research.csc.fi/taito-batch-jobs>

## Example: steps for running your Python script in Taito-shell

(0. Get yourself CSC user account)

1. Move your data and scripts to Taito (with FileZilla).
2. Connect to Taito (with Putty or NoMachine).
3. Load rspatial-env module and start R or open RStudio.
4. Check which R packages do you need and if they are available in Taito.  
\* If needed, install it yourself or ask CSC - [servicedesk@csc.fi](mailto:servicedesk@csc.fi).
5. Fix the paths of your input/output files.
6. Test your script in Taito-shell with some test data.
7. Run your scripts with all data interactively on Taito-shell or in Taito as batch job.
- (8. Make use of several cores using multiprocessing package in your Python code.)

## Example code in CSC training Github

- Examples for doing spatial analysis in CSC computing environment with:
  - Python
  - R
- Examples include also batch job scripts suitable for Taito.
- Some of the examples include samples for serial, array and parallel jobs.

<https://github.com/csc-training/geocomputing>



## More info and support

Taito user guide: <https://research.csc.fi/taito-user-guide>

Geocomputing general info: <http://research.csc.fi/geocomputing>

GeoPython documentation: <https://research.csc.fi/-/rspatial-env>

Geocomputing course materials: [https://www.csc.fi/web/training/-/geocomputing\\_2017](https://www.csc.fi/web/training/-/geocomputing_2017)

Kylli Ek, +358 50 38 12 838

[servicedesk@csc.fi](mailto:servicedesk@csc.fi)