# TTK = the Topology ToolKit

https://topology-tool-kit.github.io

- Topological analysis of multi-dimensional scalar functions
- Open source, development since 2014, first public release in 2017
- Interfaces:
  - pure C++
  - VTK/C++ (∼3X shorter code)
  - ParaView Python (∼5X shorter code) and ParaView plugin
- Installing TTK natively would require to:
  1. install very specific versions of dependencies
  2. download and patch ParaView source code + compile ParaView
  3. compile TTK: not as straightforward as suggested in the documentation, quite a few extra flags
  - the entire process in MacOS took me 3 days to figure out

# TTK = the Topology ToolKit

https://topology-tool-kit.github.io

- Topological analysis of multi-dimensional scalar functions
- Open source, development since 2014, first public release in 2017
- Interfaces:
  - pure C++
  - VTK/C++ (∼3X shorter code)
  - ParaView Python (∼5X shorter code) and ParaView plugin
- Installing TTK natively would require to:
  1. install very specific versions of dependencies
  2. download and patch ParaView source code + compile ParaView
  3. compile TTK: not as straightforward as suggested in the documentation, quite a few extra flags
  - the entire process in MacOS took me 3 days to figure out
- Fortunately, you can run a pre-built Docker image with pvserver+TTK on local port 11111 and connect to it from a local client

# TTK = the Topology ToolKit

https://topology-tool-kit.github.io

- Topological analysis of multi-dimensional scalar functions
- Open source, development since 2014, first public release in 2017
- Interfaces:
  - pure C++
  - VTK/C++ (∼3X shorter code)
  - ParaView Python (∼5X shorter code) and ParaView plugin
- Installing TTK natively would require to:
  1. install very specific versions of dependencies
  2. download and patch ParaView source code + compile ParaView
  3. compile TTK: not as straightforward as suggested in the documentation, quite a few extra flags
  - the entire process in MacOS took me 3 days to figure out
- Fortunately, you can run a pre-built Docker image with pvserver+TTK on local port 11111 and connect to it from a local client
- For client-server work on an HPC cluster you would have to patch+compile pvserver+TTK on the cluster
  - alternatively, you can use a pvserver+TTK Singularity container on the cluster for your client-server workflow

# Running client-server ParaView + TTK via Singularity on Cedar

1. Find the “pvserver+TTK” Docker image with its tag (version number)
   (a) log in to `https://hub.docker.com`
   (b) search for ‘topologytoolkit‘
   (c) select one you like, click on Container
   (d) click on Tags
   (e) copy the tag following the “docker pull” command on the right
2. Build “ParaView 5.9.1 + TTK 0.9.9 in Ubuntu 20.04” image on Cedar
   `https://docs.computecanada.ca/wiki/Singularity`

```
$ cd ~/scratch
$ wget https://raw.githubusercontent.com/moby/moby/master/contrib/download-frozen-image-v2.sh
$ sh download-frozen-image-v2.sh build topologytoolkit/ttk:5.9.1-0.9.9   # download an image
                                                                         # from Docker Hub into build/
$ cd build && tar cvf ../ttk.tar * && cd ..
$ salloc --mem-per-cpu=3600 --cpus-per-task=2 --time=0:15:0 --account=...
$ module load singularity
$ singularity build topologytoolkit591.sif docker-archive://ttk.tar   # build the Singularity image;
                                                                      # wait for ‘Build complete‘
$ /bin/rm -rf build ttk.tar
```

- alternatively, TTK source code includes scripts for building Docker and Singularity images from scratch, but it is much easier to start with a precompiled container!

# Running client-server ParaView + TTK via Singularity on Cedar (cont.)

2. Start ParaView server on a compute node

```
$ cd ~/scratch
$ salloc --mem-per-cpu=3600 --time=0:30:0 --account=...
$ module load singularity
$ singularity run -B /home -B /scratch topologytoolkit591.sif pvserver
```

3. Connect from the ParaView client on your computer

```
$ ssh cedar.computecanada.ca -L 11111:cdr___:11111
start ParaView 5.9.x on your computer and connect to localhost:11111
File | Load State - local ~/ttk/ttk-data-0.9.8/states/dragon.pvsm
Choose File Names - remote cedar:/scratch/razoumov/dragon.vtu
```