

# Data Transfer in HPC

UManitoba Spring 2026  
High-Performance Computing And Cloud  
Workshop

Stefano Ansaloni

University of Manitoba

May 20, 2026



**University  
of Manitoba**



**Digital Research  
Alliance** of Canada

# Stefano Ansaloni

Cloud Computing Specialist at University of Manitoba  
(part of the Advanced Research Computing team)

Software Developer and DevOps Specialist since 2017

Linux User/Admin since 2005

# Tools for transferring data

## Terminal user interface:

- ▶ SCP   
- ▶ RSync  
- ▶ Duck   

## Graphical user interface:

- ▶ WinSCP 
- ▶ Cyberduck  

## Web user interface:

- ▶ Globus 

## Secure Copy Protocol – SCP

The command **scp** is part of the OpenSSH suite and it behaves like the **cp** command.

It can copy a local file to a remote host:

```
scp local_file remote_user@remote_host:/path/to/remote_file
```

It can copy a remote file to the local pc:

```
scp remote_user@remote_host:/path/to/remote_file local_file
```

It is possible to specify a SSH private key as login credentials for the remote host:

```
scp -i ssh_private_key ...
```

# Secure Copy Protocol – SCP

## Advantages

- ▶ Usually pre-installed on Unix systems
- ▶ Available for a wide range of operative systems
- ▶ Simple to use

## Drawbacks

- ▶ Transfer resume not supported

## Remote Sync – RSync

The command **rsync** can be used to synchronize single files, or entire directories (either local, or remote).

When synchronizing files between hosts that allow SSH access, *RSync* does support encapsulation of the data stream in an SSH channel.

Example synchronizing a local directory to a remote host (trailing slashes “/” matter):

```
rsync --delete -AXSHax \  
    local_dir/ \  
    remote_user@remote_host:path/to/remote_dir/
```

# Remote Sync – RSync

## Advantages

- ▶ Usually pre-installed on Unix systems
- ▶ Properly synchronize directories
- ▶ Can resume interrupted transfers

## Drawbacks

- ▶ Slightly more complicated than *SCP*
- ▶ Requires extra options for controlling underlying SSH session

# Duck

*Duck* is the command line version of *Cyberduck*.

It is a server/cloud storage browser with support for multiple protocols (SFTP, S3, ...).

It can download files or directories from a remote host:

```
duck --download sftp://remote_user@remote_host/path/to/remote_file \  
            local_file
```

It can upload files or directories to a remote host:

```
duck --upload sftp://remote_user@remote_host/path/to/remote_file \  
            local_file
```

It is possible to specify a SSH private key as login credentials for the remote host:

```
duck -i ssh_private_key ...
```

## Advantages

- ▶ Properly synchronize directories
- ▶ Can resume interrupted transfers
- ▶ Can parallelize connections for a single transfer
- ▶ Supports multiple protocols

# WinSCP

*WinSCP* is probably the most common choice for Windows users that want a graphical user interface to move files between their local computer and a remote host.

Its interface can be configured as “2-panes” (similar to Norton Commander), or as “single pane” (similar to Windows File Explorer).

## Advantages

- ▶ Look&Feel similar to other Windows applications
- ▶ Can resume interrupted transfers (only for some protocols)
- ▶ Supports multiple protocols

## Drawbacks

- ▶ Only available for Windows

# Cyberduck

*Cyberduck* is probably the most common choice for Mac users that want a graphical user interface to move files between their local computer and a remote host.

It is a server/cloud storage browser with support for multiple protocols and a fairly easy interface.

## Advantages

- ▶ Properly synchronize directories
- ▶ Can resume interrupted transfers
- ▶ Can parallelize connections for a single transfer
- ▶ Supports multiple protocols

## Drawbacks

- ▶ Only available for Windows and Mac

# Globus

*Globus* is a service for fast, reliable, secure file transfer.

Designed specifically for researchers, it has an easy-to-use web interface with background monitoring features that automate the management of file transfers.

*Globus* transfers data between two so-called “Globus Endpoints” or “Data Collections”.

Since the data can reside across different organizations, it provides a way to manage and link “identities” between these organizations to facilitate data transfers.

## Advantages

- ▶ Web interface (no desktop client needed)
- ▶ Automatically tuning transfer settings
- ▶ Automatically resume interrupted transfers

## Drawbacks

- ▶ Transfers only between “Globus Endpoints” (“Data Collections”)

## Final thoughts

The tools presented are only few of those available online.

You must comply with any regulations or compliance policies that come with the data you are transferring (i.e. do not transfer data not belonging to you, or that contains information covered by any privacy act).

**Avoid** blindly downloading/running tools obtained from some obscure internet website.

YOU WANT YOUR COUSIN TO SEND YOU A FILE? EASY.  
HE CAN EMAIL IT TO— ... OH, IT'S 25 MB? HMM...

DO EITHER OF YOU HAVE AN FTP SERVER? NO, RIGHT.  
IF YOU HAD WEB HOSTING, YOU COULD UPLOAD IT...

HMM. WE COULD TRY ONE OF THOSE MEGASHAREUPLOAD SITES,  
BUT THEY'RE FLAKY AND FULL OF DELAYS AND PORN POPUPS.

HOW ABOUT AIM DIRECT CONNECT? ANYONE STILL USE THAT?

OH, WAIT, DROPBOX! IT'S THIS RECENT STARTUP FROM A FEW  
YEARS BACK THAT SYNCs FOLDERS BETWEEN COMPUTERS.  
YOU JUST NEED TO MAKE AN ACCOUNT, INSTALL THE—



OH, HE JUST DROVE  
OVER TO YOUR HOUSE  
WITH A USB DRIVE?

UH, COOL, THAT  
WORKS, TOO.

I LIKE HOW WE'VE HAD THE INTERNET FOR DECADES,  
YET "SENDING FILES" IS SOMETHING EARLY  
ADOPTERS ARE STILL FIGURING OUT HOW TO DO.

Questions?

Thank you