

Related blog post:

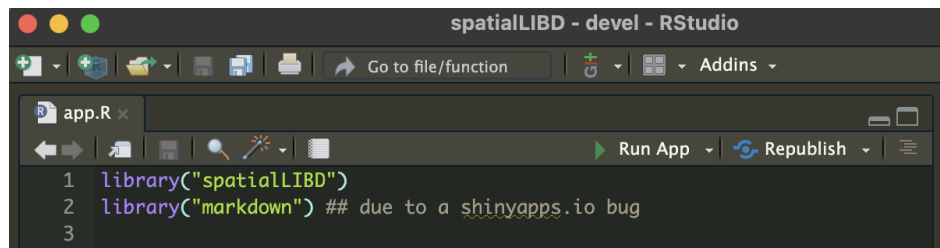
<https://colladotor.github.io/2025/03/14/initial-impressions-from-testing-posit-connect/>

## [2025-02-28] Deploying R shiny apps with Posit Connect

R packages for today:

```
if (!requireNamespace("remotes", quietly = TRUE)) {  
  install.packages("remotes")  
}  
remotes::install_cran("rsconnect")  
remotes::install_cran("BiocManager")  
BiocManager::install("spatialLIBD")  
BiocManager::install("iSEE")
```

- shinyapps.io examples
  - <https://github.com/LieberInstitute/spatialLIBD/blob/devel/app.R> deploys <http://libd.shinyapps.io/spatialLIBD>
  - [https://github.com/LieberInstitute/spatialDLPFC/tree/main/code/deploy\\_app\\_k09](https://github.com/LieberInstitute/spatialDLPFC/tree/main/code/deploy_app_k09) has the code for [https://libd.shinyapps.io/spatialDLPFC\\_Visium\\_Sp09](https://libd.shinyapps.io/spatialDLPFC_Visium_Sp09)
- They are deployed to shinyapps.io with:
  - `rsconnect::deployApp()` like [https://github.com/LieberInstitute/spatialDLPFC/blob/53fcbf53ec3e03ac41083812695d62b678ea57e4/code/deploy\\_app\\_k09/deploy.R#L17C1-L33](https://github.com/LieberInstitute/spatialDLPFC/blob/53fcbf53ec3e03ac41083812695d62b678ea57e4/code/deploy_app_k09/deploy.R#L17C1-L33)
  - Or RStudio's "publish" button when you are working on an `app.R` R script



- Configuration at shinyapps.io
  - Done via <https://www.shinyapps.io/admin/#/dashboard>
  - You need an account (username, password) and a personal access token

- Typically we use the **8 GB** “instances” (computers from Amazon), which is the **maximum memory available**.

APPLICATION 7781053 - SPATIALDLPFC\_VISIUM\_SP09

Overview

Metrics

URLs

Settings

Users

Logs

Restart

Archive

Delete

General

Advanced

Application

Instance

Instance Size

3X-Large (8 ▾)

Select larger instances if you require more memory for your application. Free and starter plans are limited to a maximum of 1GB (Large). [upgrade now!](#)

Instance Idle Timeout

15 min

Time after which your idle application will be stopped. The minimum is 5 minutes.

- Advanced settings

Worker Settings

Max Worker Processes

1

Number of worker processes that can be started in a single instance.

Max Connections

2

Number of concurrent connections allowed per worker process.

Worker Load Factor

5 %

Threshold percentage after which a new connection will trigger the addition of a new worker process (limited to the Max Worker Processes limit).

Connection Timeout

900 sec

Seconds of inactivity before a browser connection to a worker process is considered to be idle and is closed.

Read Timeout

3600 sec

Seconds of browser to worker inactivity after which the connection is considered to be idle and is closed. Use a value of 0 for non-interactive applications, e.g. a dashboard.

Startup Timeout

300 sec

Time that an instance will wait for a worker to start. Increase this number if you load a lot of data or have a longer application startup time. The maximum is 300 seconds.

Idle Timeout

60 sec

Wait time in seconds before an idle worker process with no connections is shut down. The value must be between 5 and 60 seconds.

Instance Settings

Instance Load Factor

50 %

Threshold percentage after which a new connection will trigger the addition of an application instance (limited to the Maximum Instance Limit, free tier is 1).

Start Count

2

Number of instances to bring up when an application starts. Pick a higher number if you know that your application receives flash traffic.

Build Settings

Package Cache

☒

Enable/Disable the use of package caching when your application is built. Caching allows for faster deploy times, disable it only if you suspect that the build is failing due to caching.

- 
- The 8GB gets divided by the number of processes (independent R sessions), then each R session can take a number of connections. We can also have up to 10 instances per app (so 10 machines).

- Posit Connect deployment

- `rsconnect::deployApp()` like [https://github.com/LieberInstitute/dlpfc\\_asd/blob/bb26998db9c15fd78c73667fbe7fe5b90c6ea59f/code/03\\_spatialLIBD\\_app/deploy.R#L32-L41](https://github.com/LieberInstitute/dlpfc_asd/blob/bb26998db9c15fd78c73667fbe7fe5b90c6ea59f/code/03_spatialLIBD_app/deploy.R#L32-L41)
- Or RStudio's "publish" button when you are working on an `app.R` R script
- Or the new option: a Manifest file created with `rsconnect::writeManifest()` like [https://github.com/LieberInstitute/spatialDLPFC/blob/53fcbf53ec3e03ac41083812695d62b678ea57e4/code/deploy\\_app\\_k09/deploy.R#L35-L48](https://github.com/LieberInstitute/spatialDLPFC/blob/53fcbf53ec3e03ac41083812695d62b678ea57e4/code/deploy_app_k09/deploy.R#L35-L48)
- Or the even newer option: via a GitHub repo that is getting synched automatically
  - Needs a manifest file + a "GitHub app" (which an admin from both Posit Connect and from GitHub's org needs to set up)
  - Example with a small repo: [https://github.com/LieberInstitute/spatialDLPFC\\_Visium\\_Sp09\\_pseudobulk-](https://github.com/LieberInstitute/spatialDLPFC_Visium_Sp09_pseudobulk-) where the manifest is made with [https://github.com/LieberInstitute/spatialDLPFC\\_Visium\\_Sp09\\_pseudobulk-/blob/16bffe312cd4363a3e6d38dfa7b8f33a65ab6636/deploy.R#L23-L26](https://github.com/LieberInstitute/spatialDLPFC_Visium_Sp09_pseudobulk-/blob/16bffe312cd4363a3e6d38dfa7b8f33a65ab6636/deploy.R#L23-L26) that leads to [https://conn1.libd.org/spatialDLPFC\\_Visium\\_Sp09\\_pseudobulk\\_small\\_repo/](https://conn1.libd.org/spatialDLPFC_Visium_Sp09_pseudobulk_small_repo/)
  - Cannot use soft links like [https://github.com/LieberInstitute/spatialDLPFC/blob/main/code/deploy\\_app\\_k09/spe\\_subset\\_for\\_spatialLIBD.rds](https://github.com/LieberInstitute/spatialDLPFC/blob/main/code/deploy_app_k09/spe_subset_for_spatialLIBD.rds) since only the local directory where the `app.R` and the `manifest.json` files exist gets deployed, not other parts of the repo
- More tests at [https://github.com/LieberInstitute/Posit\\_Connect\\_shiny\\_apps](https://github.com/LieberInstitute/Posit_Connect_shiny_apps)
  - [https://github.com/LieberInstitute/Posit\\_Connect\\_shiny\\_apps/tree/devel/spatial\\_hpc/code](https://github.com/LieberInstitute/Posit_Connect_shiny_apps/tree/devel/spatial_hpc/code) for [https://conn1.libd.org/spatial\\_hpc/](https://conn1.libd.org/spatial_hpc/)
  - [https://github.com/LieberInstitute/Posit\\_Connect\\_shiny\\_apps/tree/devel/spatial\\_hpc/code/02\\_stitched\\_app](https://github.com/LieberInstitute/Posit_Connect_shiny_apps/tree/devel/spatial_hpc/code/02_stitched_app) for [https://conn1.libd.org/spatial\\_hpc\\_jhpce/](https://conn1.libd.org/spatial_hpc_jhpce/)
  - [https://github.com/LieberInstitute/Posit\\_Connect\\_shiny\\_apps/tree/devel/spatialDLPFC\\_mdd\\_bpd/code/02\\_sparseMatrix\\_in\\_memory\\_version](https://github.com/LieberInstitute/Posit_Connect_shiny_apps/tree/devel/spatialDLPFC_mdd_bpd/code/02_sparseMatrix_in_memory_version) for [https://conn1.libd.org/spatialDLPFC\\_mdd\\_bpd\\_sparseMatrix/](https://conn1.libd.org/spatialDLPFC_mdd_bpd_sparseMatrix/)
- **GitHub repositories get checked automatically every 15 minutes**, though manual updates can also be triggered.
  - **Best to actually have the files on a small repo.**
  - Let's do a live test with [https://github.com/LieberInstitute/spatialDLPFC\\_Visium\\_Sp09\\_pseudobulk-](https://github.com/LieberInstitute/spatialDLPFC_Visium_Sp09_pseudobulk-) and [https://conn1.libd.org/spatialDLPFC\\_Visium\\_Sp09\\_pseudobulk\\_small\\_repo/](https://conn1.libd.org/spatialDLPFC_Visium_Sp09_pseudobulk_small_repo/)
  - Posit Connect servers can be configured such that users have ssh access, and thus can upload data separately from the R code

- **This greatly helps update apps faster, particularly when the data hasn't changed**
  - See `ls -lh`  
`/r_data/lcollado/Posit_Connect_shiny_apps`
- Manifest files contain info about all the packages you have in your computer, which then Posit Connect will use too.
  - R packages installed from GitHub have the commit "sha" twice like [https://github.com/LieberInstitute/Posit\\_Connect\\_shiny\\_apps/blob/d138b04cf56c46efa4164bf131e22ceae47b6408/spatialDLPFC\\_mdd\\_bpd/code/02\\_sparseMatrix\\_in\\_memory\\_version/manifest.json#L4926](https://github.com/LieberInstitute/Posit_Connect_shiny_apps/blob/d138b04cf56c46efa4164bf131e22ceae47b6408/spatialDLPFC_mdd_bpd/code/02_sparseMatrix_in_memory_version/manifest.json#L4926) and [https://github.com/LieberInstitute/Posit\\_Connect\\_shiny\\_apps/blob/d138b04cf56c46efa4164bf131e22ceae47b6408/spatialDLPFC\\_mdd\\_bpd/code/02\\_sparseMatrix\\_in\\_memory\\_version/manifest.json#L4930](https://github.com/LieberInstitute/Posit_Connect_shiny_apps/blob/d138b04cf56c46efa4164bf131e22ceae47b6408/spatialDLPFC_mdd_bpd/code/02_sparseMatrix_in_memory_version/manifest.json#L4930)
  - This means that you can update the manifest.json file, git push, and up to 15 min later (or manually if you log in to the Posit Connect admin interface) the app will get updated
  - See [https://github.com/LieberInstitute/Posit\\_Connect\\_shiny\\_apps/blob/devel/global/01\\_update\\_manifest\\_files.R](https://github.com/LieberInstitute/Posit_Connect_shiny_apps/blob/devel/global/01_update_manifest_files.R) for a small script that locates all the `"*write_manifest.R"` (each of them written using `here::here()`) and updates all the `manifest.json` files.
    - **This is MUCH easier for updating many apps to use the latest software available. Almost no time was spent doing this!**
- Posit Connect configuration
  - User access:

Info

Access

Runtime

Schedule

Tags

Vars

SHARING

Anyone - no login required

All users - login required

Specific users or groups

Who can view or change this application

L

lcollado lcollado

Remove All Groups and Users

INTEGRATIONS

No integrations can be assigned because none have been created yet.

To create integrations, please use the [Integrations page](#) . Only an administrator can create integrations.

CONTENT URL

Path

/spatialDLPFC\_Visium\_Sp09\_pseudobulk\_si

URL

https://conn1.libd.org/spatialDLPFC\_Visium\_Sp09\_pseudobulk\_small\_repo/

Copy

LOCKING

Unlocked

Has more options for sharing, like user groups

- Has similar options to shinyapps.io for app control, though you can also set a maximum RAM (since all your apps are on the same Posit Connect server). The **Posit Connect server we tested on had 31 GB of RAM, 200 GB of disk.**



Info



Access



Runtime



Schedule



Tags



Vars

## PROCESS SETTINGS



Min processes

0 *no minimum**enter override*

Max processes



3

*enter override*

Max connections per process



20

*enter override*

Load factor



0.5

*enter override*

## TIMEOUT SETTINGS



Initial timeout (seconds)



60

300



Idle Timeout per process (seconds)



5

60



Connection timeout (seconds)



3600

*enter override*

Read timeout (seconds)



3600

*enter override*

## PROCESS EXECUTION



Who runs this content on the server



The default user rstudio-connect



## RAM SETTINGS



Max RAM (GiB)

0 *no limit**enter override*

- As an admin, you can also check all the R processes that are running on the server, stop some, check their logs, check their memory use (here I think that the reported memory is higher than the memory use reported at shinyapps.io).

The screenshot shows the Posit Connect Admin interface. The 'Processes' tab is selected, displaying a table with one process running on the host 'conn1'.

Hostname	Content Name	Type	User	CPU	RAM	Started	Actions
conn1	spatialDLPFC_mdd_bpd	Shiny application	rstudio-connect	1.00	2.0 GiB	5 minutes ago	[Stop] [Refresh]

- HDF5Array

- In theory uses a lot less memory as the data is served through HDF5 (on disk)
- In practice, `spatialLIBD` apps don't seem to work well with `HDF5Array`.
- Compared [https://github.com/LieberInstitute/Posit\\_Connect\\_shiny\\_apps/tree/devel/spatialDLPFC\\_mdd\\_bpd/code](https://github.com/LieberInstitute/Posit_Connect_shiny_apps/tree/devel/spatialDLPFC_mdd_bpd/code) vs [https://github.com/LieberInstitute/Posit\\_Connect\\_shiny\\_apps/tree/devel/spatialDLPFC\\_mdd\\_bpd/code/02\\_sparseMatrix\\_in\\_memory\\_version](https://github.com/LieberInstitute/Posit_Connect_shiny_apps/tree/devel/spatialDLPFC_mdd_bpd/code/02_sparseMatrix_in_memory_version) (about 9.5 GB in RAM for the object).
- Aka [https://conn1.libd.org/spatialDLPFC\\_mdd\\_bpd](https://conn1.libd.org/spatialDLPFC_mdd_bpd) vs [https://conn1.libd.org/spatialDLPFC\\_mdd\\_bpd\\_sparseMatrix](https://conn1.libd.org/spatialDLPFC_mdd_bpd_sparseMatrix).

The screenshot shows the Posit Connect Admin interface with two processes listed in the 'Processes' tab.

Hostname	Content Name	Type	User	CPU	RAM	Started	Actions
conn1	spatialDLPFC_mdd_bpd	Shiny application	rstudio-connect	0.99	2.0 GiB	17 minutes ago	[Stop] [Refresh]
conn1	spatialDLPFC_mdd_bpd_sparseMatrix	Shiny application	rstudio-connect	0.00	15.0 GiB	27 minutes ago	[Stop] [Refresh]

- The one using `HDF5Array` takes 4 min 45 seconds to load on my laptop (just the app), then clicking on the "spot-level" tab takes another 3 min 48 seconds or so before anything is shown. After that, it's all fast except if you change from `logcounts` (default) to `counts` which triggers another 3 min wait. On the Posit Connect server it sometimes doesn't load (it almost always doesn't; it loaded for me once but by the time I noticed, the app had timed out).
- EDIT: <https://github.com/LieberInstitute/spatialLIBD/commit/b80d92c3271a6ad92859f79a3bc343f77bad9bf2> fixed the issues with `HDF5Array` on `spatialLIBD`. We could deploy 120 capture areas with `lowres` images and both `counts` and `logcounts` assays using a max of 2.9 GB of RAM from what I saw reported by Posit Connect.

posit Connect Content People Integrations Admin Documentation

Metrics Mail Settings Tags Audit Logs Scheduled Content System Checks Processes Queue

Processes

Hostname	Content Name	Type	User	CPU	RAM	Started	Actions
conn1	DLFFC_asd_raw	Shiny application	rstudio-connect	0.00	4.5 GB	33 minutes ago	
conn1	spatialDLFFC_mdd_bpd	Shiny application	rstudio-connect	0.00	2.9 GB	9 minutes ago	

- while 2.9 max GB of RAM would indicate that we could deploy the apps on shinyapps.io with the 8 GB RAM limit there, the `HDF5Array` files typically are larger than **the max 3 GB file limit imposed on shinyapps.io** as documented at <https://docs.posit.co/shinyapps.io/guide/applications/>

- Stitched data also worked
  - See <https://conn1.libd.org/spatialAmygdala/> with code from [https://github.com/LieberInstitute/Posit\\_Connect\\_shiny\\_apps/tree/devel/spatialAmygdala/code](https://github.com/LieberInstitute/Posit_Connect_shiny_apps/tree/devel/spatialAmygdala/code)
  - I saw it reach about 15.7 GB at some point during my tests. Screenshot is for 12.3 GB. Data is 5.54 GB estimated with `lobstr::obj_size()`.

posit Connect Content People Integrations Admin Documentation

Metrics Mail Settings Tags Audit Logs Scheduled Content System Checks Processes Queue

Processes

Hostname	Content Name	Type	User	CPU	RAM	Started	Actions
conn1	spatialAmygdala	Shiny application	rstudio-connect	0.65	12.3 GiB	7 minutes ago	