

# 10 Minutes to Data Science: Transitioning Between RAPIDS cuDF and CuPy Libraries | NVIDIA Developer Blog

---

 <https://developer.nvidia.com/blog/10-minutes-to-data-science-transitioning-between-rapids-cudf-and-cupy-libraries/>

Nick Becker

Sun Jun, 20 23:50



## 10 Minutes to Data Science: Transitioning Between RAPIDS cuDF and CuPy Libraries

*This post was originally published on the RAPIDS AI Blog.*



[RAPIDS](#) is about creating bridges, connections, and clean handoffs between GPU PyData libraries. Interoperability with functionality is our goal. For example, if you’re working with [RAPIDS cuDF](#) but need a more linear-algebra oriented function that exists in [CuPy](#), you can leverage the interoperability of the GPU PyData ecosystem to use that function. Just like you can do with NumPy and pandas, you can weave cuDF and CuPy together in the same workflow while keeping the data entirely on the GPU.

The 10-minute notebook series called “[10 Minutes to cuDF and CuPy](#)” was formed to help encourage this interoperability. This is an introductory notebook that explains how easy it is to transition between the two libraries if your workflow can benefit from it. In this tutorial, we show how the [CUDA Array](#) and [DLPack](#) interfaces allow us to share our data between cuDF and CuPy in microseconds. This gives us near-instant access to the best of both libraries.

Sorry, something went wrong: [Reload?](#)

Sorry, we cannot display this file.

Sorry, this file is invalid so it cannot be displayed.

How impactful is GPU-accelerated array processing? Turns out, for many operations on large arrays you can get [more than a 100x speedup](#) using CuPy on the GPU compared to NumPy on the CPU. With that much horsepower at your fingertips for both dataframe and array based workflows, cuDF and CuPy can fundamentally change the way data science is done and how you work.

Want to get started with RAPIDS and CuPy? Check out [cuDF](#) and [CuPy](#) on Github and let us know what you think! You can download pre-built Docker containers for our latest release from [NVIDIA NGC](#) or [Dockerhub](#) to get started or install it yourself via [Conda](#). Need something even easier? You can quickly get started with [RAPIDS in Google Colab](#) and try out all the new things we've added with just a single push of a button. Don't want to wait for the next release to use upcoming features? You can download our nightly containers [from Dockerhub](#) or install via [Conda](#) to stay at the tip of our development branch.

## About the Authors