



QuiltData.com
contact@quiltdata.io
+1-415-323-0193

Empowering Quants to trade faster From Excel files to data packages

Case Study GFS Bank

GFS Bank is a Fortune 1000 corporation in financial services with over 1.5 trillion USD in assets under management. In the interest of client privacy and time-to-publication, "GFS Bank" is a proxy for the company name.

Problem Forecasting and trading delayed by manual data prep

GFS purchases data assets in a variety of file formats for use in predictive modeling. The assets are then distributed to more than 20 quants in five subsidiaries spread across geographies, security groups, and development platforms.

The root cause of GFS's problem was a labor-intensive approach to data management. Data engineers were preoccupied with storing files in blob storage and posting links to a shared wiki. Broken links, corrupt files, and divergent file versions were commonplace. Quants were preoccupied with manually browsing the wiki, downloading the data, and then extracting the data into a variety of modeling tools, including Python, R, and Excel.

The result was a **common anti-pattern in data science**, shown in Fig. 1. As a result, forecasting and trading were delayed by data prep. In an industry where milliseconds mean the difference between profit and loss, such delays are problematic.

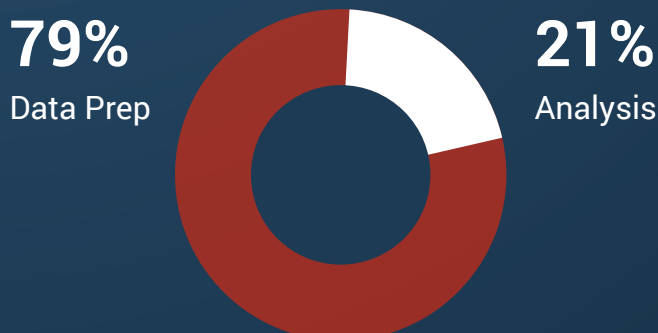


Figure 1 - A common failure mode in data science: more time spent on data prep than on analysis. Forbes.com, "Cleaning Big Data: Most Time-Consuming, Least Enjoyable Data Science Task, Survey Says," 2016.

Solution Quilt data packages automate data prep

GFS Bank was able to distribute the first data assets to its Quants within 24 hours of installing Quilt. With minimal further engineering time, Quilt can scale to serve the bank's entire Quantitative asset pipeline.

Figure 2 shows how data flows through Quilt to Quants at GFS.



GFS purchases data assets for use in predictive modeling by Quants.



Data assets occur in a variety of file formats, including Excel and CSV.



`quilt build` automatically packages file assets so they're versioned, tracked and deployable across platforms.



`quilt push` securely publishes a package to the Quilt registry.

The Quilt registry is a secure, compliant **data router** where Quants can get the latest, most accurate data.



`quilt install` distributes the latest data. Each install is authenticated and logged.

Quants get right to modeling in Jupyter, trading, and making money. There's no messy data prep.

Trading 10X Faster Measuring the impact of Quilt

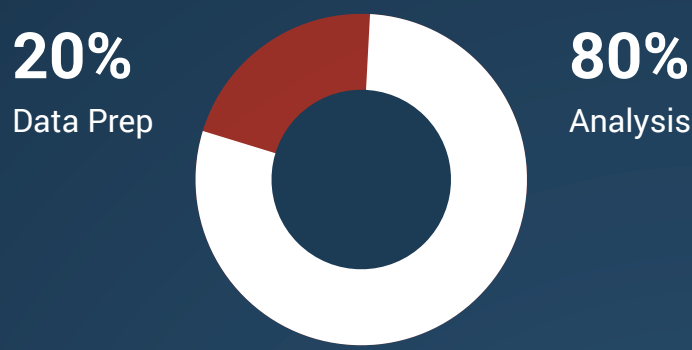


Figure 3. On average, Quants who consume data from Quilt packages spend 75% less time on data prep (finding, cleaning, and organizing data) than Quants who access the same data through files.

As a result, Quants who use Quilt trade faster than Quants who use files. See blog.quiltdata.com, "Data packages for fast, reproducible analysis in Python," for more.

"Quilt is a seamless way to ensure that our Quants are successful." –AP, Data Engineer at GFS Bank

Before and after Improving speed, security, and reliability

Table 1 indicates how data pipeline activities changed with the introduction of Quilt.

Activity	Before (using files)	After (using Quilt packages)
Upload assets to blob storage	Manual process, unreliable uploads	Automatic, scripted uploads managed by Quilt
Expose data assets via the web	Wiki links get stale, require manual maintenance	Data automatically cataloged by Quilt
Quants access data	Different API for each platform (R, Python, Excel), requires custom code, access is neither authenticated nor logged	Concise, uniform API for data packages across platforms; access is authenticated and logged

But wait... Objections to adopting a new solution

Given GFS's scale and commitment to existing data infrastructure, some common objections arose when data packages were proposed as a solution to the problem of efficiently distributing data to quants. **Table 2** indicates how these objections were overcome.

Objection	Resolution
Don't want to change existing data infrastructure	Quilt data packages are compatible and additive to existing infrastructure
Don't want to teach new behaviors or systems to data engineers and quants	Quilt manages data like code. Engineers familiar with git, Docker already know Quilt commands.
No time for long integration cycles	Quilt integrates with existing systems by converting files into packages in a matter of minutes

Return on data Discoverable, auditable, and reproducible

The key benefit of using Quilt data packages over files is an effect that we call return on data. "Return on data" means that it doesn't matter how much data your organization collects, it matters how much data your organization can leverage. Quilt packages maximize return on data by instantly bringing data into code, where it can be modeled and traded upon by Quants.

Because Quilt packages are managed like code, GFS's trading data became discoverable (packages are automatically cataloged), reproducible (packages are versioned), and auditable (each package access is logged) with zero additional engineering effort. Quilt packages also offer "checks" that ensure data is compliant (free of personally identifiable information, GDPR compliant, etc).

Want to maximize your return on data? Get in touch

QuiltData.com
contact@quiltdata.io
+1-415-323-0193

