PREFECT

Jellyfish Chooses Prefect Cloud to Power Their Engineering Management Platform

Jellyfish delivers business critical data directly to customers using Prefect Cloud

COMPANY

JELLYFISH

INDUSTRY

AI

HEADQUARTERS

Boston, MA

PRODUCTS USED

PREFECT

CLOUD

2x Speed

Reduced pipeline runtime by more than 50%.

SYNOPSIS

Introduction

Jellyfish's management platform aligns engineering efforts with greater strategic initiatives, delivering productivity analytics to synchronize engineering teams with the rest of the organization.

Challenge

Jellyfish needed to scale business-critical data processing for clients and improve visibility to quickly diagnose issues, while the customer base grew.

Solution

Jellyfish utilized the building blocks of Prefect to migrate critical Celery-based tasks, used Prefect Cloud as an operational dashboard, and developed a crucial partnership with the Prefect team and its growing community.

Results

Prefect Cloud serves as the central platform for observing and orchestrating pipelines, resulting in up to 50% speed increase in data processing, faster bug identification and resolution, and a customer success system built using the Prefect Cloud API.

Introduction

Jellyfish strives to align engineering work with business priorities. Their platform reaches out to numerous sources (GitHub, Bitbucket, Jira, Aha!, Slack, and more) and combines signals from these platforms with business goals to present a unified view of what engineering teams are working on and how that work contributes to company objectives. To achieve this, they rely heavily on their data pipelines, which are unique to each customer. Since customers can trigger refreshes within the platform, Jellyfish's pipelines are absolutely critical to the success of their product. As the team pushed forward toward an MVP, they realized that their initial workflow solution wasn't going to cut it and began to investigate alternatives.

Challenge

The primary goal was to scale business-critical data pipelines and reduce the amount of time to process customer data for timely content delivery. Additionally, the team sought to improve their ability to understand and diagnose issues in their data process as they continued to scale both the customer base and volume of processed data.

Jellyfish had a production-ready web application and data pipelines. Now, they needed to prepare to scale up and take on more customers, which introduced three challenges:

1. Unlocking parallelism within their data pipelines to ingest customer data more quickly

2. Making their pipelines highly observable and introducing alerting to ensure rapid response and resolution times when problems did occur

3. Finding a trustworthy and reliable tool to achieve objectives 1 and 2

Their initial implementation ran all tasks serially using Celery. However, limitations of the tool prevented several initiatives for the team. First, unlocking parallel computing with Celery proved to be a complicated endeavor. Second, the team's current implementation required firsthand management of metadata, introducing further operational overhead to integral procedures and compounding resolution time for debugging as the team scaled.

> " At one level Prefect ticked the basic requirement of a workflow orchestrator (like Airflow) but I could also see how Prefect was a technology that could help us scale by improving observability to our team and encouraging adherence to data engineering best practices (testing, functional paradigms, etc).
>
> JEFF BRAINERD, DATA ARCHITECT

Jeff and his team were not only seeking a workflow orchestration technology to build business-critical systems, but also a supportive team to partner with, a technology with a stake in their system to ensure success. Upon review of the Prefect ecosystem and the engineering team behind the project, Jeff and his team could clearly gain from the benefits of Prefect's open-source module: transparency, accessibility, a consistent pace of innovation, and a multitude of integrations.

> " I was very impressed with the quality of the technology as well as the overall energy of the project. Since we were building business-critical systems, we really wanted a partner, not just a technology.
>
> JEFF BRAINERD, DATA ARCHITECT

With a clear candidate of orchestration technology in mind, the true challenge was to carry out a migration efficiently while ramping up with brand new technologies for the team. Prefect Core provided the building blocks to allow the team to implement a system using existing infrastructure technologies like Docker and ECS. However, the team was not familiar with Prefect's workflow semantics and supplemental technology like Dask's distributed computing library.

> " ...no one on our team had any experience with either Dask or Prefect, so I would say the main challenge was how to move the migration process forward efficiently while at the same time coming up to speed with these brand new (to us) technologies.
>
> JEFF BRAINERD, DATA ARCHITECT

The migration challenge was a familiar one, an exercise exemplifying the company's overall mission, optimizing engineering practices to align with overarching business objectives. Prefect's mission to eliminate negative engineering positioned the two technologies as natural allies, a serendipitous partnership to improve the effectiveness of engineering teams in conjunction with their larger enterprises.

Solution

Prefect's data flow semantics and rich API provided a clear-cut means to create tasks and arrange them into a graph structure, allowing the team to convert existing business logic into production ready processes. Additionally, the team's underlying infrastructure consisting of containerized code in multiple environments on different cloud platforms could carry over to the new system, eliminating the need for a completely new deployment.

> *The fundamental mechanics of converting from Celery-based tasks to Prefect was very straightforward. Other than some refactoring and plumbing (due to the fact that we needed to do this as a side-by-side implementation since this was a running production system) we didn't have to make changes to our existing business logic.*
>
> **JEFF BRAINERD, DATA ARCHITECT**

Prefect Cloud's UI proved to be an effective operational dashboard for the team's data processes, providing pivotal insight into the state of the system. The team no longer needed to maintain their metadata and could rely on Prefect's logger for overall state and rich history details, reducing the overall resolution time for debugging issues. Not only could the team discern performance across flow runs, but also quickly jump to error tracebacks and implement critical updates.

> *We've found it very useful not only to understand the details of our running processes, but also as a mechanism for making manual adjustments as a starting point for automating some behaviors (like conditionally restarting tasks).*
>
> **JEFF BRAINERD, DATA ARCHITECT**

Throughout the implementation process, the team was empowered to realize their overarching company mission internally—optimizing their data processes and quickly onboarding new engineers. Despite the size of Jellyfish's engineering team and the scope of the migration, the team felt like they were a part of a larger team—able to rely on the Prefect team and their Slack Community for debugging and best practices.

> *While there are always issues that arise and more new features we might want with any new technology, the Prefect team has been great in supporting us and we have always felt like we could overcome any challenge that might arise. That was really important to us when we chose Prefect as our technology partner.*
>
> **JEFF BRAINERD, DATA ARCHITECT**

Solution

The full migration to Prefect for business-critical pipelines that deliver data directly to customers.

The pythonic nature of Prefect's API allowed the team to quickly migrate Celery tasks into Prefect tasks and flows.

> *We are currently running all of our business-critical data processing runs through Prefect (approximately 100 per day).*
>
> **JEFF BRAINERD, DATA ARCHITECT**

Newly unlocked parallelization in data processing reduces the time to produce fresh data for customers.

Prefect's integrations with Dask allows users to quickly parallelize their tasks and harness distributed computing.

> *We are unlocking the parallelization in our data processing, which will allow us to reduce the time it takes to provide fresh data to our customers, and giving our team observability into those processes in order to manage our customer relationships and troubleshoot unexpected issues.*
>
> **JEFF BRAINERD, DATA ARCHITECT**

> *Reduced expected pipeline runtime by more than 50%.*
>
> **JEFF BRAINERD, DATA ARCHITECT**

Prefect's Cloud UI serves as an operational dashboard for the team to answer critical performance questions.

Prefect's integrations with Dask allows users to quickly parallelize their tasks and harness distributed computing.

> " *At present the main cost saving for engineering is the improved observability we have into our data processes. It has allowed us to debug issues more quickly and understand complex behaviors in ways we couldn't before.*
>
> **JEFF BRAINERD, DATA ARCHITECT**

New engineers can onboard to Prefect with ease, realizing Jellyfish's company mission to align engineering efforts with greater business objectives.

The intuitive API and flexibility of Prefect's workflow semantics allows new engineers to pick up the team's data practices quickly and effectively.

> " *Our team has adapted well to using Prefect. The documentation is clear and useful and the core concepts of Prefect are intuitive and easy to understand. I've been very pleased with how quickly our engineers are getting up to speed using Prefect.*
>
> **JEFF BRAINERD, DATA ARCHITECT**

New data engineering practices exemplify the company's authenticity in its mission—the engineers developing Jellyfish's platform fully optimize their efforts to realize strategic initiatives.