WHITEPAPER

Full-Stack Orchestration in the Age of the Data Product

ASTRONOMER

Table of Contents

Introduction

Our reliance on data has evolved a lot over the past decade. Once regarded simply as the rows and columns loaded into data warehouse tables to power BI dashboards, data is now so much more. It has become a “product” — with source data packaged alongside artifacts such as metadata, transformation logic, docs and tests, access policies — exposed as APIs and ready for any number of consumers and diverse use cases.

Data products are powering everything from analytics and AI to data-driven applications that drive insights and actions within live applications.

Examples include retail recommendations with dynamic pricing, automated customer support, predictive churn scores, financial trading strategies, regulatory reporting, and more.

There are many reasons why enterprises adopt data products. Key drivers include the improved reliability and trust in data, composability and reusability, democratized data development and usage, faster innovation with agility and adaptability, closer alignment to the business, heightened security and governance, underpinned with lower cost and risk.

Challenges to reliably delivering data products

As all Chief Data Officers and Data Engineering leaders know, while the timely and reliable delivery of every product recommendation, dashboard, or fine tuned AI model looks easy, the reality is very different. This is because every data product is reliant on a complex web of intricate and often opaque interactions and dependencies between an entire ecosystem of software, systems, tools, and engineering teams. Much like manufacturing supply chains that take raw materials as an input and deliver finished products to customers as an output, there is huge complexity in reliably delivering data products, with a lot that can go wrong in the production process.

Data products have become business critical — any failure can have a direct impact on revenue and customer satisfaction. But the methodologies, frameworks, and infrastructure for developing, testing, and operating them in most organizations is at best immature, and in many cases, non-existent.

There are four key challenges faced by data leaders:

1. Orchestration and observability are **fragmented** across three separate layers of the data platform, as shown in Figure 1 below. As a result, they have no visibility or trust into the quality of the data product because they are unable to understand what tasks in the data pipeline ran and when, what's supposed to run, and whether things ran in the proper sequence and within the prescribed SLAs.

2. Platform and data engineers are **powerless** to prevent data downtime and pipeline errors. They spend their time reacting to failures, rather than proactively managing the data product. Issues are often not detected until the data product is being used (or is missing), by which time it's too late.

3. Infrastructure provisioning has **no awareness** of the real time computational demands of the data pipeline. This means that either budgets are wasted running resources that are not needed, or SLAs are missed because there are insufficient resources available to meet demand.

4. Custom tooling and **stifled developer experience** reduces the pace of innovation. It is not unusual to find each team adopting its own narrow, ad-hoc approach to building and managing data products and pipelines, cobbled together with custom scripts, reports, tools, and spreadsheets to track dependencies between tasks in each layer of the data platform.

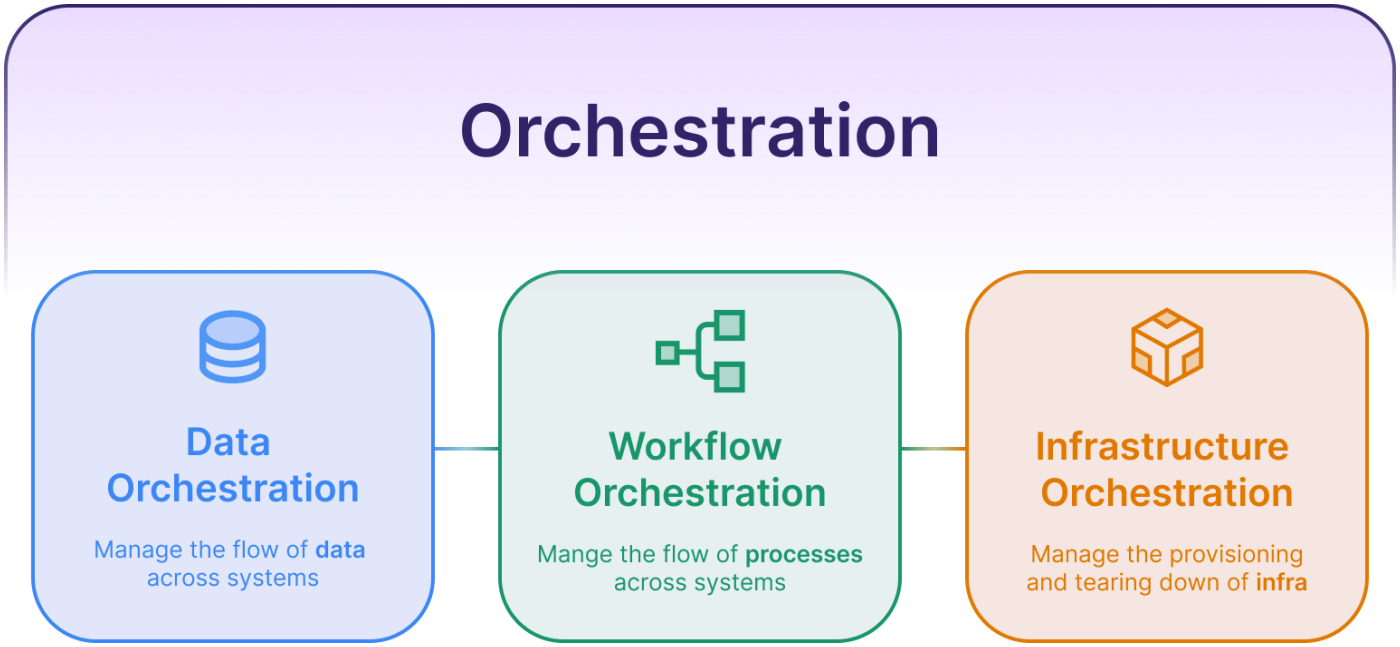

Figure 1: Reliable data products require orchestration across three layers.

The way we develop, orchestrate and observe data products needs to change. What we need to do is unify orchestration with observability across the full data stack in a single platform while applying best practices from software engineering to data engineering.

Modern, full-stack orchestration is a new approach designed for the age of the data product. By unifying orchestration and observability across the stack, the reliability and trust of data products is improved, development velocity is increased, costs are lower, data and platform engineering teams are more productive, and critical data assets are better secured and governed.

Improving what we have today is a major step forward, but the opportunities don't end there. Modern full-stack orchestration extends how organizations use data products to drive competitive advantage by elevating data products into strategic asset classes that drive innovation.

Engineered for the age of the data product: Apache Airflow® and Astronomer

Over the past decade Apache Airflow has emerged as the de-facto standard for managing data workflows and pipelines. Astronomer further expands Airflow's capabilities through its Astro platform. With observability, multi-tenancy, deeper security controls, compute awareness, lineage tracking and more, Astro provides a platform for complete orchestration of pipelines and the data products they produce.

Looking to the future, the developments driven by the Apache Airflow community and Astronomer are focused on delivering a unified orchestration and observability platform, managed with software engineering best practices. The goal of that platform is to seamlessly manage data products across the entire supply chain, autonomously correcting issues at any layer of the stack and at any stage of the pipeline.

Astro is the fully-managed cloud platform built on Apache Airflow. It is the only orchestration platform today that unifies every layer of the stack, and is furthest along in the vision of providing data and platform engineering teams with autonomous, self-healing pipelines to power their data products.

In this paper, we'll cover the evolution of orchestration, the required capabilities needed for any modern orchestration platform, and the benefits data and platform engineering teams can expect from adopting the best-in class solution. We'll highlight engineering teams who are embracing modern orchestration today along with the results they are seeing before wrapping up with resources to get you started on the journey to modern, full-stack orchestration.

The State of Orchestration Today and Future Vision

Orchestration provides the automated coordination and management of complex IT tasks, workflows, and resources to streamline processes, enhance efficiency, and improve quality of outputs.

Data products are reliant on orchestration. This is because it is orchestration that coordinates and manages the interactions and dependencies between the ecosystem of raw data, workflows, hardware infrastructure and engineering teams that are responsible for building the data product — from interconnected data sources, to processing pipelines with custom business logic, compute, storage, and networking, security and governance frameworks through to the engineers themselves. Orchestration enables the reliable, scalable, and consistent delivery of data products to consumers — whether those consumers are human users, data-driven applications or AI models.

A key enabler for the agility and adaptability of data products is for the orchestration tooling to be agnostic to system components. This enables data teams to quickly and consistently add, remove, and compose pipelines interacting with different software and systems to meet the needs of different use cases and to enhance resource utilization. Ultimately this adaptation is critical in empowering technology leaders to drive innovation and maintain a competitive edge with data.

Today, delivering a data product reliably, on time and every time, requires data and platform engineering teams to orchestrate and manage three separate layers in their data pipelines: the data layer, the workflow layer, and the infrastructure layer. While each layer comes with its own set of tools, each operates independently with no awareness of the other layers.

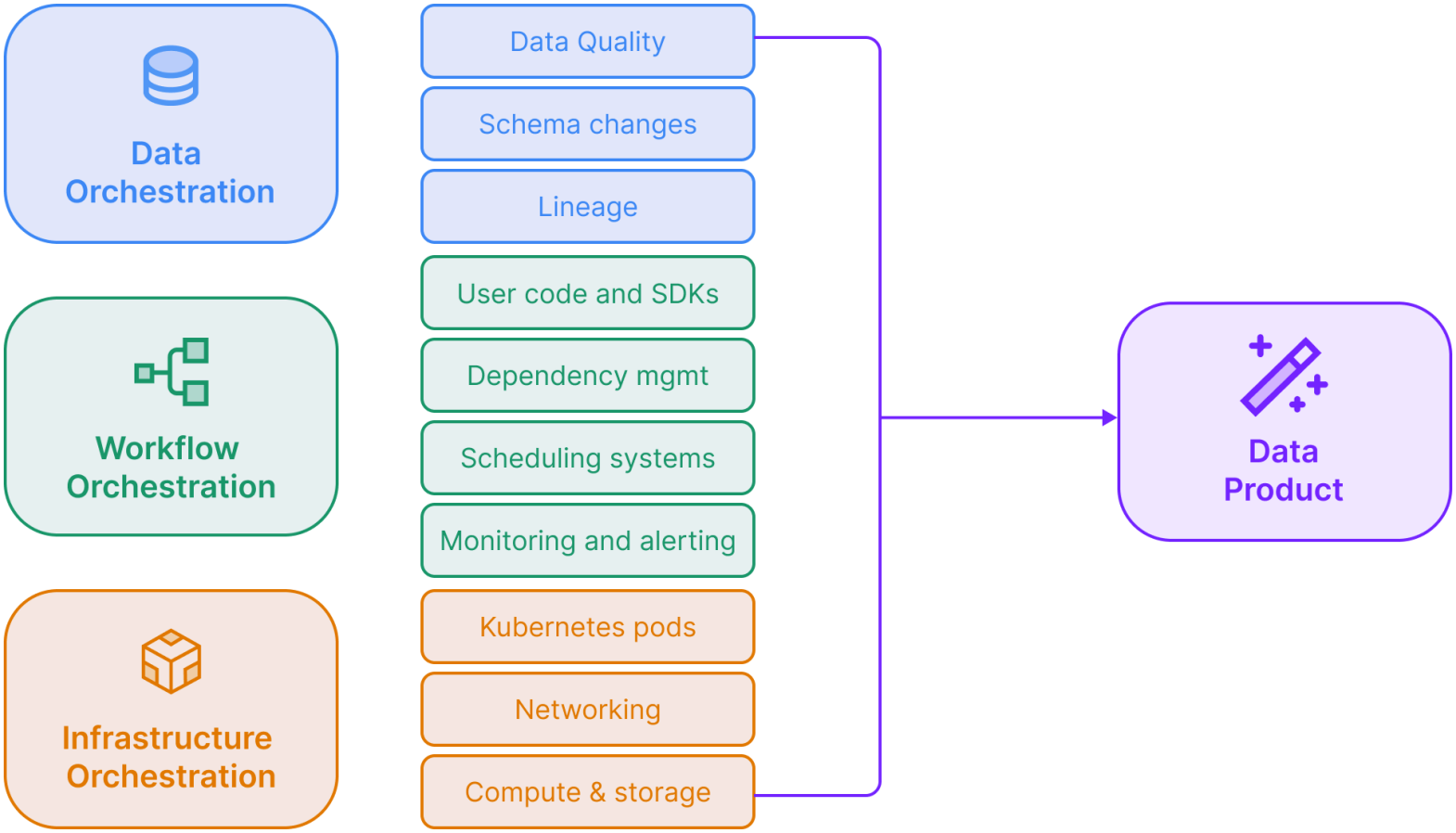

Figure 2: Illustrating some of the key responsibilities at each layer of the stack. Errors or delays in each of these responsibilities can impact the reliable delivery of a data product.

It's not enough to just detect that a database schema has changed, or that one task within a pipeline is running at high latency, or a Kubernetes pod has failed. These may initially appear as isolated incidents that can be quickly remediated, however one delay often starts a cascade of errors and failures further into the pipeline which quickly overwhelms the system and the people. These issues — which are often only detected when the data product is consumed - can compromise the very reliability and accuracy of the data product itself. Unnecessary infrastructure costs are also incurred with provisioned systems waiting idle on upstream workflows to complete.

To try and mitigate these risks and costs, teams find themselves stitching together tools from different vendors and open source projects, along with manually tracking and managing pipeline dependencies across multiple teams and owners. For their most critical pipelines, engineering teams often persist checkpoints at regular stages of the data pipeline, continuously monitoring them to check status. If any stage is running behind schedule, a Severity One ticket is automatically generated and routed to on-call engineers. These engineers then begin the arduous process of inspecting the logs of each tool in the stack to try and root-cause and remediate the issue.

This manual tracking and alerting creates significant risk as critical data products are delayed, impacting downstream consumers and business operations. Furthermore, these activities consume valuable engineering resources that would be better spent building new data products that empower the business. And, crucially, these legacy processes still don't get organizations any further forward in proactively and preemptively managing the data product supply chain — teams want more than just knowing something has gone wrong. They want to know ahead of time what may go wrong so they can prevent it. As data becomes more distributed and workflows grow in complexity, moving to modern, full-stack orchestration becomes ever more critical for platform and data engineering teams.

Lets dig into each of the three layers to better understand their purpose, along with common required capabilities across the stack.

Data Orchestration

Data orchestration automates and manages the collection, transformation, integration, and delivery of data across various systems. Its goal is to ensure that the data product is accurate, consistent, and available when needed.

Each stage of the data pipeline is often controlled with bespoke tools that provide visibility only of that stage. They are blind to the end to end supply chain or dependencies between different stages. These challenges are amplified by the emergence of newer consumers such as generative AI models, driving further fragmentation on top of fast evolving and relatively unproven technology stacks.

Workflow Orchestration

Workflow orchestration automates and manages the sequence of tasks in a data pipeline, ensuring they are executed in the correct order, at the right time, and with the necessary resources. Its primary job is to streamline complex workflows, enhance operational efficiency, ensure reliability, and maintain consistency across the stages of the pipeline needed to produce the data product. Workflow orchestration is also responsible for deploying the data product into data-driven apps, AI models, and analytics reporting.

Workflow orchestration is one of the most mature layers of the stack, but is populated with many legacy vendors and their proprietary products that offer only limited integrations with modern, cloud-native data sources, AI frameworks, and application architectures.

Infra Orchestration

Infrastructure orchestration works with Infrastructure as Code (IaC) tools as part of its workflow to dynamically provision or decommission infrastructure based on the requirements of specific tasks within the data pipeline. Infrastructure orchestration tools monitor the execution of compute jobs, triggering IaC scripts to dynamically scale up and tear down resources based on workload demands.

Like orchestration tools in the other layers of the stack, infrastructure tools have little to no visibility of the supply chain or dependencies across the data pipeline. Unnecessary infrastructure costs are often incurred with provisioned systems waiting idle on upstream workflows to complete.

Required Capabilities for Modern, Full-Stack Orchestration

Data engineers require a single platform that unifies orchestration with observability across the entire pipeline powering the data product's journey, proactively alerting them when potential issues are detected. To deliver on the promise of orchestrating data, workflow, and infrastructure as a unified stack, we need our platform to meet the following four requirements.

1. Automation backed by real time monitoring and alerting

Our platform needs to **automate** end-to-end processes and **manage dependencies** across the data product supply chain and every layer of the data pipeline — from data ingestion, integration, and transformation with task scheduling and infrastructure provisioning, all the way through to consumption — ensuring correct sequencing and execution while minimizing manual intervention and errors.

Our engineers need **real-time monitoring**, **detailed logging**, and **alerting** to quickly identify, troubleshoot, and resolve potential issues before they cause outages in the data pipeline. Integrated **error management** facilitates quick identification of issues and their root causes by correlating data across different layers in the stack, leading to faster resolution that minimizes downtime and potential data loss.

Ultimately, the platform leverages data from all orchestration layers to predict potential issues and preemptively address them before impact to consumers, ensuring the reliability and trust of our data product.

2. Unified management, observability and governance

A single orchestration platform provides a **unified interface and programmatic APIs** that centralizes the management and observability of all orchestration tasks across different teams, significantly reducing the complexity of handling multiple tools. This "single pane of glass" approach provides ease of use and lowers the learning curve by offering a cohesive platform for designing, monitoring, and managing workflows, data pipelines, and infrastructure. Additionally, it streamlines the configuration and management of dependencies across each of these layers, ensuring smoother and more reliable operations.

The orchestration platform must **seamlessly integrate** with a wide range of data sources, tools, platforms, and services to ensure smooth data flow and interoperability whether on-premise or in the cloud. Our orchestration platform needs to be extensible to support new technologies as they emerge — for example novel AI frameworks as they are adopted by developers and data scientists.

Centralizing security controls unifies the management of security policies, access controls, and compliance requirements across the entire orchestration stack. With deep observability, engineers can monitor security events across the stack, providing a unified approach to detecting and mitigating threats.

By integrating **data lineage** into the platform, the flow and transformation of a data product as it traverses each stage of the data pipeline can be tracked from source to consumption. Data and compliance teams gain deep transparency and traceability into the data product, meet governance and auditing requirements, can better assess impacts on any changes in the data pipeline, and simplify troubleshooting.

3. Democratized pipeline development

The orchestration platform enables data engineers to apply the same methodologies and best practices used by software engineers. This includes pipeline authoring across a variety of different tools such as IDEs, notebooks, command line, along with testing frameworks, and configuration-as-code. But data engineers should not be a bottleneck to authoring and scheduling data pipelines. Other functions such as data analysts, data scientists, AI engineers, and software developers should be treated as first-class citizens. For example, notebook based environments enable data analysts and data scientists to write pipelines in their preferred fashion.

Together, different teams can get their pipelines into production faster with tools including source and version control along with CI/CD integration, testing pipeline stages for correctness before promoting their code to unlock faster, safer, and more collaborative development cycles.

4. Enhanced scale, performance, and cost optimization

The orchestration platform needs to elastically scale and optimize resources and processes to handle varying data volumes, task loads, and infrastructure demands for both batch and real-time workloads without compromising performance or inflating costs.

For example, it's often impossible to accurately predict how much data needs to be processed in each run of the data product. Being able to elastically scale compute as needed ensures each task executes within its scheduled time window without wasting costs on idle resources. With observability across the stack and all the associated dependencies, we can trust that computing, storage, and networking resources are optimally allocated only once upstream workflows have been completed, and torn down when the pipeline stage has executed.

End-to-end optimization provides comprehensive performance tuning across data pipelines, ensuring low-latency and high throughput execution. Real time performance monitoring flags when a task is bottlenecked, potentially impacting the freshness of the data product, enabling timely intervention.

The Road to Modern Orchestration: The Hierarchy of Needs

Delivering on the promise of modern, full-stack orchestration requires a set of capabilities that progressively build on one another to extend from authoring workflows to providing visibility of events across the stack through to autonomous, self-healing management, as illustrated in the figure below.

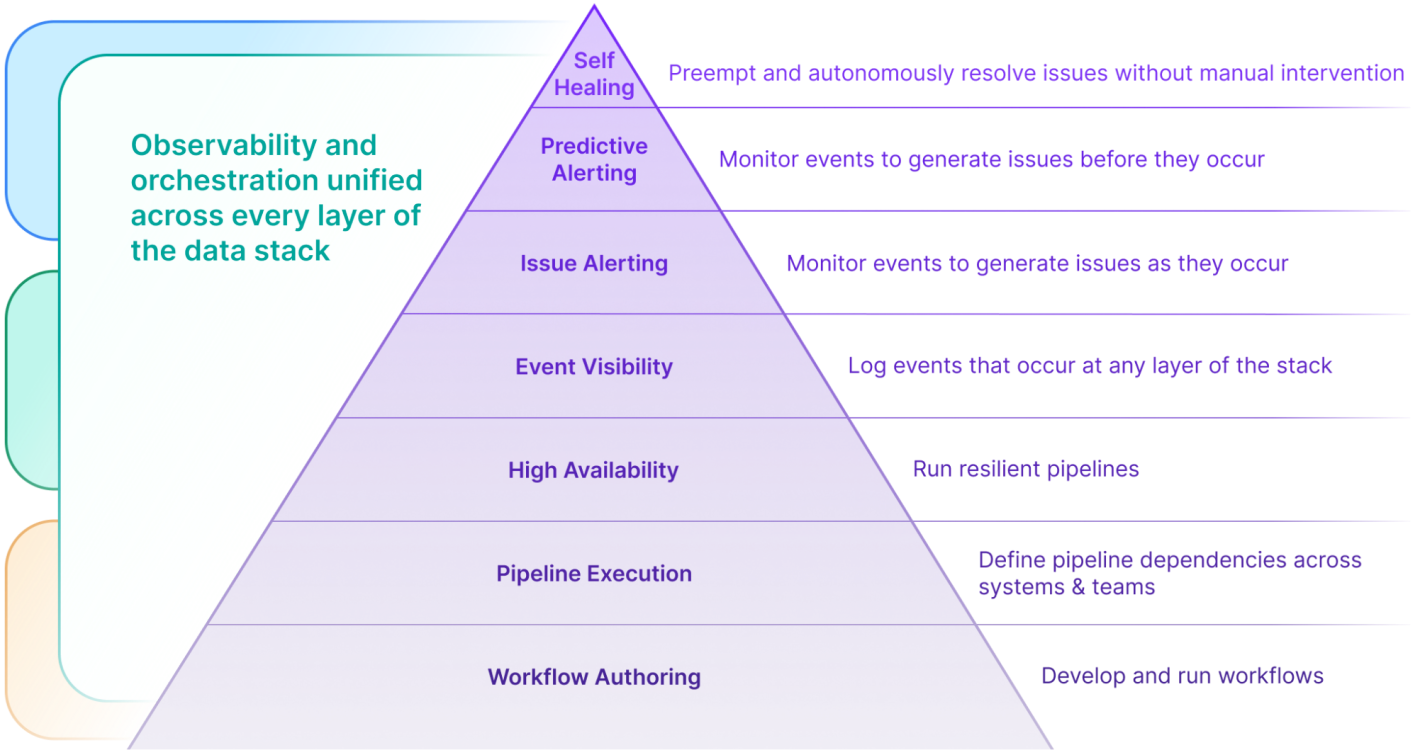

Figure 3: Progressively meeting the needs for modern full-stack orchestration

Today Airflow powered by Astro meets the requirements in the lower five layers of the hierarchy and has many of the features required to deliver on the predictive alerting layer. **No other** orchestration tool offers this depth of functionality. While self healing with full autonomy is a future vision, the Airflow and Astro roadmap already has the key elements in place to deliver that outcome.

Airflow running on Astro: The Foundation for Modern Orchestration

Apache Airflow is the industry's de-facto standard for expressing pipelines and orchestrating data flows as code. Created at Airbnb as an open-source project in 2014, Airflow was brought into the Apache Software Incubator Program in 2016 and announced as a Top-Level Apache Project in 2019. Now, it's widely recognized as the industry's leading and most advanced workflow management and orchestration solution.

Airflow provides data integrations with most of the popular databases, applications, and tools, as well as hundreds of cloud services — with more added each month. The power of a large and engaged open community ensures that Airflow offers comprehensive coverage of new data sources and other providers, and remains up to date with existing ones.

How does Airflow work?

Airflow provides the orchestration capabilities that are integral to building data products on modern cloud-native data platforms. It automates the execution of jobs, coordinates dependencies between tasks, and gives organizations a central point of control for monitoring and managing data pipelines, workflows, and the underlying infrastructure that runs them.

In Airflow, data pipelines and workflows are abstracted as directed acyclic graphs, or DAGs. Hundreds of Airflow operators are the building blocks of DAGs, giving you pre-built functionality you can use to simplify common tasks used to build data products, like running SQL queries or interacting with cloud storage and database services. Additionally, Airflow decorators allow you to effortlessly turn any Python script into an Airflow task, providing you with full flexibility to connect to any tool or service that has an API. You can combine operators and decorators to create DAGs that automate complex pipelines — like integrating newly ingested data with user feedback for AI model training and tuning, or pre-processing data from dozens of operational systems for building data driven apps or regulatory reporting.

Airflow's built-in logic allows you to focus on making sure your data pipelines are reliable and performant — without worrying about writing custom code to manage complicated dependencies or recover from pipeline failures. Review the Airflow architecture page to learn more about key concepts, features, and users.

What is Astro from Astronomer?

Astro is the fully managed modern data orchestration platform powered by Apache Airflow. Astro augments Airflow with enterprise-grade features to enhance developer productivity, optimize operational efficiency at scale, meet production environment requirements, and more. Astro enables companies to place Airflow at the core of their data operations, ensuring the reliable delivery of mission-critical data pipelines and products.

Stepping through the required capabilities discussed earlier, Apache Airflow powered by Astro provides the following capabilities along with planned roadmap enhancements that deliver on the full vision of modern orchestration.

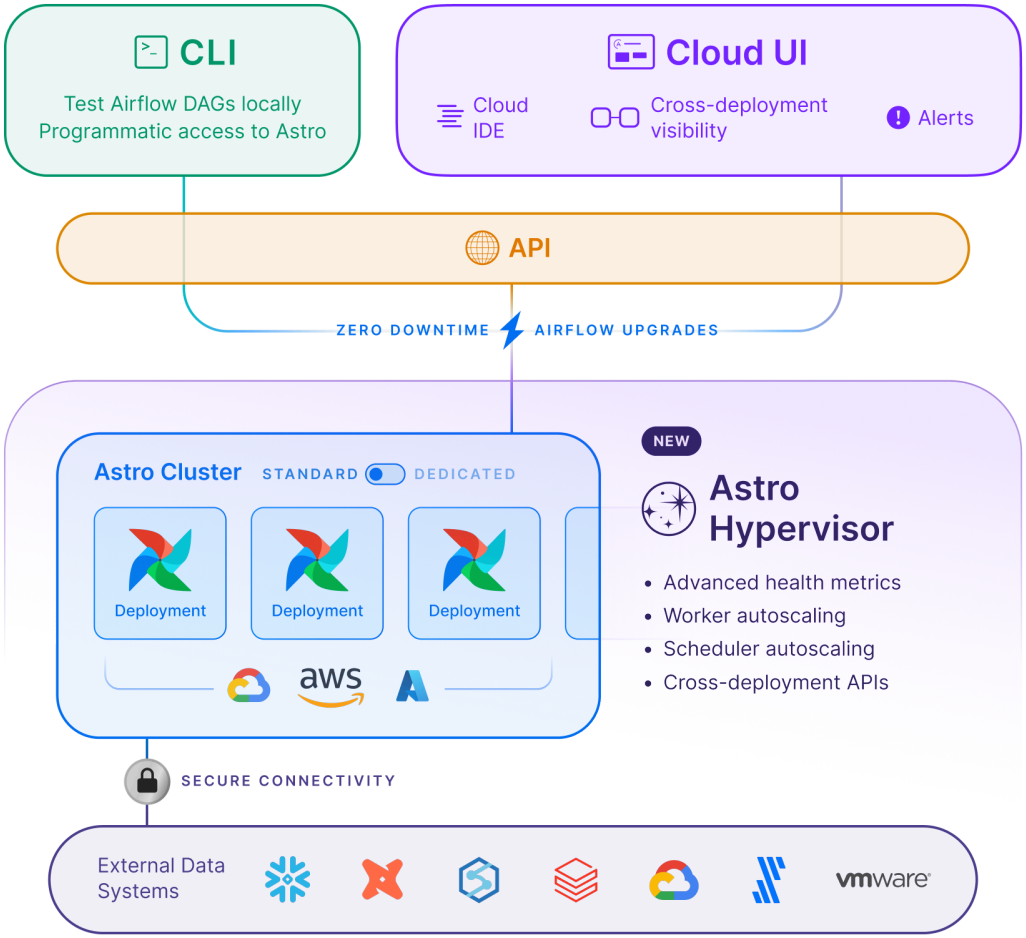

Figure 4: Astro is the unified data platform built on Apache Airflow that ensures data products are delivered on time, securely, and accurately

1. Automation backed by real time monitoring and alerting

Comprehensive automation with unified monitoring: As noted above, Airflow automates and manages the execution of complex business logic and workflows powering data pipelines. It coordinates the sequence of tasks, managing dependencies and scheduling execution times. Airflow's DAGs streamline the configuration and management of task dependencies, ensuring smooth and reliable operations across the stack.

With Airflow's Datasets and data-aware scheduling, data and platform engineers can create explicit dependencies that orchestrate workflows based on complex logic involving updates to a set of data sets. This approach helps optimize resource usage and data consistency across workflows.

Airflow's web-based UI provides detailed visibility into task execution, performance metrics, and logs, enabling proactive monitoring of workflows across the data product supply chain. Astro extends Airflow with advanced features such as real-time monitoring of pipeline status with data-centric alerts.

With an end-to-end view of data flows, workflow execution, and infrastructure health, proactive monitoring and alerting in Airflow and Astronomer serves as the foundation for full-stack observability.

Error management and observability: Airflow's integrated logging and alerting systems facilitate quick identification of anomalies. Building on Airflow's error management, Astro correlates events across different layers of the stack in real time, unlocking faster root cause analysis and resolution. For enterprises with their own internal monitoring systems, the Universal Metrics API will expose metrics and alerts from the data pipeline, enabling seamless integration and enhanced visibility across the entire technology estate.

Roadmap

Alerting and self-healing are key to delivering on modern orchestration, and there are many enhancements in development:

- Astro will be able to detect and proactively heal any Airflow system issue, automatically resolving issues that would otherwise cause pipelines to fail. This includes non-responsive celery workers, Airflow scheduler heartbeat issues, and more.
- Improving monitoring and error management, engineers will be able to configure fine-grained rules to control when alerts are triggered by any component within the data platform. Alerts will span every layer of the stack from system health and task failures to anomalies such as pipeline stages taking longer to execute than a specific threshold or normal duration.
- Further speeding and simplifying issue resolution, LLM-powered summaries will be used to automatically analyze and surface errors from logs, providing engineers with fast debugging and prescriptive recommendations on corrective actions.

2. Unified management, observability and governance

Unified UI: Airflow provides a centralized web-based UI that allows data and platform engineers to design, schedule, and monitor workflows across the data product supply chain in one place. This simplifies the management and observability of tasks across different teams.

Astro builds upon these native capabilities with cross-deployment APIs & UIs enabling features such as federated observability views that aggregate pipeline data across any number of Airflow deployments and platform-wide metadata. These give data and platform teams everything they need to run multi-tenant Airflow across their entire organization. As a managed service, Astro offloads many operational tasks while centralizing orchestration flows into a single platform, reducing the complexity of handling multiple tools and interfaces.

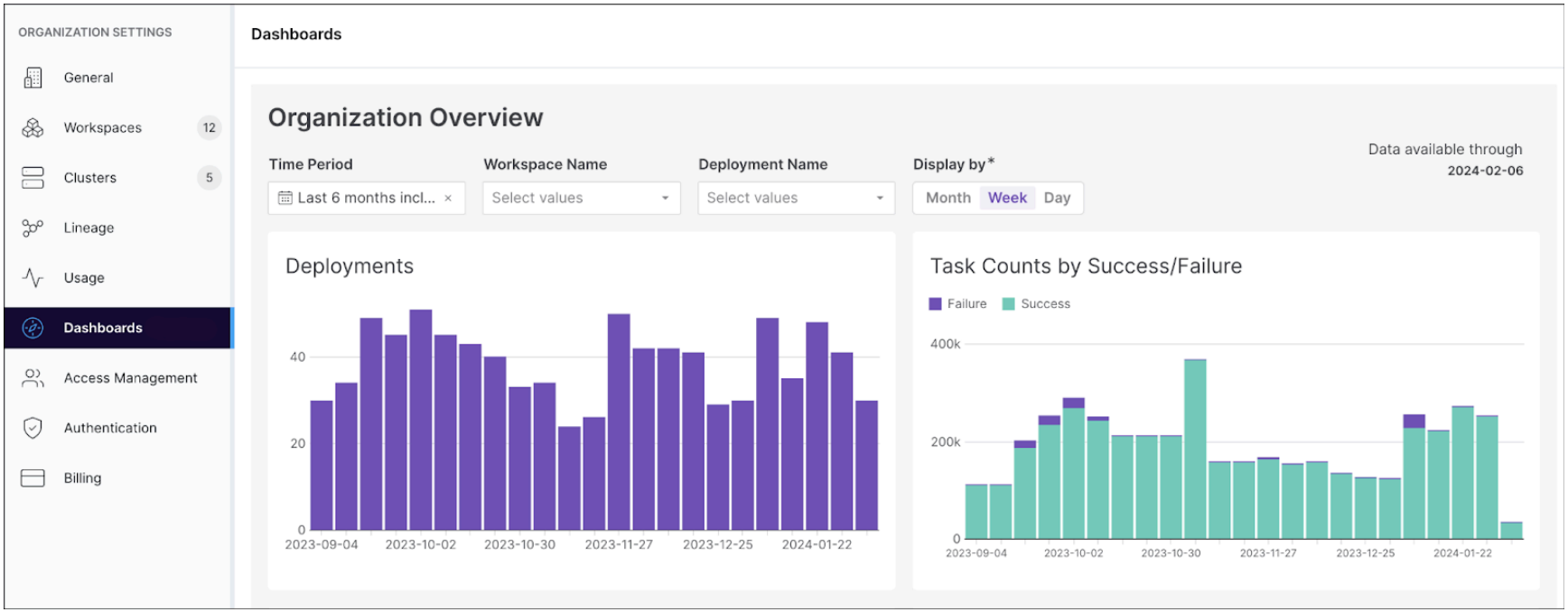

Figure 5: Astro dashboards provide at-a-glance summaries about activity across Airflow deployments in your organization.

Unified API access: Airflow offers a robust REST API that simplifies integration with external systems and services. The Astro API provides programmatic integration and access control to underlying pipeline infrastructure. With the Astro Terraform provider, data and platform engineers are able to integrate Astro into their continuous delivery workflows.

Unified security controls: Airflow supports integration with common authentication services. Astro enhances security with centralized management of security policies, fine-grained RBAC to configure custom roles for different users and environments, centralized connection and secrets management, private networking, encryption and compliance readiness across the entire orchestration stack.

Building on these controls, Astro's advanced observability features enable engineers to monitor security events in the stack, facilitating a comprehensive approach to threat detection and mitigation that protects the privacy and integrity of data products.

Roadmap

The ability for both business and data teams to better define and track data lineage is a key development focus. By integrating Astro with OpenLineage, metadata is collected from pipeline components such as datasets, schedulers, tasks, and source systems, enabling you to piece together your organization's entire data supply chain. Specifically lineage unlocks new capabilities across multiple teams:

- Business leaders and compliance teams will be equipped with a deeper understanding of how data products are produced and consumed, along with the value they deliver.
- Engineers can identify the root cause of complex issues and understand the impact of changes, improving the reliability, integrity, and agility of data products.
- Architects can use the collected metadata to optimize costs on the underlying compute engine.

3. Democratized pipeline development

Airflow and Astro provide a range of tools that enable multiple teams and personas — i.e., data, software, and AI engineers, data analysts and scientists — to get pipelines into production faster.

The Astro Cloud IDE is a notebook-inspired development environment for writing and testing data pipelines with Astro. The Cloud IDE lowers the barrier to entry for new Apache Airflow users to create standardized pipelines based on established best practices. Basic Airflow actions such as creating dependencies, passing data between tasks, and connecting to external services — tasks that previously required knowledge of Airflow-specific coding practices — can now be configured with the Astro UI. This means users need only focus on defining business logic to build pipelines using the interface of their choice — whether that be declarative Python, SQL, or low-code form templates.

The Astro CLI enables engineers to install, run, and test Airflow in a containerized local environment from their command line in under five minutes. With CI/CD integration and branch-based deployments they can test pipeline stages and dependencies in isolated environments before promoting code into the mainline trunk.

Roadmap

Further democratizing pipeline development while integrating with developer tools and workflows are a major focus for Airflow and Astronomer evolution:

- The Astro Cloud IDE is being integrated with an updated and improved version of the DAG-Factory, with Astronomer taking over maintenance of the project. DAG-Factory provides a low code library for dynamically generating Apache Airflow pipelines without users having to be familiar with either Python or Airflow primitives.
- Tools from Astronomer will be integrated into leading IDEs such as VS Code, enabling engineers to build and debug pipelines directly within their preferred development environments.
- With Automated Preview Deployments, developers will be able to dynamically create and destroy ephemeral Airflow deployments as they write and test code before submitting pull requests.
- The new integration of dbt on Astro will provide a seamless solution that combines dbt and Apache Airflow on a single platform. This feature allows teams to efficiently manage and monitor their data transformations and workflow orchestrations together, simplifying both deployment processes and operational observability. With this integration, Astro helps data teams enhance their productivity, improve data reliability, and reduce the complexity typically associated with managing separate systems across their data pipelines and data products.

4. Enhanced scale, performance, and cost optimization

Elastic scaling with performance insights: Airflow's ability to scale horizontally by adding more worker nodes supports linear performance scaling as workflows become more complex and data volumes grow. Features like Astro's dynamic workers elastically adjust resources based on workload demands, ensuring compute and storage is available only as and when data products need it. With "Scale to 0 Deployments", engineers can schedule zero-cost downtime to optimize development deployment cost savings.

Building on the monitoring and observability features discussed above, Airflow and Astro provide real-time insights into task performance, identifying bottlenecks and enabling proactive intervention to maintain the freshness and reliability of data products. Airflow's scheduler efficiently allocates tasks to worker nodes, optimizing resource utilization and ensuring balanced load distribution across the infrastructure. With task-optimized worker queues, Astro optimizes resource allocation across data processing, workflow execution, and infrastructure provisioning.

Resource and cost management: With Astro's comprehensive view of resource usage across the entire orchestration stack, data teams get visibility into resource consumption for budgetary planning and chargeback.

Cost control is a first-class feature on Astro with built-in dashboards showing detailed break-downs of deployment, compute, and ephemeral storage cost per workspace and data pipeline over time. This single pane-of-glass view provides administrators with the tools to effectively ensure governance of budget allocation and quickly identify any sources of excessive spending.

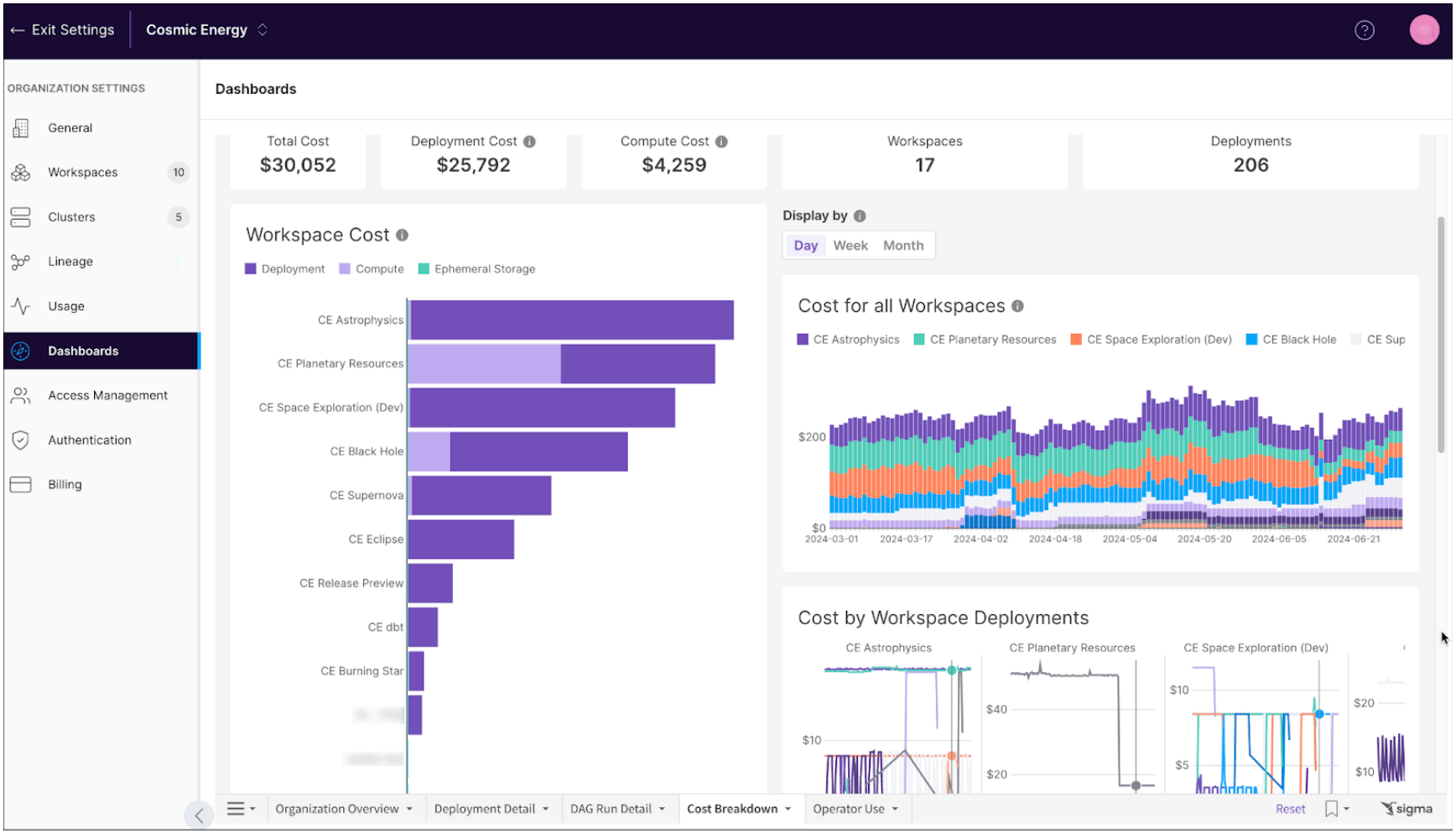

Figure 6. Astro dashboards provide granular insights into cost-breakdown per deployment, allowing for precise governance of budget allocation.

Building on Astro's native cost controls, an application-specific example of how Airflow can be used directly for managing specific service costs is Astronomer SnowPatrol, an open source tool for anomaly detection and alerting of Snowflake usage. Powered by machine learning, SnowPatrol identifies anomalous usage activity. It also tracks the Snowflake costs associated with every Airflow pipeline and task. Astronomer plans to expand these capabilities to other data warehouses and data lakes in the future.

Roadmap

Future developments include:

- Task-level resource observability by surfacing CPU and memory consumption of each system in the data pipeline. Deeper granularity enables better performance and cost optimization across every stage of pipeline execution.
- Support for GPU-node types will be added to pipeline execution stages, further improving the performance of data products used for AI model training and fine-tuning.

Industry Leaders Embracing Modern Orchestration

Autodesk

Autodesk, a global leader in software for architects, builders, engineers, designers, and 3D artists, recognized the need to adopt DataOps methodologies and cloud-native services to enhance its leadership position. The data science team also saw opportunities to integrate more Machine Learning (ML) into business processes. However, legacy workflow scheduling software hindered these initiatives due to its lack of support for modern cloud services and ML frameworks, along with constant reliability issues and its lack of scale.

Autodesk's platform team turned to Airflow, powered by Astronomer, for a fully-managed service to orchestrate data pipelines and workflows across both cloud-native and on-premise infrastructure. By working with Astronomer consultants, over 500 critical data pipelines managed by 25 engineering teams were migrated to Airflow in just 12 weeks.

Nick Wilson, Senior Manager of the Autodesk Platform team, stated, "Our adoption of Astronomer and Airflow is about helping our teams adopt modern software engineering practices and providing them with a self-service infrastructure they can manage intuitively."

Now, engineering time is spent developing new data products and services instead of fixing workflow failures. Read more in the full case study.

FanDuel

FanDuel Group is a sports-tech entertainment company and the premier gaming destination in the United States. The company has a presence across all 50 states, with approximately 17 million customers and nearly 30 retail locations.

FanDuel's internal and external stakeholders depend on timely data products that feed into reports, dashboards, and other analytics that support the organization's day-to-day, strategic, and operational decision making.

FanDuel's growing data needs strained their open-source Apache Airflow setup, unable to scale with the increasing complexity and volume of data pipelines. The system's limits in handling large concurrent users and workloads prompted the need for a more powerful orchestration tool.

Ahead of the 2022 NFL season, FanDuel partnered with Astronomer to migrate to Astro as its cloud-based orchestration platform. Astro's elastic autoscaling and deferrable operators enhanced performance and stability. This migration, combined with a review of resource-intensive data pipelines allowed FanDuel to optimize performance and reduce complexity.

By switching to Astro as its full stack orchestration platform, FanDuel reduced cloud resource usage by 35% and increased task handling capacity by 305% per Airflow worker. The consistent and stable resource usage eliminated the need for additional Kubernetes capacity, cutting costs and improving pipeline reliability and execution speed.

Now the company can meet its BI and analytics demands efficiently, supporting rapid business growth with dependable, cost-effective orchestration. Read the full case study for more insights.

Trellix

More than 40,000 customers, including nearly 80% of the Fortune 500 rely on Trellix for advanced cybersecurity powered by AI, automation, and analytics. The company's open and native extended detection and response (XDR) platform helps organizations confronted by today's most advanced threats gain confidence in the protection and resilience of their operations.

The company's Enterprise Data Platform team recognized early on that data products would help it better elevate data into an enterprise asset. Following a series of mergers and acquisitions, the company sought to standardize on a modern data stack and integrate it into its cloud native architecture. As a part of that, it needed a cloud native orchestration tool to work across data pipelines, workflows, and infrastructure.

The company selected Airflow powered by Astronomer as that cloud native orchestration layer. By running the fully-managed Astro service with Astronomer they were able to remove operational overhead from its engineers, freeing them up to deliver data products to the business. You can hear more in this short interview with the head of Trellix's data platform team.

Investment management and high frequency trading

A US-based investment management company turned to Astronomer for data-powered apps supporting the high frequency trading of tens of USD billions of assets.

The data engineering team was unable to keep pace with the growth of new analytics and reporting data products demanded by the business. They faced challenges in orchestrating infrastructure managed by Kubernetes and maintaining Apache Airflow for 25+ engineering teams.

The solution for them was to turn to the fully managed Astro cloud service. Offering enhancements in operational automation, version upgrades, and security controls, data engineers were able to focus their efforts on helping the business build and evolve their data products.

Getting Started on Your Journey

Many organizations are still early in their journey towards data products. But to data and platform engineering leaders it's clear that moving to modern, full-stack orchestration is essential if their organizations are going to realize the value of these initiatives.

Engineered for the age of the data product, Airflow powered by Astro delivers the platform that unifies orchestration and observability, managed with the developer tools that make your data engineers more productive. To get started, contact us. We will schedule a design review with your team to assess your current state and provide actionable recommendations to support you on your journey. You can also explore Astro on your own either by creating a free-trial or taking a guided tour.

Safe Harbor

The development, release, and timing of any features or functionality described for our products remains at our sole discretion. This information is merely intended to outline our general product direction and it should not be relied on in making a purchasing decision nor is this a commitment, promise or legal obligation to deliver any material, code, or functionality.