celigo

celigo

integrator.io

Next-generation iPaaS

The highly scalable and robust iPaaS

Guide

SKU: GD-IOSS-20210308

Table of Contents

Core application fundamentals

There are no practical limits within a single integrator.io account regarding:

The number of applications that can be connected.	The number of flows that can be defined.	The number of flows that can run in parallel.	The size of data that can be processed.

- integrator.io is architected as a streaming platform where huge data is always broken down into smaller pages of data. This allows very big data to travel through our system in a scalable manner and seamlessly flow into external apps that do not natively support huge payloads.
- Pages of in-process data are temporarily stored in highly redundant data stores (such as S3 or MongoDB), and Amazon SQS is used to guarantee processing at scale for all the individual pages of data in transit. If any system goes offline, the architecture above allows for elegantly pausing and resuming flow-processing activities without losing any data.
- Data sent to integrator.io listener APIs is only acknowledged after temporarily being persisted to redundant data storage and successfully queued in SQS. This protocol allows external applications to be certain that their data will be processed by a flow, or that it needs to be resent, etc.

- The integrator.io scheduler is robust enough to recognize when integration flows miss their last scheduled run due to a downtime event. It will automatically schedule flows to run immediately if they are overdue.
- integrator.io has the resilience to recognize intermittent network errors and automatically retry them.
- integrator.io has the ability to recognize field errors and auto remove fields from API retry requests, so that critical integration flows do not fail due to field-level data errors.
- Errors that cannot be automatically recovered are displayed on user-friendly dashboards, and customers can troubleshoot these errors for 30 days – including manually modifying and retrying failed records.
- integrator.io supports a large number of configuration options to tune the performance of an integration flow. For example, you can control the page size of data traveling through a flow or the number of concurrent requests a specific connection is allowed to make at any given time, etc. Integration flows can also be set up to process only delta data such that external applications are not overwhelmed by large amounts of unchanged data being synced.

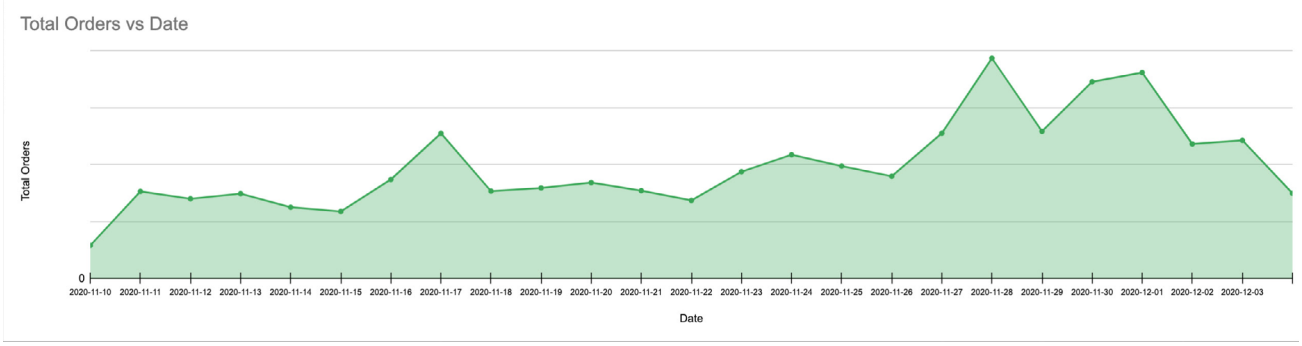
Technical fundamentals

- integrator.io is a 100% multi-tenant platform built on entirely elastic infrastructure at Amazon Web Services (AWS), running in an Amazon Virtual Private Cloud (VPC).
- Amazon Simple Storage Service (S3) is used to temporarily store customer data. Read more about Amazon S3 data durability.
- Amazon Simple Queue Service (SQS) is used for queues and messaging. Read more about Amazon SQS as it relates to scalability, reliability, and security.
- MongoDB Atlas is used to store integration definitions. Read more about MongoDB Atlas.
- Confluent (Kafka) is used to stream process event data. Read more about Confluent.
- Amazon Simple Email Service (SES) is used to send email notifications. Read more about Amazon SES.

- Amazon ElastiCache is used for caching. Read more about Amazon ElastiCache.
- Amazon Route 53 is used for DNS. Read more about Amazon Route 53.
- Amazon Web Application Firewall (WAF) is used to protect against common web exploits affecting availability, security, etc. Read more about Amazon WAF.
- Amazon Shield is used to protect against DDoS attacks. Read more about Amazon Shield.
- Application services built by Celigo engineering are always designed to be horizontally scalable.

Stability and reliability statistics

- Celigo has maintained a 99.99% uptime for the last three years. Contact support for a recent uptime report.
- There is NO scheduled downtime ever.
- We report system outages to an independent status page.
- Our platform is designed to handle high traffic periods such as Black Friday and Cyber Monday without outages or delays. The chart below represents the load of orders synced through Celigo's platform during the peak holiday shopping period for BFCM 2020.



- The number of orders synced through Celigo's Integration Apps for the weekend of Nov 27 through Dec 1, 2020 more than doubled that of the weekend before. Additionally, this load was four times greater than the average weekend outside the holiday season.

- As this real world example illustrates, our stable platform can reliably and efficiently handle load increases of over 100% without incident. This is made possible thanks to the following core technical fundamentals established previously:

    - Our iPaaS, integrator.io, is designed as a "streaming platform" that can break down large data payloads into smaller pages of data. This means that we can make huge amounts of data travel seamlessly even to apps that can't normally support large data payloads (read: Black Friday orders).

    - We've implemented protocols that essentially allow your applications to know whether the data they send will be processed by an integration flow. If not, they will be re-sent to try again.

    - We support a large number of configuration options to tune the performance of your integration flows. For example, you can set up your integration flows to only sync changed or updated data, rather than everything under the sun. To learn more about optimizations and best practices to prepare your integrations for the holidays, you might be interested in this article on Leveraging multiple connections in your Integration App flows to improve throughput.

    - integrator.io uses Amazon Simple Queue Service (SQS) for queues and messaging and Confluent (Kafka) to stream process event data. Both are industry-leading solutions for scalability and reliability.

    - integrator.io has an uptime of 99.999% (and repeating) over the past three years, because customers depend on this uptime to run their essential business processes. Additionally, we never have scheduled downtime.

Security fundamentals

Employees

All Celigo employees are required to pass a background check. In addition, employees in engineering, services, support, and operations (basically anyone with access to anything deemed security sensitive) are required to use LastPass, with multifactor authentication enabled, to store and generate all credentials used to perform job functions. Engineering employees with access to production systems are also required to undergo varying levels of security training at least annually. All Celigo employees are always granted access only to the minimal number of applications or systems needed to perform their job functions.

Application

integrator.io is built using best-of-breed technology frameworks and secure software development practices. Production and testing environments are completely segregated from each other, and customer data is never used in QA or developer testing. Security-related bugs are always assigned the highest priority, and a root cause analysis is performed for all major bugs that make it into production. Both vulnerability and penetration testing are performed at least annually. HackerOne is used to engage outside security researchers to expose vulnerabilities in the integrator.io platform (for bounty). Access to the integrator.io web app is protected by username/password (passwords are one-way hashed), and access to the API is protected by bearer tokens. Both web and API access require SSL.

Customer data

All data temporarily stored and processed by integrator.io is encrypted in motion and at rest. Sensitive credentials stored in integrator.io are encrypted via AES 256, and is never viewable in plain text by anyone. The encryption keys used to encrypt and decrypt information or data are always kept physically separated from the encrypted information or data at rest. All integrator.io core application information is stored in a high-availability MongoDB cluster, and full backups are generated daily. For the external data being processed and integrated, a combination of the integrator.io primary MongoDB application database and also Amazon S3 are used for temporary storage. External data is never persisted for more than 30 days, and it is only persisted for the purpose of safeguarding data while in transit, or to facilitate error recovery and retry capabilities.

Security webpage

See the Celigo privacy policy, cookie policy, and GDPR compliance.

Compliance

We have SOC 2 reports available, are GDPR ready, and are almost ready for HIPAA.

Operational fundamentals

- Celigo has a full DevOps team on staff monitoring the integrator.io platform 24/7. The DevOps team has employees in multiple different locations, and each employee on the team is fully equipped to work remotely or from a Celigo office.
- Pingdom is used to independently monitor integrator.io uptime percentages. If Pingdom discovers anything is offline, then PagerDuty will contact an on-call DevOps engineer.
- Celigo engineering actively uses a variety of tools to analyze logs, application stats, machine stats, etc., so that systems are always in tip-top shape.
- All bug fixes, enhancements, new features, etc., undergo a rigorous testing and review process before any changes are pushed to the production platform environment.

1820 Gateway Drive, Suite 260
San Mateo, CA 94404 USA
650.579.0210

www.celigo.com
sales@celigo.com