

BLUEPRINT

How to Implement SnapLogic



Contents

Get Started With SnapLogic	3
Setting Up SnapLogic	4
Team Size	4
Users and Roles	4
Access Controls	5
Plex Setup	5
Nodes Setup	6
Node Sizing	6
CI/CD	7
Security and Privacy	8
Password Management	8
Metadata	8
Sensitive Data	8
Encryption and Data Management	8
Recommended Naming Guidelines	9
Project Spaces	9
Recommended Naming Guidelines	9
Project Naming Convention	9
Pipeline Naming Convention	9
File Names	10
Accounts	10
Tasks	10
Integration Strategy and Execution	11
Prioritizing Integrations	11
How to Proceed	11
Considerations	11
Enterprise Automation Pipelines	12
Human Resources	12
Sales and Marketing	12
Finance and Accounting	13
Customer Support	13
Scaling Integrations and Beyond	14
AI Enhancements	14
SnapLogic Best Practices	15
Pipeline Design and Development	15
Pipeline Management	15
Tasks	15
Administration	16
Additional Resources	16

Get Started With SnapLogic

Whether your organization seeks to automate end-to-end systems and processes, unite your data for analytics and reporting, or modernize your legacy applications and integrations to achieve digital transformation, an integration platform-as-a-service (iPaaS) allows you to orchestrate all data and applications, on-premises or in the cloud.

Companies like AstraZeneca, Box, Schneider Electric, Sony, Carfax, Eero, and others partner with SnapLogic to help them deliver products and services to market faster, delight customers with exceptional customer experiences, and achieve business results.

This blueprint outlines a plan with steps and considerations for implementing SnapLogic in your environment. You can customize the plan based on your organization's needs, use cases, and outcomes.

On average, companies that implement SnapLogic see:

83%

improvement in time
to market

2.6

full-time employees required to
develop and administer app and
data integrations

Days

to deploy an
integration project

Setting Up SnapLogic

Consider the following items before setup:

Team Size

Identify a dedicated person or team who will be responsible for implementing SnapLogic. The size of the team depends on the size of your organization, the size of the infrastructure, and the integration requirements.

For smaller, leaner organizations, we recommend starting out with 1 person to build, maintain, and govern integrations in the SnapLogic platform. They don't have to be an FTE fully dedicated to managing the SnapLogic platform. They can manage their own nodes.

For larger organizations, the size of the team may range between 2-6 people to manage the entire infrastructure. Ideally, some of the team members will have core DevOps skills to automate VMs and nodes. For example, if you have 6 team members, two can manage AWS,

two manage Microsoft Azure, and the other two can manage on-premises data. They would be responsible for automating, restarting, managing error messages, SSL renewal, load balancer, and storage set-up.

Users and Roles

Typical roles and responsibilities will vary, depending on the size of your organization. Larger organizations will require additional responsibilities for managing the platform, including:

- Monitoring and observability of the platform
- Governance processes to ensure pipeline quality
- Planning scheduled tasks across a large-scale deployment

Small org (>500 people)	Medium org (500-900 people)	Large org (1,000+ people)
<p>Integration Developer (small, medium & large) - Focuses on creating, implementing, and maintaining integrations between different systems, applications, and data sources in SnapLogic. They must be adept in data and app integration principles, and have a good grasp of the APIs and data formats of the systems they are integrating.</p>		
<p>DevOps Engineer (small, medium & large) - Works on the infrastructure and processes that support the development, deployment, and operation of integration solutions within the SnapLogic platform. With an additional focus on automating and optimizing deployment pipelines, ensuring continuous integration and continuous delivery (CI/CD) of the integrations. Monitoring the health and performance of the integrations.</p>		
	<p>Platform Administrator (medium & large) - Responsible for the day-to-day management and operations, including user management, security settings, and troubleshooting issues that arise. Also works closely with integration developers and architects to implement and support their requirements and might be involved in setting up and maintaining the necessary infrastructure for the platform's optimal performance.</p>	
	<p>Integration Architect (medium & large) - Responsible for designing and overseeing the overall integration strategy and architecture within the organization, ensuring that the integration solutions developed align with the business goals and IT strategy. They work on defining the best practices for integration, choosing the right integration patterns, and ensuring the scalability and performance of the integration solutions.</p>	
	<p>Platform Owner (large) - Responsible for the overall management and success of the SnapLogic platform within the organization. Involves strategic decision-making, budgeting, resource allocation, and long-term planning across multiple business units & projects. Ensuring that the platform meets the business needs, manages the relationships with SnapLogic as a vendor, and aligns the platform's capabilities with the organization's objectives. They may also be responsible for ROI analysis and reporting on the effectiveness of the platform.</p>	

Access Controls

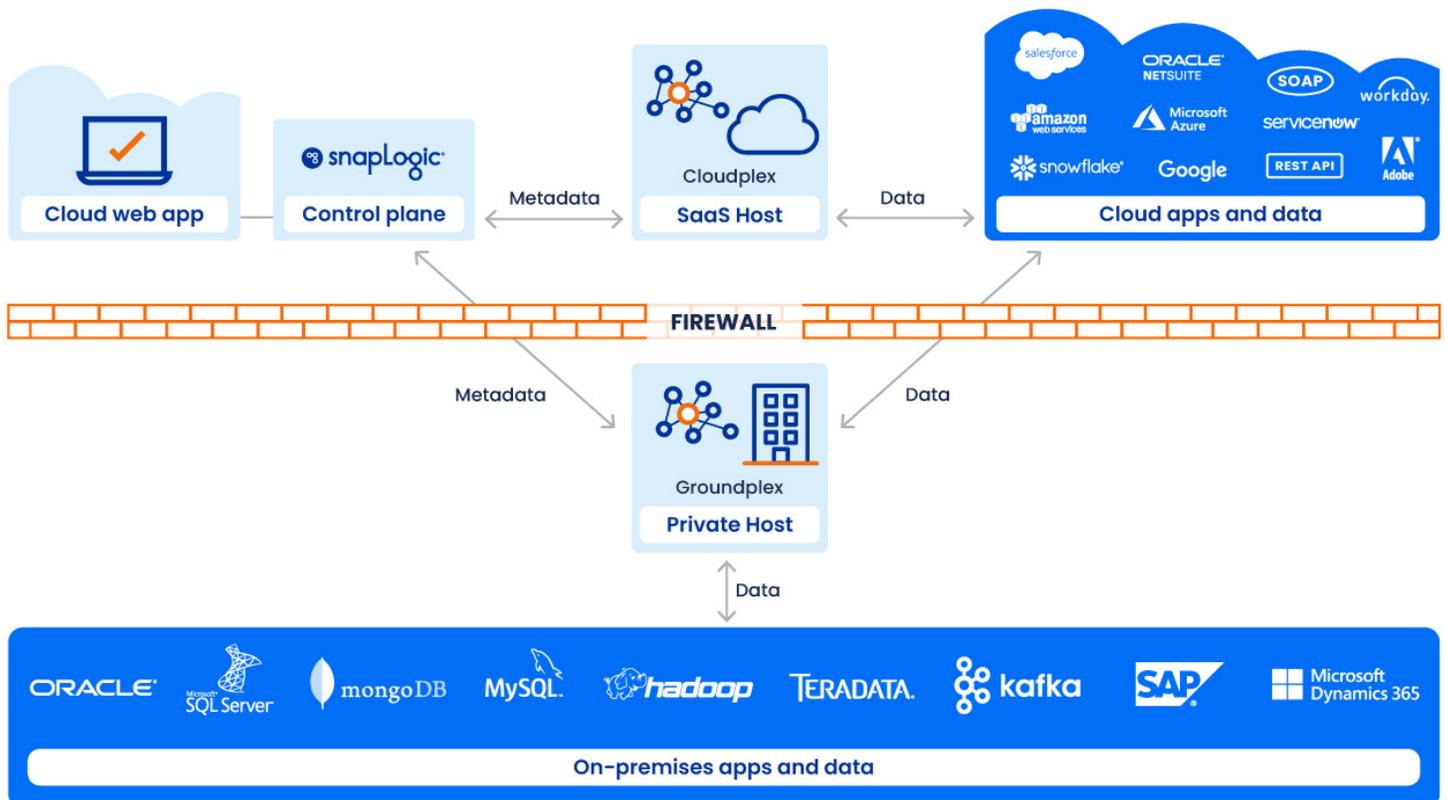
You should have at least one org administrator who will be responsible for creating users and user groups, creating projects and project spaces, creating Groundplexes, and adding nodes to Groundplexes. Designate integrators to specifically build in the development environment, then integrators who can test in the staging environment, and a few who will deploy the integrations in the production environment.

Plex Setup

You will set up a Cloudplex, Groundplex, or both depending on where you plan to run your integration pipelines. SnapLogic is divided into two main parts: the Control Plane and the Data Plane. You come in contact with the Control Plane through the SnapLogic user interface. Behind the scenes, the Control Plane will communicate with the Data Plane, which is connected to Snaplexes, and coordinate the flow of the data in pipelines. The Data Plane runs a pipeline in an integration runtime, called a Snaplex, which is a collection of computing resources. A Snaplex can be deployed on a server or in a container. The Data Plane can connect to a number of different types of Snaplexes.

There are two types of Snaplexes:

- A **Cloudplex** is a SnapLogic-managed plex where we run computing resources in the cloud to process your data. Cloudplex is the easiest and the most ideal solution if you want SnapLogic to manage everything.
- A **Groundplex** is a plex that runs in your own managed domain, on your hardware resources, whether it's on-premises or in the cloud. Even though it includes 'ground' in the name Groundplex, a Groundplex can be in the cloud. The distinction is that it runs behind your firewall on hardware that you manage. In some cases, organizations prefer running their pipelines only in Groundplexes, regardless of whether they are on-premises or cloud applications, to adhere to their organization's security and compliance requirements. For example, you can configure enhanced account encryption on Groundplexes and have full ownership of the key for encryption.



You will need to be a SnapLogic Admin to set up Snaplexes. Go to the SnapLogic Manager to set up a new Snaplex. Click the **Snaplex** tab in the page that displays the assets for that Project. Then the 'Create Snaplex' dialog that appears contains the following tabs:

- Settings
- Logging
- Node Properties
- Node Proxies (Enabled only for Groundplex nodes)

Depending on your company's architecture infrastructure strategy (Cloud only, on-premises only or hybrid), you should take in consideration which Snaplex type to use. You can start by setting up 1 Snaplex (Cloudplex or Groundplex) in your environment. We recommend using Cloudplexes for ease of management. We have also seen customers who deploy Cloudplexes who want a more SaaS-like experience, may have a small DevOps team or plan to have a NoOps IT environment.

On the other hand, Groundplexes are usually deployed for legacy applications that reside behind your firewall, and for companies that need to adhere to industry compliance, security standards, national and international data laws, and more.

Nodes Setup

A Node is a dedicated processing engine that is governed by a Groundplex or Cloudplex. The Node can be installed on a physical or virtual machine running a Windows or Linux based operating system. We recommend setting up a minimum of two nodes in the beginning so that if there are hardware or software failures on a node, you will have a backup node to keep the job running. Additionally, we also recommend customers to calculate the number of nodes with 'N+1' where the N is the minimum of nodes needed to run workloads, and the extra node can be used for load balancing during rolling restarts.

Be aware that you will need to distinguish production and non-production nodes. Depending on how DevOps teams operate, Some DevOps teams run nodes 24/7, while other DevOps teams shut down non-production Groundplex nodes in the evenings and weekends to reduce their own infrastructure costs.

Identify the type of integration jobs you will do in order to set up the appropriate number of nodes. We recommend that you run batch or streaming jobs or handle sensitive and non-sensitive data in separate nodes.

Node setup considerations:

- Identify batch, event/streaming data, and API workloads
- Critical vs. non-critical workloads
- Sensitive vs. non-sensitive data
- Long-running jobs vs. short-running jobs
- Environments: Development, Staging, Production
- Teams or departments that need separate execution nodes
- Nodes can be containerized so that jobs run in their own container
- Ultra pipelines should have dedicated Feedmaster Nodes (two are recommended for high availability) and Execution Nodes, as they are always running. Ultra pipelines provide the speed and scalability to run the most important integrations that require high availability, high throughput, and persistent execution.

Node Sizing

You can estimate the size and number of nodes needed by speaking with a SnapLogic Solutions Engineer to determine what will fit your business. We typically recommend starting with medium-sized nodes and increasing as needed. Node sizing will be indicated on the contract.

The chart below shows examples of SnapLogic implementation depending on the size and scope of your integration needs.

Small	Medium	Large (Global distributed team, 24/7 support)
SnapLogic team: 1-3	SnapLogic team: 4-10	SnapLogic team: 10+ (on-site and offshore)
1-5 Snaplex	2 Snaplex (1 non-prod, 1 prod)	8 Snaplex (2 dev, 2 staging, 4 prod), 2 sandboxes
Nodes: 2 (1 dev/test, 1 prod)	Nodes: 8 (4 dev/test, 4 prod)	Nodes: 24 (16 dev/test, 8 prod)

Some customers deploy additional Snaplex sandboxes to test SnapLogic’s quarterly releases before they make the updates in their production environment. This is to ensure mission-critical jobs are not disrupted and the customer is ready to update their production environment.

CI/CD

We support CI/CD products such as GitHub, GitLab & Azure Repos, in addition to the out-of-the-box Copy and Migrate features.

Customer Spotlight

One of our largest customers set up 8 nodes in their environment initially. They had two in the Development environment, two in the Test/Stage environment, and four in the Production environment. They expanded to 16 and 24, respectively, as they migrated their integrations to SnapLogic.

They migrated their batch jobs first as they were scheduled, predictable, and would not disrupt any business processes. Once they completely migrated their batch jobs, they transitioned to migrate streaming, real-time data, which is more complex as they are intertwined with business processes.

 **Tip:** Some ETL pipelines are resource intensive due to certain transformations. You may need to move such transformations into a child pipeline then make use of Pipeline Execute to balance the work across multiple nodes. Alternatively, you can keep pipelines lightweight if you do ELT, where you do all the data transformation at the destination source (ie. Snowflake, Amazon Redshift, SAP, or others).

Security and Privacy

Password Management

- Single Sign On (SSO) is a convenient way for users to log into multiple software systems without needing to enter their username and password for each system. SnapLogic supports SSO through the Security Assertion Markup Language (SAML 2.0) standard. The supported authentication methods include OpenAM, OKTA, and Ping
- Password security and permissions: SnapLogic provides a comprehensive set of password complexity requirements and expiry windows that are consistent with web application best practices, if you do not use SSO

Metadata

The SnapLogic Metadata (definitions of pipelines, tasks, execution runtimes, etc., and not customer data) is stored in the SnapLogic Control Plane (currently running on the Amazon EC2 infrastructure). The metadata is secured inside the protected SnapLogic environment, and only accessed by the SnapLogic Control Plane Services – no access is permitted by any outside service.

Sensitive Data

SnapLogic Control Plane encrypts sensitive data such as account passwords, secret keys, and other sensitive fields depends on the account type. Such fields are encrypted by default with keys generated during the execution node deployment and SnapLogic supports replacing these keys with customer-specified encryption keys if desired.

Encryption and Data Management

SnapLogic provides adequate protection of sensitive customer data through a combination of access controls and encryption.

SnapLogic encrypts data at the disk level with the account data stored in a server-side encrypted bucket in the Amazon S3 environment. We also support AWS Private Link and Credential Vault Secrets management.

- Enhanced Account Encryption: By default, SnapLogic orgs use keys provided by SnapLogic for the authentication of infrastructure components, such as Snaplex nodes, and encryption of the accounts used in the pipelines. With this infrastructure, the user may use a combination of cloud- and ground-based Snaplexes. To implement enhanced security, users may use the Enhanced Account Encryption feature. With this feature, the user designates his or her own key-pair for use within their Groundplex deployment, and does not share the private key with SnapLogic. The data is encrypted with a public key before it leaves the SnapLogic domain, then is decrypted with a private key on the Groundplex. In this scenario, account information is only usable in the Groundplexes where the private key is available. These Groundplexes can run in customer domain – either in the cloud or in their own data centers. Enhanced Account Encryption is not available for Cloudplex deployments.

Below are useful links to help you set up your SnapLogic environment:

[Requirements for Groundplex](#)

[Snaplex Installation on Windows](#)

[Snaplex Installation on Linux](#)

[Configuration Options](#)

[Networking Setup](#)

Recommended Naming Guidelines

Project Spaces

Org admins can set up project spaces to organize projects with associated assets, including accounts, files, tasks, and pipelines, within an org. In these project spaces, users and teams can access the appropriate folder to obtain the assets they need for a particular project. In larger organizations, we see project spaces created for specific teams or departments so they can view the pipelines and tasks they use.

Recommended Naming Guidelines

- All the names should be in title case
- Valid characters are alphanumeric and underscore (_)
- Leading or embedded blanks are not allowed

Project Naming Convention

How SnapLogic project names are assigned varies company by company. However, it is critical to have a uniform method of assigning project names so that users know where to find assets in their respective project spaces.

Below is a recommendation on how you can name your projects:

<Business Domain> or <Business Domain>_<Business Group>

Example:

a. <Business Domain>

HR
Finance
Sales
Marketing
Legal
CRM

b. <Business Domain>_<Business Group>

HR_Analytics
Sales_Corporate

c. <Business domain>_<Business Group> (Project spans across multiple Business domain)

Sales_SalesOps
HR_Recruitment

d. Product Specific or it Developing the Project for the Specific Product

Product<{Product/Application Name}>_<Group/Function Name>
ProductAnalytics_Tableau
Product_PartnerProgram

e. Short Lived Projects viz. Data Synchronization Projects

DataSync<Application name>
DataSync_CustomerAccounts

Pipeline Naming Convention

A pipeline contains a number of Snaps put together to perform a business process or to orchestrate a data flow between different end points.

There are two different types of pipelines:

- **Parent Pipeline:** This pipeline is exposed as an end point to the consumer applications, the parent pipeline can be a set of Snaps or a set of child pipelines
- **Child Pipeline:** This pipeline is a sub pipeline called or referenced in the parent pipeline, a single child pipeline can be referenced in more than one parent pipeline

Below are recommendations on how you can name your pipelines:

a. Parent Pipeline: Every parent pipeline can represent the business function with the appropriate action.

Example 1:

<verb><Business function>
getCustomerOrder

Example 2:

If there are multiple consumers using the same functionality and requires separate pipelines for each consumer, consumer system names can be appended to the pipeline name as follows. In case of multiple providers, provider and consumer names can be appended to the pipeline.

<verb><Business function>_< optional from system>_< optional to system>

From system and to system are optional elements
submitPaymentAuthorization_SAP_Coupa where data flow from SAP to Coupa

b. Child Pipeline: Child pipeline name can represent the specific business functionality. You can also prepend child pipelines with the same characters, such as sub_ or z_ as shown below.

Example:

child_<verb><specific business functionality>

child_getAddress
child_publishInvoice
child_submitOrder

File Names

These are files referenced in pipelines and recommend naming these files the same as the pipeline name, whether they are referenced in a parent pipeline or child pipeline.

There are some exceptions to files that are generated from third party applications and must be used as-is in SnapLogic. In this case, make sure this is documented in the Snap notes.

Below is an example of how to document original file name in the Snap notes:

child_getAddress child pipeline is using the file as a csv input, the recommended file name is *child_getAddress.csv*

submitPaymentAuthorization_SAP_Coupa parent pipeline is using a xml file as a input, the recommended file name is *submitPaymentAuthorization_SAP_Coupa.xml*

Accounts

SnapLogic accounts are usually the login credentials to the specific data provider or an application. These accounts are specific to the SnapLogic environment, and required to be created in each environment (Dev, Stage, QA and Prod). We recommend creating an account per SnapLogic Project.

Note: Use the “Accounts” tab within your project directory. An account can be created in the “Accounts” tab within the specific project.

Example:

<Application/Database>_<Project Name>

1. SFDC connectivity for HR project - “SFDC_HR”
2. ActiveMQ connectivity for Sales Compensation Project - “JMS_Sales_Compensation”

Tasks

Using a SnapLogic Task is a way to execute your pipelines using a fixed schedule or by accessing a URL (triggered).

Below is the recommended naming convention for tasks:

<Pipeline name>_Task

Example:

Task created for the getCustomerOrder pipeline - getCustomerOrder_Task

If there are multiple tasks required to create for the same pipeline then you can use the task type name in the task name.

Example:

Triggered task for getCustomerOrder pipeline - getCustomerOrder_Triggered_Task

Real-time (Ultra) task for getCustomerOrder pipeline - getCustomerOrder_RealTime_Task

For larger, distributed companies, we recommend limiting task creation to only SnapLogic Admins.

Integration Strategy and Execution

Prioritizing Integrations

Once you have set up SnapLogic, you are ready to start building integration pipelines and/or migrating your legacy integrations to SnapLogic. Create a chart to outline all the integrations you need to build or migrate. We recommend starting with low-impact, high-value integrations or new integrations that cause low-to-no disruptions to the business and/or existing integrations before moving onto more complex integrations.

How to Proceed

If you are migrating your legacy applications and integrations, we recommend using SnapLogic to take a fresh approach to these systems and integrations rather than performing a lift-and-shift. With SnapLogic, you can build simple, modular, interconnected pipelines that open new doors for extension and reusability. One distinct advantage of this approach is that you will be able to identify which modular pipeline breaks so that you can easily fix the pipeline and not have to review a monolithic pipeline.

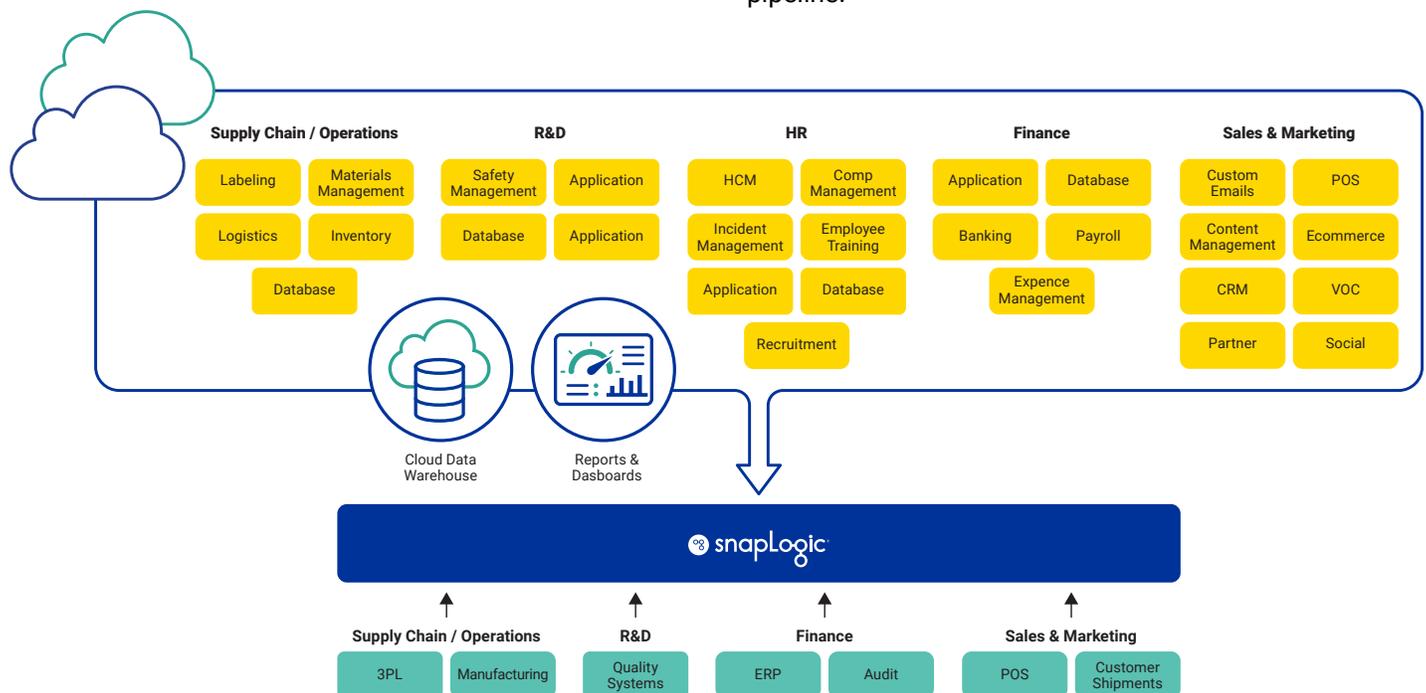


Figure 1: Hybrid integration architecture on a Groundplex

Work with your business stakeholders to build out your desired architecture that will help them automate business processes and data reporting. In some cases, you may completely run in the cloud or run a hybrid environment, connecting cloud applications with on-premises systems.

Considerations

- Consider migrating batch jobs instead of streaming, real-time data. Batch jobs are predictable and scheduled so you know when they are being processed. Then move on to processing streaming, real-time data as these are more complex, unpredictable, and they may touch many business processes that cannot be interrupted
- Consider building or migrating integrations that do not touch external data, such as customer data

Enterprise Automation Pipelines

We have pre-built Enterprise Automation pipelines that help you speed up integrations. Below are the pre-built pipelines based on business process and department. You will see more information about each business process and how to configure the pipeline on the links provided below. You can also find more pre-built pipelines on the [SnapLogic Community](#).

Human Resources

Business Process	Pipeline	Video
Employee Onboarding	Pipeline	Video
Employee Offboarding	Pipeline	Video
People Analytics	Pipeline	Video
Recruitment Automation	Pipeline	Video
Employee Data Management	Pipeline	
Create Request Ticket in ServiceNow When Employee in Workday Joins	Pipeline	
Scheduled Employee Data Batch Update From Oracle to Workday	Pipeline	
Insert New Employee into Workday	Pipeline	
Create or Update Skilljar User	Pipeline	

Sales and Marketing

Business Process	Pipeline
Move Salesforce Opportunity to NetSuite Sales Order	Pipeline
Sync Excel to Marketo	Pipeline
Create Opportunities in Microsoft Dynamics CSM and Create an Invoice in Workday if Sale Occurs	Pipeline
Get Leads From Marketing System and Write to a File	Pipeline

Finance and Accounting

Business Process	Video
Continuous Close	Video
Spend Management	Video

Customer Support

Business Process	Video
NetSuite Case to ServiceNow Incident	Pipeline
NetSuite Contact ServiceNow User	Pipeline
Salesforce Contact to ServiceNow User	Pipeline
Salesforce Case to ServiceNow Case	Pipeline
Creating a ServiceNow Incident From a JIRA Issue	Pipeline

Scaling Integrations and Beyond

We have seen many companies begin their SnapLogic journey with a centralized team building and managing SnapLogic integrations. Over time, they decentralize integrations and offer SnapLogic as a self-service model to enable their business counterparts to build their own integrations and scale.

Larger companies have created a self-service program with different levels of self-service and support depending on how technical the integrator is. In many cases, there are departments who will prefer integrations built for them, while other departments prefer more autonomy to build their own integrations.

As you start building out a self-service program, consider the doing following:

- Gather initial set of integration successes achieved by the central team
- Refer to [Sigma Framework for Operational Excellence](#) to standardize best practices
- Automate deployment using CI/CD to streamline the development lifecycle
- Outline the process of how users can get access to SnapLogic
- Create a Wiki for new users to get started with SnapLogic. This may include materials from SnapLogic, such as access to the SnapLogic University (online learning), How-to videos and exercises, SnapLogic documentation, SnapLogic Community, company-specific guidelines on how to use SnapLogic, Do's and Don'ts, and more
- Identify Admins or SnapLogic experts users can contact for additional support
- Identify type of integrations that users can build and manage on their own and mission-critical integrations that SnapLogic Admins need to be involved in

AI Enhancements

The [SnapLogic AI suite](#) of tools harnesses advanced AI to predict, map, and normalize your data. Then, leveraging GenAI technology, large language models (LLMs), and natural language processing, users can securely construct integrations and build automated applications.

SnapGPT

Rapidly create data pipelines, workflow prototypes and documentation, and perform data extraction and transformation tasks.

GenAI Builder

Build custom, LLM-powered applications and automated workflows for your business, without coding.

AutoIDP

Extract critical data and insights from documents and unstructured content to automate timely but tedious business processes.

AutoSuggest

Move faster with an intuitive recommendation engine for integration workflow and pipeline construction.

AutoPrep

Relieve your team from manual coding for data preparation and cleansing tasks.

To enable AI enhancements in your SnapLogic platform, reach out to your customer success manager for details.

SnapLogic Best Practices

Pipeline Design and Development

- **Do not assume that data preview provides a complete representation of the data**

Data preview is limited to the first 50 records of the data source by default, customizable in user settings up to 2000 records (be mindful that pulling a large number of records from a source, especially a verbose one like NetSuite or Workday, will have an adverse effect on the performance of your browser). All subsequent data previews down the pipeline will work only with that initial preview data set, so your actual resulting data may vary from what you see in the preview data.

- **Avoid large pipelines triggered by events**

When a pipeline is called in response to an event, the caller has to wait for the response until the entire pipeline completes. If the pipeline is large and takes a long time to complete, the caller may time out and mark a failure even though the pipeline is still running and processing data. All subsequent data previews down the pipeline will work only with that initial preview data set, so your actual resulting data may vary from what you see in the preview data.

- **Do not schedule a chain reaction**

When possible, separate a large pipeline into smaller pieces and schedule the individual pipelines independently. Distribute the execution of resources across the timeline and avoid a chain reaction so that you can quickly identify which pipeline may be erroneous instead of diagnosing and re-configuring a large pipeline.

- **If your pipeline fails, retry the validation**

If a pipeline fails for unknown reason, click **Save** after any modifications, then hold the Shift button on your keyboard while clicking the Validate Pipeline button to refresh the cached validation data. Just hitting the Validate Pipeline button may use the previously cached results.

For scheduled pipelines, close all open-ended Snaps (remove open output/error views).

Start ultra pipelines with listener Snaps like JMS Consumer.

Select **Ignore empty stream** in the JSON Formatter Snap to prevent generating empty output when no input data is provided.

You can find more SnapLogic best practices [here](#).

Pipeline Management

- **Rename Snaps when you place them in your pipeline**

By giving each Snap in your pipeline a unique name, it will be easier to identify the correct log information for that Snap in the runtime logs, especially if you are using multiple instances of the same Snap.

- **Maintain pipeline versions**

Accidentally deleting or making serious blunders in a pipeline could result in days of lost work. Some general guidelines for pipeline backups include exporting pipelines after significant milestones (major changes, a new release), renaming the pipeline file (.slp) to indicate the event, and storing the exported pipelines in an external repository such as GitHub, GitLab, or Bitbucket.

Tasks

- **Triggered Tasks: General Information**

Pipelines configured as triggered tasks can expose a maximum of **1 unconnected output view**.

The Task Execute Snap will timeout at the platform after **15 minutes**. This is irrespective of whether the pipeline is active or idle.

- **Triggered Tasks using the Cloud URL**

If the execution time of the task exceeds **15 minutes**, the platform will timeout the request and return an HTTP 504. This is enforced globally by the platform and cannot be modified.

By default, the remote request will wait until the pipeline execution is complete. Upon completion, the platform will return a response document containing the HTTP code of the pipeline exit status. If the pipeline fails during execution, additional system statistics may be returned in the same response document.

If the pipeline exposes an unconnected output view, the documents generated by the view will override the default response document.

- **Triggered tasks using an on-premises URL**

If the Snaplex node on which the task is running is patched to **mrc205** or higher, there is no platform-enforced restriction on execution time.

If the Snaplex node is not patched to at least **mrc205**, the task may fail after **10 minutes**. The 10 minute timeout for local url (pre-mrc205) happens for pipelines which are not active, if the pipeline output view is continuously streaming results, the timeout does not apply.

By default, the remote request will return asynchronously after starting the pipeline. The platform will not return a default response document.

If the pipeline exposes an unconnected output view, the remote request will wait until the pipeline execution is complete. The documents generated by the view will become the response to the request.

Administration

- **Do not use an admin user for development**

Create a separate user login for each developer. By default, a project will be created for them, but you can also give them either full access or only read and execute permissions on other projects. Using the admin user would give you access to all projects.

- **Create accounts in individual projects, not in the shared project**

Accounts store credentials to access other applications. Unless it is an account you know everyone in your organization needs, do not save it in the Shared project. Instead, create projects for specific applications and store the Account in that project.

Additional Resources

[SnapLogic Community](#) – Join the Integration Nation to exchange ideas, tips and best practices with other SnapLogic users, employees and partners

[SnapLogic Documentation:](#)

[SnapLogic Administration and Configuration](#)

[Snap References](#)

[SnapLogic Release Notes](#)

[Ultra tasks](#)

[Enhanced Account Encryption](#)

[Resumable Pipelines](#)

[API Management](#)

[SnapLogic AI](#)

[SnapGPT: integrations on command](#)

[GenAI Builder: automated applications on command](#)

For more information, contact your dedicated SnapLogic account manager.

Connect your entire enterprise and automate all your end-to-end business processes with SnapLogic today. snaplogic.com/contact-us

Enterprise Automation Pioneers



SnapLogic is the leader in **generative integration**. As a pioneer in AI-led integration, the SnapLogic Platform accelerates digital transformation across the enterprise and empowers everyone to integrate faster and easier. Whether automating business processes, democratizing data, or delivering digital products and services, SnapLogic enables you to simplify your technology stack and take your enterprise further. Thousands of enterprises around the globe rely on SnapLogic to integrate, automate and orchestrate the flow of data across their business. Join the **generative integration** movement at snaplogic.com.