# Ask the Expert: Architecture Q&A Session

Regan Knepp, Community Events & User Groups Lead, Boomi

boomi

## Join us at Boomi World 2025 as we showcase the UNLIMITED power of AI-driven integration and automation

From groundbreaking announcements and inspiring keynotes to hands-on sessions and vibrant networking, don't miss out on Boomi World 2025. Lock in Early Bird savings by registering today!

# Boomi World 2025

Dallas, Texas
May 12–15, 2025

**Secure your spot at boomiworld.com**

## WHY ATTEND BOOMI WORLD?

**Stay Ahead of the Curve**
Hear directly from Boomi executives and technologists on our vision for AI-empowered business.

**Get Trained and Certified**
Sign up for pre-conference training and get certified on key Boomi technologies, equipping you with the skills needed to navigate today's fast-changing technology landscape.

**Build New Partnerships**
The Partner Summit is a must-attend event for Boomi partners. And be sure to check our sponsorship opportunities.

**Sharpen Your Expertise**
With breakout sessions tailored for developers, integration architects, business users, and executives, Boomi World offers valuable content for everyone. Check the online agenda often for the latest updates.

**Accelerate Inspiration**
Get inspired by innovative companies across industries that are accelerating business outcomes, taming complexity, and transforming their organizations using the Boomi platform.

# Upcoming February Events

## Virtual Events:

### ● Ask the Expert: Architecture Office Hours

**Date:** February 4, 2025, 1:00pm EST
**Registration:** Please Register here
**Agenda:** Join us for Boomi Office Hours, a live Q&A session with Boomi Professional Services experts! Submit your questions when you register or bring them to the session.

### ● Higher Education User Group Meetup

**Date:** February 12 2025, 1:00pm EST
**Registration:** Please Register here
**Agenda:** Join us as we meet for our monthly virtual Higher Education User Group!

### ● Document and Process Properties

**Date:** February 18, 2025, 1:00pm EST
**Registration:** Please Register here
**Agenda:** What is a Boomi Document? What's the difference between Document Properties and Process Properties, and how do I use them? Join Senior Integration Captain Brad Detlefsen as he explains these details, talks through use cases, and reveals some tips and tricks.

https://usergroups.boomi.com/

# Jarosław Gromadziński

## Strategy Advisor / Senior Architect

- Based in Warsaw, Poland
- Started in Boomi PSO EMEA in March 2021
- Member of Boomi's Professional Services Organization and Global Architectural Team
- Specialize in defining strategies in the scope of integration, automatization, and data management area, designing and deploying solutions based on Boomi solutions
- Help customers bring a strategic mindset with expertise in emerging trends such as AI/ML.
- Strong background in project leadership, digital transformation and data management.

*Being in Boomi since March 2021*
*I completed over 120 projects for 103 different customers from all over the world.*

# Questions

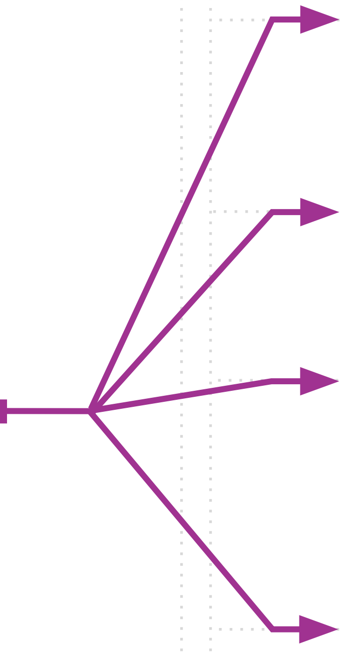
**Question:**

What are some
Boomi Architecture best practices?

# What are some Boomi Architecture best practices?

## Boomi Runtime Blueprint - Key Golden Standards

The **Golden Standard** refers to best practices and guidelines that **help organizations** effectively set up, manage, and optimize Boomi.



### Runtime Selection: Be Strategic

WHEN IMPLEMENTING BOOMI'S RUNTIME COMPONENTS,
IT'S ESSENTIAL TO CHOOSE THE RIGHT RUNTIME TYPE BASED ON ORGANIZATION'S SPECIFIC NEEDS AND USE CASES

### Landing Place Empowerment

BEGIN WITH A THOROUGH ASSESSMENT TO MATCH HARDWARE AND SOFTWARE RESOURCES WITH YOUR PERFORMANCE NEEDS.

### Disaster Recovery and Fault Tolerance

Design runtime configurations that include plans for disaster recovery and failover strategies to ensure business continuity in case of failures.

### Runtime Fine-Tuning

Optimizing your Boomi Runtime is vital for ensuring high performance, scalability, and cost-effectiveness.

# What are some Boomi Architecture best practices?

Runtime Selection: Be Strategic

### Understand Integration Requirements

Assess the integration requirements, including data volume, frequency of data transfers, and the complexity of integrations. This helps in determining which runtime might be the most efficient for your organization.

### Evaluate the Runtime Types Available

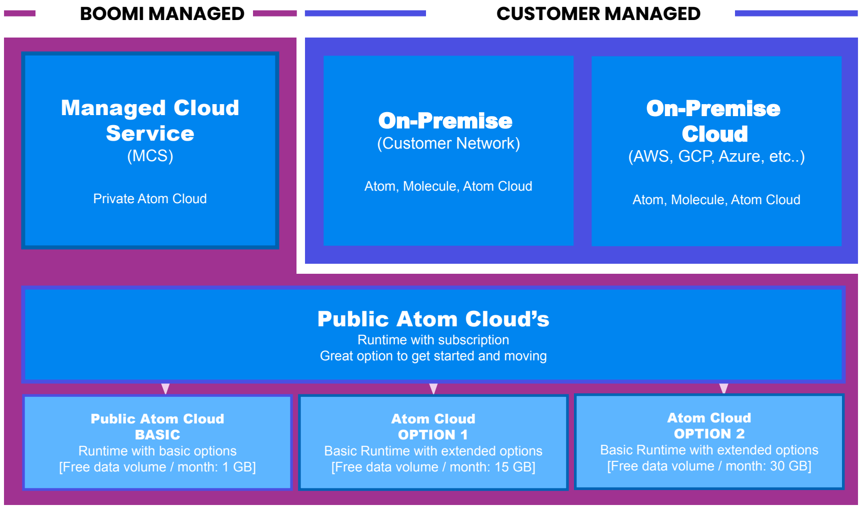
Consider different Runtime environments available in Boomi, such as Single Atom, Molecule, Managed Cloud Services (MCS), and the Boomi Atom Cloud. Each has its advantages depending on the specific use case and organizational needs.

### Security and Compliance

Understand the security implications of each runtime option, including data encryption, access controls, and compliance with data protection regulations. Ensure that the chosen runtime complies with organization's regulatory requirements..

### Plan for Maintenance and Support

Evaluate the internal resources available for supporting and maintaining Runtime infrastructure. If the organization lacks the bandwidth or expertise for heavy maintenance, a Boomi cloud-based solution may be more practical.



BOOMI MANAGED

CUSTOMER MANAGED

**Managed Cloud Service**
(MCS)
Private Atom Cloud

**On-Premise**
(Customer Network)
Atom, Molecule, Atom Cloud

**On-Premise Cloud**
(AWS, GCP, Azure, etc..)
Atom, Molecule, Atom Cloud

**Public Atom Cloud's**
Runtime with subscription
Great option to get started and moving

**Public Atom Cloud BASIC**
Runtime with basic options
[Free data volume / month: 1 GB]

**Atom Cloud OPTION 1**
Basic Runtime with extended options
[Free data volume / month: 15 GB]

**Atom Cloud OPTION 2**
Basic Runtime with extended options
[Free data volume / month: 30 GB]

# What are some Boomi Architecture best practices?

## Runtime Selection: Be Strategic

Boomi's Runtime **T-Shirt Sizing** presents a Boomi experience, structured approach to **estimate** Runtime selection.

### Recommended Boomi Runtime T-shirt sizing

### Boomi Atoms

| Size | Transactions/day | Nodes | Node Size |
|---|---|---|---|
| **Small** | <500k txns/day | 1 | 4x16x100* |
| **Medium** | 500k-1m | 1 | 8x32x200 |
| **Large** | 1m+ | 1 | 16x64x200 |

### Clustered Boomi Molecules / Private Cloud Molecules

| Size | Transactions/day | Nodes | Node Size | File Share |
|---|---|---|---|---|
| **Small** | ~500k & < 1 GB batch | 3 | 4x16x100 | 200 GB / 50 MiBs |
| **Medium** | 500k-1m & 1-2 GB batch | 3-5 | 8x32x200 | 500 GB / 50-150 MiBs |
| **Large** | 1m+ & 2 GB batch | 5-10 | 8x32x200 | 1TB / 150+ MiBS |

### Boomi APIM Gateway

| Size | Transactions/day | Nodes | Node Size | File Share |
|---|---|---|---|---|
| **Small** | 100k API | 3 | 4x16x100 | 200 GB / 50 MiBs |
| **Medium** | < 1m API | 3-5 | 8x32x200 | 500 GB / 50-150 MiBs |
| **Large** | > 1m API | 3-5 | 8x32x200 | 1TB / 150+ MiBS |

**Node Size** - (CPU x RAM x Disk)

**Note.** All calculations may vary based on the amount of work each integration performs on the Runtime.

# What are some Boomi Architecture best practices?

## Landing Place Empowerment

### Consistent OS and Machines Architecture

Standardize all machines hosting Boomi Atoms or Molecule nodes to have a consistent OS architecture (e.g., same Linux or Windows version) and hardening approach to reduce compatibility issues and streamline configuration management.

### Utilization of Dedicated Machines

Allocate dedicated machines for each Boomi Atom or Molecule node to isolate runtime environments, improving security and resource management.

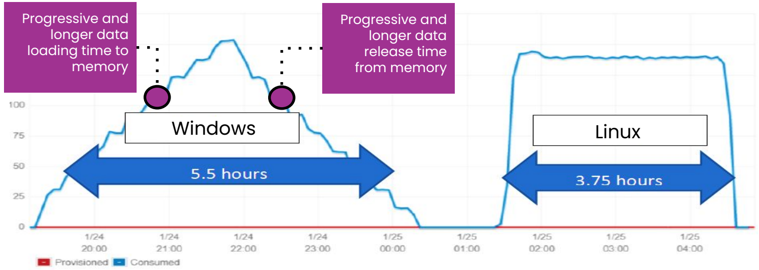
### Efficient storage space architecture

Use separate disks for the OS and Boomi runtime installation/working directory to improve performance by avoiding disk I/O contention, simplify resource management, and ensure sufficient space for high-volume scenarios, reducing the risk of system failures.

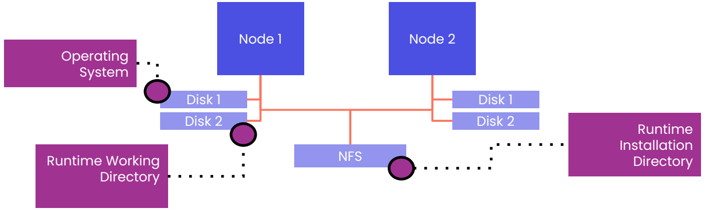
### Enhanced Security and Compliance

Using dedicated VMs enhances security by isolating processes, reducing cross-service risks, while ensuring compliance through encrypted disk partitions for sensitive data.

***Operating system considerations***



***Storage layer configuration***



# What are some Boomi Architecture best practices?

## Landing Place Empowerment

**Establishing** adequate compute resources is **critical** for supporting landing places.

### Recommended Instance Types

| Runtime Type | Size | Amazon EC2 Instance Types | Azure Virtual Machines |
|---|---|---|---|
| **Atom**<br>Mixed Batch/Real-Time | **Small** | m5.xlarge | Standard_D4s_v5 |
| | **Medium** | m5.2xlarge | Standard_D8s_v5 |
| | **Large** | m5.4xlarge | Standard_D16s_v5 |
| **Molecule & Atom Cloud**<br>Mixed Batch/Real-Time | **Small** | m5.xlarge | Standard_D4s_v5 |
| | **Medium** | m5.2xlarge | Standard_D8s_v5 |
| | **Large** | m5.2xlarge | Standard_D8s_v5 |
| **Molecule & Atom Cloud**<br>Real-Time Optimized | **Small** | c5.2xlarge | Standard_F8s_v2 |
| | **Medium** | c5.4xlarge | Standard_F16s_v2 |
| | **Large** | c5.4xlarge | Standard_F16s_v2 |
| **APIM Gateway** | **Small** | c5.2xlarge | Standard_F8s_v2 |
| | **Medium** | c5.4xlarge | Standard_F16s_v2 |
| | **Large** | c5.4xlarge | Standard_F16s_v2 |

**M5 Instance Family** instances offer a balance of compute, memory, and networking resources for a broad range of workloads.

**C5 Instance Family** is designed for compute-intensive workloads

**Fsv2 Instance Family** is optimized for compute-intensive tasks.

**Dsv5 Instance Family** is optimized for general-purpose workloads.

# What are some Boomi Architecture best practices?

Disaster Recovery and Fault Tolerance

## Comprehensive Disaster Recovery (DR) Plan

Develop a detailed disaster recovery plan that outlines recovery time objectives (RTO) and recovery point objectives (RPO) to meet organizational requirements for uptime and data protection.

## Data Backup Strategies

Implement automated and regular data backup strategies for configurations, log files, and other essential data. Backups should be stored in a secure, offsite location to protect against data loss due to primary site failures.

## Health Check Mechanisms

Implement health check mechanisms to continuously monitor the status of Runtimes. Automated health checks can identify and alert teams to system issues before they escalate into failures.

## Fault Tolerant Architecture

Design integration processes with decoupled components to isolate faults and prevent cascading failures. This approach helps in maintaining the overall system stability.

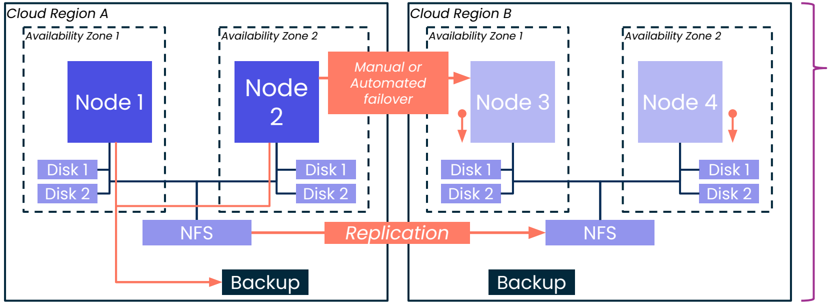
## Disaster Recovery and Backup Considerations

01 **Single Cluster Spanning Data Centers**
Because of network latency, try to place Runtime clusters in one data center.

02 **Active/Active Multi-Region Deployment**
Employ a strategy where multiple instances of Boomi Atoms/separate Molecule or Boomi Cloud clusters are deployed across different geographical regions, allowing for load balancing, redundancy, and improved availability.

03 **Active/Passive Multi-Region Deployment**
Employ a strategy where the Molecule operates with active instances processing requests and managing integrations. The passive instances (or backups) remain idle under normal conditions but are fully configured and ready to take over if the active node fails.

… example configuration based on Public Cloud IaaS resources:

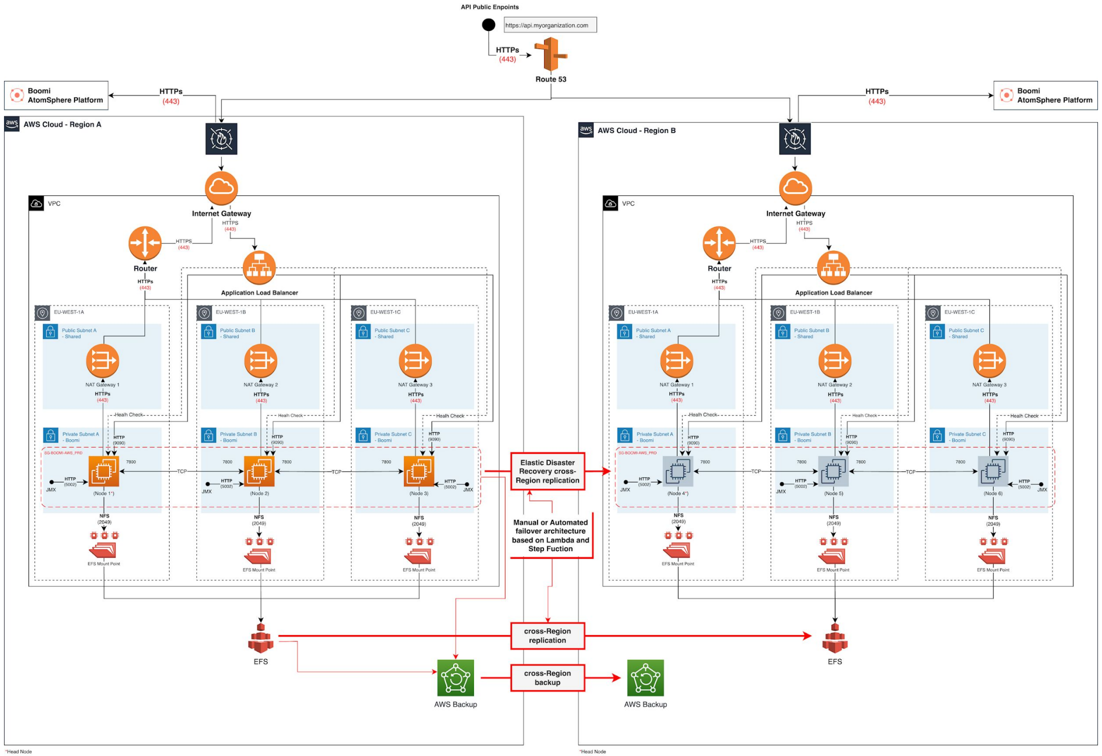


Decomposing the Molecule cluster into multiple AZs in one Region gives us HA + DR in one

# What are some Boomi Architecture best practices?

## Disaster Recovery and Fault Tolerance

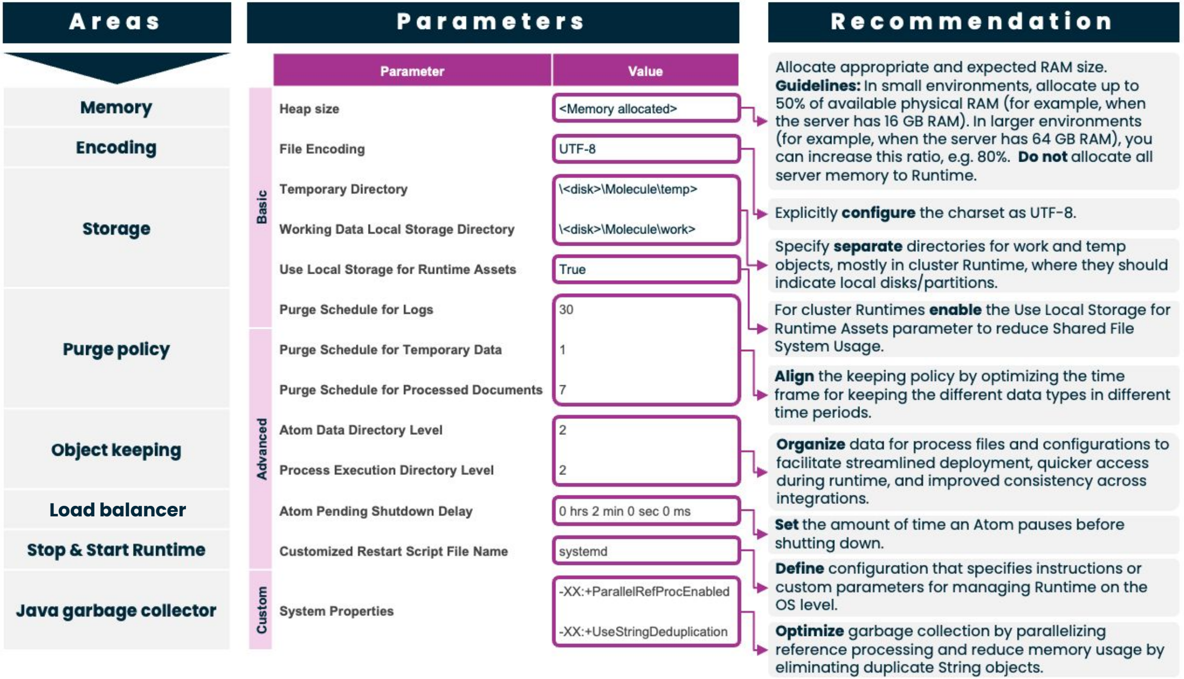
What could an example **Disaster Recovery** configuration based on AWS Public Cloud resources look like?



# What are some Boomi Architecture best practices?

## Runtime Fine-Tuning

An **important** element of configuration is also the **fine-tuning** of the Runtime environment after its implementation.



| Areas | Parameters | | | Recommendation |
|---|---|---|---|---|
| | | Parameter | Value | |
| Memory | Basic | Heap size | <Memory allocated> | Allocate appropriate and expected RAM size. **Guidelines:** In small environments, allocate up to 50% of available physical RAM (for example, when the server has 16 GB RAM). In larger environments (for example, when the server has 64 GB RAM), you can increase this ratio, e.g. 80%. **Do not** allocate all server memory to Runtime. |
| Encoding | | File Encoding | UTF-8 | Explicitly **configure** the charset as UTF-8. |
| Storage | | Temporary Directory<br>Working Data Local Storage Directory | \<disk>\Molecule\temp><br>\<disk>\Molecule\work> | Specify **separate** directories for work and temp objects, mostly in cluster Runtime, where they should indicate local disks/partitions. |
| | | Use Local Storage for Runtime Assets | True | For cluster Runtimes **enable** the Use Local Storage for Runtime Assets parameter to reduce Shared File System Usage. |
| Purge policy | | Purge Schedule for Logs | 30 | **Align** the keeping policy by optimizing the time frame for keeping the different data types in different time periods. |
| | Advanced | Purge Schedule for Temporary Data | 1 | |
| | | Purge Schedule for Processed Documents | 7 | |
| Object keeping | | Atom Data Directory Level<br>Process Execution Directory Level | 2<br>2 | **Organize** data for process files and configurations to facilitate streamlined deployment, quicker access during runtime, and improved consistency across integrations. |
| Load balancer | | Atom Pending Shutdown Delay | 0 hrs 2 min 0 sec 0 ms | **Set** the amount of time an Atom pauses before shutting down. |
| Stop & Start Runtime | | Customized Restart Script File Name | systemd | **Define** configuration that specifies instructions or custom parameters for managing Runtime on the OS level. |
| Java garbage collector | Custom | System Properties | -XX:+ParallelRefProcEnabled<br>-XX:+UseStringDeduplication | **Optimize** garbage collection by parallelizing reference processing and reduce memory usage by eliminating duplicate String objects. |

**Question:**

Best practices around integrating with AWS EventBridge. Is there specific guidance on message format, how to insert traces, etc. What would be the pros/cons of utilizing cloudevent protocol to formalize these payloads?

# Best practices around integrating with AWS EventBridge. Is there specific guidance …

- **Boomi documentation**
  Amazon EventBridge connector

  https://help.boomi.com/docs/Atomsphere/Integration/Connectors/int-Amazon_EventBridge_connector_ea3d78b3-432b-42d8-ab7d-1083f4e29394

- **Boomi Process Library**
  Amazon EventBridge Basics

- **Community**
  Getting Started with the Amazon EventBridge Connector
  Recipe: Connect with Amazon EventBridge
  AWS EventBridge Connector not Escaping Characters as Expected

**Additionally…**

- **Error Codes and Formats**
  Please see this AWS document for an update list of Error Codes and other Formats.

  https://docs.aws.amazon.com/eventbridge/latest/APIReference/CommonErrors.html

# Best practices around integrating with AWS EventBridge. Is there specific guidance …

### Message Format

When integrating Boomi with Amazon EventBridge, it's crucial to adhere to the message format requirements set by AWS to ensure successful transmission and processing of events:

- Entries: An array of event entries, with each entry containing the required fields.

```
{
  "Entries": [
    {
      "Source": "string",
      "DetailType": "string",
      "Detail": "string",
      "EventBusName": "string",
      "Resources": ["string"]
    }
  ]
}
```

### Required Fields

Each entry within the Entries array must include the following fields:

- Source (String)
  - Identifies the origin of the event (e.g., com.boomi.myprocess).
  - Use a consistent naming convention (e.g., reverse domain name style).
- DetailType (String)
  - Describes the type of event (e.g., OrderPlaced, ProcessCompleted).
  - Helps consumers filter and handle events appropriately.
- Detail (JSON String)
  - Contains the payload of the event. This is the business-specific data (e.g., order details, user data).

```
{
  "Detail": "{\"orderId\":\"12345\",\"status\":\"completed\"}"
}
```

- EventBusName (String)
  - Specifies the name of the event bus to which the event will be sent.
  - Use the default event bus (default) or custom event bus names as configured in AWS.
- Resources (Optional Array of Strings)
  - Identifies the resources involved in the event. This field is optional and may include ARNs (Amazon Resource Names).

# Best practices around integrating with AWS EventBridge. Is there specific guidance …

## Pros of Utilizing CloudEvents Protocol

- **Standardization Across Systems**
  CloudEvents provides a vendor-neutral specification for event data, enabling seamless interoperability between various platforms, including Boomi, AWS, and others.
- **Enhanced Interoperability**
  Many event-driven platforms and services support CloudEvents natively, which can simplify integration and maintenance.
- **Extensibility**
  The CloudEvents specification allows you to define custom attributes while maintaining compatibility with the base structure.
- **Improved Event Routing and Filtering**
  CloudEvents has predefined attributes such as type, source, and subject, making it easier to filter and route events in EventBridge.
- **Future-Proofing**
  As CloudEvents adoption grows, using the standard can help avoid compatibility issues and make transitions to other platforms smoother.

## Cons of Utilizing CloudEvents Protocol

- **Increased Complexity**
  Adopting CloudEvents introduces additional steps in designing, transforming, and validating payloads in Boomi.
- **Overhead in Payload Size**
  CloudEvents adds metadata fields (e.g., specversion, type, source, id) to every event, increasing the payload size.
- **Compatibility Adjustments**
  Some legacy systems or services may not support CloudEvents, requiring additional transformations to and from the format.
- **Boilerplate Fields**
  Certain fields in CloudEvents (like specversion or time) are often required but may not add immediate value in simpler use cases.
- **Tooling Gaps**
  While AWS EventBridge supports CloudEvents, tooling in Boomi may require custom scripting or transformations to fully align with the format.

### Conclusion

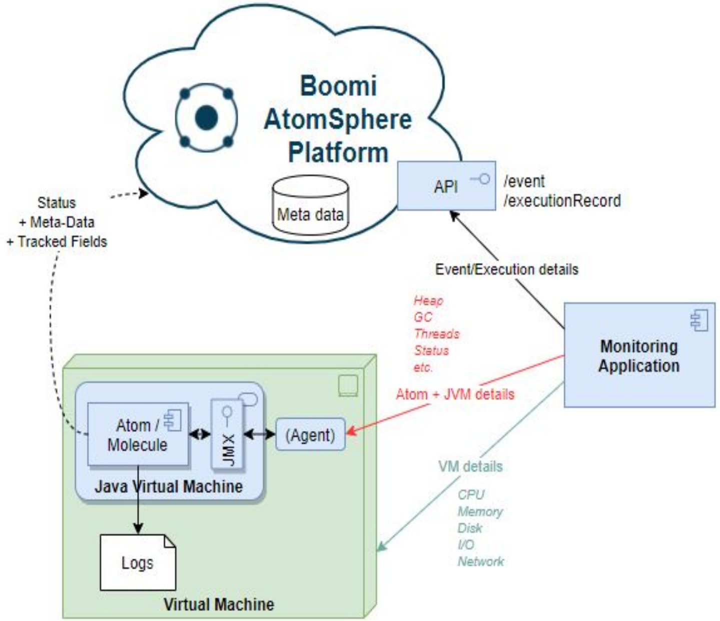
The decision to use CloudEvents with Boomi and AWS EventBridge depends on the complexity of your architecture and your need for standardization. While CloudEvents offers interoperability and extensibility, the trade-offs in complexity and overhead should be carefully considered based on the integration's scope and future scalability needs.

## Questions:

How to best monitor for infra failures?

# How to best monitor for infra failures?

## Owners of Atoms, Molecules, or Atom Clouds should monitor the machine(s) on which they run, as well as the applications' status



In order to look for signs of problems and gather performance statistics, it is recommended to monitor elements such as input, output, disk space, the number of nodes online, Atom memory, machine memory, how many processes are running per machine, etc.

### Key elements recommended for monitoring

1. Operating System/Hardware
2. Boomi Runtimes apps
3. Integration Processes

# How to best monitor for infra failures?

## Components recommended to be monitored

### Virtual Machine

Standard metrics are to be monitored (and trended for capacity planning) here, including:

1. CPU usage
2. Memory usage
3. Disk space usage
4. Inodes usage (Linux only)
5. Disk I/O average latency (read/write)
6. Network adapter health

### Java Virtual Machine

The Java Virtual Machines and the Boomi runtime nodes both expose data via JMX (MBeans) which should be monitored using a compliant tool and trended over time. Below is the list of meaningful properties available, and in bold the ones which are highly recommended.

See: https://help.boomi.com/bundle/integration/page/r-atm-System_monitoring_with_JMX.html

It is recommended to monitor minimum **General JVM status parameters**

| Attribute | Format | Description |
|---|---|---|
| **JVM heap used** | Numeric, in bytes | The number of bytes of memory that the Java virtual machine (JVM) is currently using. This value should be monitored to look for problematic scenarios, such as a point in time stress (the Atom suddenly consumes a lot more memory than normal) or a general increase in memory over time. Increases over time could indicate a memory leak, depending on how frequently the Atom is restarted. It is more likely that they indicate changes in Atom usage, which might necessitate increasing the Atom's memory allocation or adjusting the workload. |
| **JVM heap max** | Numeric, in bytes | The maximum size for the heap. The JVM will never attempt to use more heap memory than this value. |
| **JVM non-heap used** | Numeric, in bytes | The number of bytes of non-heap memory that the JVM is currently using. |
| **JVM non-heap max** | Numeric, in bytes | The maximum size for non-heap memory. |
| **JVM thread count** | Numeric | The number of active threads in the JVM. This value should be monitored for point in time anomalies and trends over time. |
| **JVM uptime** | Numeric, in milliseconds, cumulative | The amount of time that the JVM has been running. |
| **JVM CPU time** | Numeric, in nanoseconds, cumulative | The CPU time that has been used by the JVM. This value might not be available on all platforms, in which case -1 is returned. |
| **Garbage collection count** | Long, counter, cumulative | The total number of garbage collections that have occurred in the JVM. |
| **Garbage collection time** | Long, in milliseconds, cumulative | The accumulated garbage collection time for the JVM. |

# How to best monitor for infra failures?

Functional and nonfunctional Monitoring capabilities on the Boomi Integration Layer

## Default Boomi AtomSphere Platform Capabilities

1. **Dashboards (UI)**

   The Dashboards provide historical and summary information about process executions and inbound requests.

2. **Process Reporting (UI)**

   The Process Reporting console provides detailed information about process executions including process logs and documents.

3. **AtomSphere RSS Feeds (Event Framework)**

   Platform Events are available via RSS feed per Atom or at the account level. RSS are the simplest approach for receiving platform Events. If you have an incident reporting system or monitoring tool that can consume RSS feeds, consider using them.

   **IMPORTANT:** RSS feeds are not updated in real-time, so these should NOT be used for real-time monitoring.

4. **Platform API (Event Framework)**

   Events can be consumed programmatically via the Event API. There is no UI for the list of Events.

   **IMPORTANT:** Event records are available for 7 days.

5. **AtomSphere Email Alerts (Event Framework)**

AtomSphere users can choose to receive platform Events via email. Users can choose to subscribe to Events by Log Level and Type.

- Alert subscriptions are configured per user, not per account.
- Administrators cannot subscribe users to emails on their behalf.
- Email alerts are batched and sent from the AtomSphere platform periodically every few minutes to avoid triggering spam filters.
- The subject and format of the email messages cannot be modified.
- Email alerts are NOT sent for Events occurring in "Test" Environments (i.e. Environments classified as "Test„).
- For Low Latency executions, process.execution Events are NOT generated, even for process errors and Exception shapes, so no email alerts will be delivered.
- Email alerts do not consume a connection license.

**IMPORTANT**: It is not recommended to use email alerts as primary alerting mechanisms, these should be used for notification purposes only.

## Additional features and possibilities

1. **Boomi Insights**

   Boomi Insights provides a holistic view of key metrics across the entire Boomi Integration landscape so that customers can easily monitor, troubleshoot, and optimize all their Boomi operations.

   - Data is stored on the Boomi website.
   - Additional license/subscription required

2. **Custom (Built into Process Flow)**

   Custom error handling and notification logic can be built into the process design/flow as required. e.g. **Custom Error handling Integration Framework**

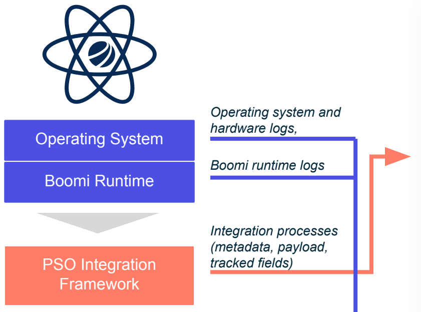
# How to best monitor for infra failures?

## Example Common Monitoring

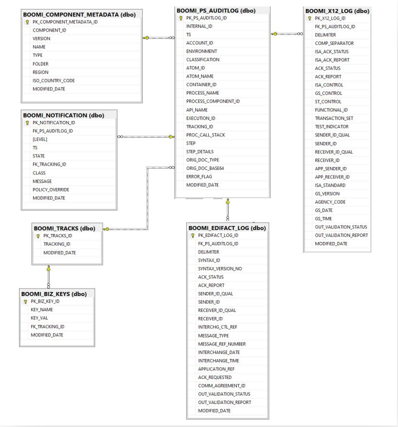
Step 1 – Data acquisition

Step 2 – Error handling Framework data persist and structurize

Step 3 – Data analyze and Visual


Operating System
Boomi Runtime
PSO Integration Framework
Operating system and hardware logs,
Boomi runtime logs
Integration processes (metadata, payload, tracked fields)

*Error handling technique*



| BOOMI_COMPONENT_METADATA (dbo) |
|---|
| PK_COMPONENT_METADATA_ID |
| COMPONENT_ID |
| VERSION |
| NAME |
| TYPE |
| FOLDER |
| REGION |
| ISO_COUNTRY_CODE |
| MODIFIED_DATE |

| BOOMI_PS_AUDITLOG (dbo) |
|---|
| PK_PS_AUDITLOG_ID |
| INTERNAL_ID |
| TS |
| ACCOUNT_ID |
| ENVIRONMENT |
| CLASSIFICATION |
| ATOM_ID |
| ATOM_NAME |
| CONTAINER_ID |
| PROCESS_NAME |
| PROCESS_COMPONENT_ID |
| API_NAME |
| EXECUTION_ID |
| TRACKING_ID |
| PROC_CALL_STACK |
| STEP |
| STEP_DETAILS |
| ORIG_DOC_TYPE |
| ORIG_DOC_BASE64 |
| ERROR_FLAG |
| MODIFIED_DATE |

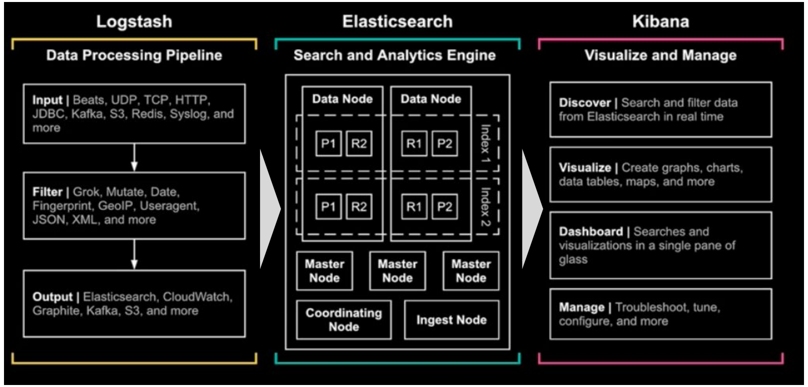
| BOOMI_X12_LOG (dbo) |
|---|
| PK_X12_LOG_ID |
| FK_PS_AUDITLOG_ID |
| DELIMITER |
| COMP_SEPARATOR |
| ISA_ACK_STATUS |
| ISA_ACK_REPORT |
| ACK_STATUS |
| ACK_REPORT |
| ISA_CONTROL |
| GS_CONTROL |
| ST_CONTROL |
| FUNCTIONAL_ID |
| TRANSACTION_SET |
| TEST_INDICATOR |
| SENDER_ID_QUAL |
| SENDER_ID |
| RECEIVER_ID_QUAL |
| RECEIVER_ID |
| APP_SENDER_ID |
| APP_RECEIVER_ID |
| ISA_STANDARD |
| GS_VERSION |
| AGENCY_CODE |
| GS_DATE |
| GS_TIME |
| OUT_VALIDATION_STATUS |
| OUT_VALIDATION_REPORT |
| MODIFIED_DATE |

| BOOMI_NOTIFICATION (dbo) |
|---|
| PK_NOTIFICATION_ID |
| FK_PS_AUDITLOG_ID |
| [LEVEL] |
| TS |
| STATE |
| FK_TRACKING_ID |
| CLASS |
| MESSAGE |
| POLICY_OVERRIDE |
| MODIFIED_DATE |

| BOOMI_TRACKS (dbo) |
|---|
| PK_TRACKS_ID |
| TRACKING_ID |
| MODIFIED_DATE |

| BOOMI_BIZ_KEYS (dbo) |
|---|
| PK_BIZ_KEY_ID |
| KEY_NAME |
| KEY_VAL |
| FK_TRACKING_ID |
| MODIFIED_DATE |

| BOOMI_EDIFACT_LOG (dbo) |
|---|
| PK_EDIFACT_LOG_ID |
| FK_PS_AUDITLOG_ID |
| DELIMITER |
| SYNTAX_ID |
| SYNTAX_VERSION_NO |
| ACK_STATUS |
| ACK_REPORT |
| SENDER_ID_QUAL |
| SENDER_ID |
| RECEIVER_ID_QUAL |
| RECEIVER_ID |
| INTERCHG_CTL_REF |
| MESSAGE_TYPE |
| MESSAGE_REF_NUMBER |
| INTERCHANGE_DATE |
| INTERCHANGE_TIME |
| APPLICATION_REF |
| ACK_REQUESTED |
| COMM_AGREEMENT_ID |
| OUT_VALIDATION_STATUS |
| OUT_VALIDATION_REPORT |
| MODIFIED_DATE |



**Input** | Beats, UDP, TCP, HTTP, JDBC, Kafka, S3, Redis, Syslog, and more

**Filter** | Grok, Mutate, Date, Fingerprint, GeoIP, Useragent, JSON, XML, and more

**Output** | Elasticsearch, CloudWatch, Graphite, Kafka, S3, and more

**Discover** | Search and filter data from Elasticsearch in real time

**Visualize** | Create graphs, charts, data tables, maps, and more

**Dashboard** | Searches and visualizations in a single pane of glass

**Manage** | Troubleshoot, tune, configure, and more

## Question:

How to increase max execution time in the public cloud?

# How to increase max execution time in the public cloud?

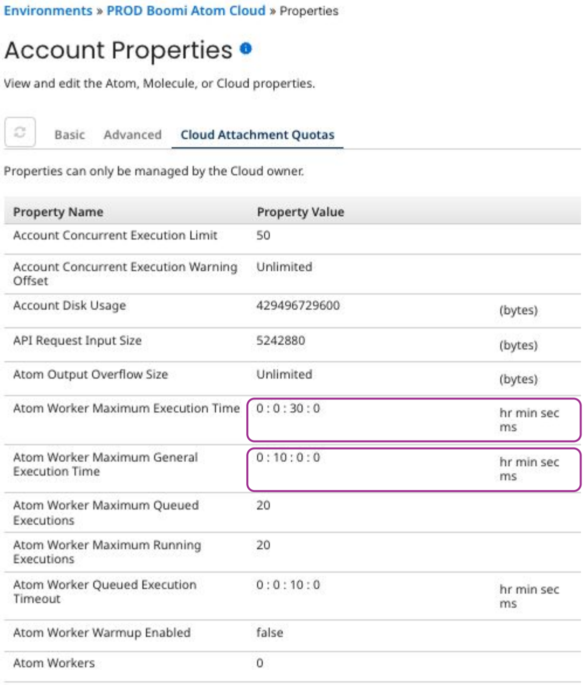
In general, the **Maximum Execution Time** parameters for the **Boomi Atom Clouds** (public runtime managed by Boomi) **cannot be modified**, they are permanently assigned to the Boomi Atom Cloud Runtime.

However, in many cases, it's possible to configure a different **"process mode"** to accommodate **different needs.**

What's the situation that's prompting this question?

For example:

- Scheduled based process longer than 24 hours?
- Listener based process longer than 30 seconds (or 10 minutes in some cases)?



Environments » PROD Boomi Atom Cloud » Properties

## Account Properties

View and edit the Atom, Molecule, or Cloud properties.

Properties can only be managed by the Cloud owner.

| Property Name | Property Value | |
|---|---|---|
| Account Concurrent Execution Limit | 50 | |
| Account Concurrent Execution Warning Offset | Unlimited | |
| Account Disk Usage | 429496729600 | (bytes) |
| API Request Input Size | 5242880 | (bytes) |
| Atom Output Overflow Size | Unlimited | (bytes) |
| Atom Worker Maximum Execution Time | 0 : 0 : 30 : 0 | hr min sec ms |
| Atom Worker Maximum General Execution Time | 0 : 10 : 0 : 0 | hr min sec ms |
| Atom Worker Maximum Queued Executions | 20 | |
| Atom Worker Maximum Running Executions | 20 | |
| Atom Worker Queued Execution Timeout | 0 : 0 : 10 : 0 | hr min sec ms |
| Atom Worker Warmup Enabled | false | |
| Atom Workers | 0 | |

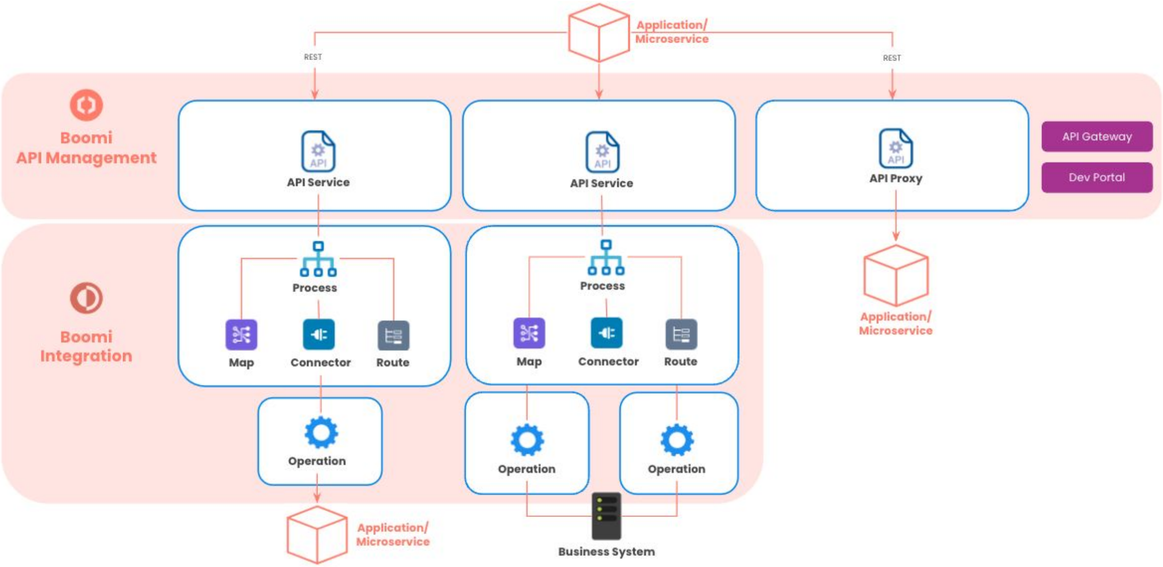
https://help.boomi.com/docs/Atomsphere/Integration/Getting%20started/c-atm-The_Boomi_Atom_Clouds_6c3bfab5-2fdc-4fde-a7cb-5f0c591a9f08

**Question:**

How do we define Micro Services framework within Boomi and expose to outside parties?

# How do we define Micro Services framework within Boomi and expose to outside parties?

**Defining a microservices** framework within Boomi involves designing modular, reusable, and scalable **integrations** that align with the principles of microservices. You can expose these microservices to external parties using Boomi **API Management / API Control Plane**.



## Question:

Although there are lots of articles scattered around the net for messaging patterns supported via integration, do we have a Boomi Community article or documentation which would be helpful for any architects to visit?

# […]do we have a Boomi Community article or documentation which would be helpful for any architects to visit?

Yes, the Boomi Community offers several resources that can assist architects in understanding and implementing messaging patterns within Boomi integrations. Here are some notable articles:

- **Boomi Blueprint: Integration Patterns:** This article discusses various integration patterns, including the Publish/Subscribe pattern, which ensures that integration components are reusable and that source and target applications are decoupled from each other.

  See: COMMUNITY.BOOMI.COM

- Series of **Sync Strategies** Articles**:** Four-part series of articles that describes methods and considerations for keeping multiple applications in sync, focusing on real-time and event-driven synchronization strategies.
  - **Sync Strategies Part 1:** One-Way Syncs
  - **Sync Strategies Part 2:** Two-Way Syncs
  - **Sync Strategies Part 3:** Real-Time Syncs
  - **Sync Strategies Part 4:** Syncing Multiple Applications

  See: COMMUNITY.BOOMI.COM

These articles offer valuable insights into implementing effective messaging patterns within Boomi integrations.

# Questions

# THANK YOU!!

boomi