

## RELEASE NOTES

# Working Together Has Never Been Easier

AMI now facilitates concurrent development on the same layout with a powerful merge tool, and with the new REST API Server, AMI data and monitoring information are now accessible.

## Accelerate projects with concurrent layout development

We all know how painful it is when multiple people are working on the same project or document, and changes need to be merged. Traditional merge tools tend to be simple text based editors with no intelligence around context. At 3forge, we felt there needed to be a better way, so we created the world's first context aware graphical tree based merge solution. Now, when multiple users are editing the same dashboard, they are prompted with an intuitive tool that understands the code, and can automatically merge changes. Merge conflicts are also easily resolved using a choice-based approach. The hierarchical approach means even the largest layouts let the merger focus on the areas needing attention.

## Connect microservices with gRPC

Google Remote Procedure Call (gRPC) has become a reliable option for companies looking for a faster and more efficient alternative to JSON, bi-directional streaming, strongly-typed APIs, and other useful options for microservices and internal connectivity. AMI now offers multiple options for streaming/querying information from a gRPC server, including a feed handler for real-time message subscriptions, a datasource adapter for query-based retrieval, and a realtime Processor for web-based subscriptions which can be configured during runtime.

## Interact with AMIDB using a REST API

AMI's historical and real-time databases are highly scalable, the virtualization of data streams that AMI provide is very effective at shielding end-users from the complexity of various sources, and entitlement management is deeply granular, so many users look to standardize their data access layer on 3forge. The missing part was to get the ability to query the data from other systems using a simple programmatic interface rather than JDBC. In this release, 3forge is addressing this need with a REST API server that opens a query endpoint for controlled, scalable, and reliable data requests with virtually no technical constraints.

## Monitor extensive deployments via a REST API

Enterprise clients often have hundreds of instances of AMI running across many desks and locations on mission-critical workflows, so monitoring the health of such large deployments requires a more systematic approach, including through the use of external systems.

AMI now provides its own REST server API to deliver monitoring metrics and retrieve statistics about the AMI session and the virtual machine it is running in via various endpoints. AMI's own load balancers are now using these metrics to route traffic as well.

## Add real-time market data from QuantHouse

Introducing market data directly into dashboards and layout can help operators reach faster decisions. In this release, AMI introduces a new market data feed for QuantHouse market data with flexible configuration. With this new feed handler, users can subscribe to the instruments of their choice and start building real-time interactive dashboard with Level 1 (best bid and offer) and Level 2 (full market depth) data.

## Additional features included in this release

- 23164** The JDBC adapters now supports temporal data types. Developers can now directly import time information, without additional formatting in AMI, straight from the JDBC adapter.
- 23198** The method `getBrowserIp()` is now available in the data filter plugin allowing developers to get browser information from within the data filter itself.
- 23193** Addition of three new methods to the `AmiScript` class `FormButtonField`:
  - `getAutoDisable()`
  - `isAutoDisabled()`
  - `resetAutoDisabled()`These methods introduce the ability to configure buttons to be disabled on-click by a user at the developer's discretion. This introduces tighter control over user-thrashing and reducing redundancy for idempotent behavior.
- 23264** Added the optional parameter to specify which relay the session method `sendEmail()` and `sendEmailSync()` should use. This grants greater granular control to developers using multiple relays.

**23263** Introduced further support for casting to `List` and `Set` from other collection types, including casting JSON formatted data to `Set`. Easier casting and data conversion within AMI gives developers greater control by enabling more choice in data storage options – all with minimal effort from the developer.

**23261** Added the `SHOW CPU` command to admin console. This command displays recent CPU usage for easier debugging and metric analysis.

**23260** The Center command `SHOW VARS` can now be used within the web interface. This command can be used in any window that accepts AmiScript allowing developers to easily see and access different session variables in AMI.

**23232** Python-style syntax has been added for the following:

- `Map {:, :, :}`
- `List [,,]`

For examples utilizing Python syntax, please refer to the documentation for both `Map` and `List`, as well as our [reference documentation](#).

**22779** Added two more Openfin methods:

- `open();`
- `raiseIntentByApp();`

These methods allow developers more options for integrating multiple applications and windows within the Openfin framework. Use the `raiseIntentByApp` method to specify which app handles the intent.

**22871** The WebBalancer now supports the following headers: `X-Foward-For`, `X-Forward-Host`, and `X-Forward-Port`. In combination with `WebBalancer` routing, developers can easily keep track of which IPs and ports are being used.

**22807** Implemented performance upgrades on datamodel tables. Internal changes on how tables are implemented in the backend result in improved blender performance for developers.

**23004** Text fields now have a new callback: `textfield::onAutocomplete`. Developers can now programmatically introduce some logic for autocomplete in text fields beyond the existing callbacks.

**23583** Transient columns are a key feature for enabling pivot table functionality. We have added the Table Panel AmiScript method `TablePanel::removeTransientColumns()` to remove unnecessary columns for improved performance, especially in real-time cases.

**23843** Added the ability to execute scripts directly in an AMI layout via `Layout.exec()`. This method allows for scripts to be deferred and run after some delay on the current layout. This is a highly flexible tool to allow developers to implement deferred logic as required.

**23517** Additional methods for tree panels are now available for developer use in AMI. These are:

- `TreePanelColumn::autosize()`
- `TreePanelColumn::setWidth()`
- `TreePanelColumn::setHeaderStyle()`

These enable greater front-end customizability.

**To access this release, please contact [support@3forge.com](mailto:support@3forge.com)**



## About 3forge

3forge is the leading enterprise software provider specializing in real-time data integration, visualization, and high-performance application development. Since 2011, we have helped global investment banks, hedge funds, and asset managers build bespoke, mission-critical applications — faster and more efficiently than traditional development methods.

Unlike fragmented solutions, 3forge provides a **fully integrated** platform tailored to the needs of financial institutions. Whether managing risk, optimizing trade execution, or processing vast amounts of real-time data, our platform delivers **unparalleled speed, reliability, and security** —helping you **turn complexity into a competitive advantage**.

For more information, contact [info@3forge.com](mailto:info@3forge.com)

## Legal Statement

This document, including all text, graphics, and any related content, is proprietary and confidential information of 3forge, LLC. All rights are reserved. Unauthorized reproduction, distribution, transmission, or publication of this material, in whole or in part, without the express written consent of 3forge, LLC is strictly prohibited.