Streamlining Development:
3forge's Latest Release

In Q4 2023, 3forge enhanced its real-time data management and visualization suite with major updates, designed to enhance efficiency, customization, and user experience. Each of these highlighted features represents our ongoing commitment to innovation and excellence.

Smart Debugger

Our web-based IDE just got much better. The 3forge in dashboard debugger now includes 'step over' and 'step to cursor'. These improvements offer developers more granular control over the debugging process, allowing for more efficient troubleshooting and optimization of scripts. By streamlining the debugging workflow, developers can identify and resolve issues more quickly, enhancing the overall development experience.

Variable Based Styling

A notable advancement in customization and styling, 3forge Web now supports variables within its styling framework. This feature allows for stylesets to contain business domain-based styling parameters that can be used throughout the dashboard, avoiding the hardcoding of colors, etc when including conditional formatting.

Enhanced PDF Methods

The Enhanced PDF functionality introduced in this release significantly expands PDF reporting catering to a wide range of reporting needs. Capabilities for custom fonts, different languages and emoticons broaden the scope for creating more personalized and engaging documents. Furthermore, the ability to create columns within PDFs enables users to design reports with multi-column text layouts, enhancing readability and presentation. These enhancements empower professional-looking PDF documents directly from the AMI platform, accommodating a diverse array of reporting requirements with greater flexibility and creativity.

AMI Web

- Added last executed runtime to tree header
- Add style variable sorting
- In style editor, clicking + when no colors gives message to user
- Make clickable and oneclick set to off by default for transient columns
- Support breakpoints in where/for conditional statements
- Support custom fonts with specialized styles
- Transparent color supported in all color style fields
- Added step over and step to cursor in AmiScript dm debugger and step over in custom method portlet
- Match AMIDB Shell Tool and Datamodel limit behaviors
- Add tooltip support for tables
- Datamodel::onComplete() gets called after
- downstream visualizations are updated
- Don't clear out realtime datamodels when adding static visual
- Create realtime panel is no longer modal
- Optimization: only invoke onScrollbar callback if implemented
- Allow adding datamodels directly on realtime visualizations, don't include non-broadcast tables in realtime feeds
- Support entryset()
- Refresh dm stats panel when dm changes name
- Limit processor schema should be output schema of lower processor
- In View Data panel, added data type to column sub-header
- Params column is deprecated
- Support saving tree sort
- Ensure CSS transition is removed when cell finishes flashing
- Support saving tree filter
- AmiWeb style now supports variables
- Added custom css to styles
- Added PdfBuilder::createColumn(). This allows users to create columns within pdfs so that they can create columns of text that spans in 2,3, or more columns.
- ColumnarRow support clear
- Add logging=off/on to ds queries
- Multicheckbox: color dodge on X based on field background color; reduced opacity for better visibility

AMI Center/AMIDB/AMIHDB

- Optimized HDB query for group by partition column, order by partition column limit
- Added property ami.hdb.blocksize=8192 . This should match the block size of the disk that hdb is read/writing
- to
- Optimization if row count is 0 there are no values to read
- HDB optimizations faster group by indexed column
- Optimized copying data from db result set to columnar tables
- In HDB don't allow duplicate index on column
- Optimize HDB scenario with where != on partition column
- HDB optimization: avoid duplicate sort
- Support entryset,values,keyset for datafeed objects
- Support putAll and put on row (in amiscript trigger)
- In HDB, make sure directory doesn't already exist when starting new partition
- Disable link to panel menu option if there is nothing to link
- AmiScript trigger support old_row and new_row
- Remove the spin check which causes high CPU usuage when many runnables when completed at once
- Added allowExternalUpdates option on PROJECTION trigger

AMI Relay

- Better error code on admin port conflict

AMI Script

- Add two methods to Session:callCommand(appId, commandId, connectionId, table) and Session::callCommandById(commandUniqueAmiId)
- Update Password::toString() to just return ********
- Add AmiScriptClass Password, new Password(pass), Password::getRawPassword(), Password::clear()

- Added TablePanel::removeColumn(). This removes transient columns only.
- Add StringBuilder to AmiScript classes
- Support rounding large numbers
- Make AmiScript string::split() same as strSplit()
- Support negative numbers in startsWith offset
- Added TablePanelColumn::setClickable(...) AmiScript method
- Added TablePanelColumn::setEditable(), TablePanelColumn::getEditType(), TablePanelColumn::isEditable(), TablePanelColumn::setHeaderStyle(), TablePanelColumn::setTitle(). Added support for styling transient columns.
- Added PdfBuilder::addColumn(),
- PdfBuilder::endColumn() AmiScript methods
- Add callback onBeforeEdit() on TablePanel.
- Added PdfBuilder::addCustomFont() method. Can now add different languages/emoticons to pdfs.
- Add Map::putAll()
- Add Map::containsKey() and Map::containsValue()
- Optimized AmiScript trigger. Also support upserting with partial column participation
- Added AmiScript::timestampNano()

Adapters

- Allow property to disable listener (stop listening for updates) for Hazelcast
- Add debug property for Hazelcast
- Added ResultSetGetter::set(...) for prepared statements. Need for Oracle Adapter UPLOAD to set type for UNIONing null values
- Make schema registry optional for Kafka
- Ssh adapter return tables have better names: stderr, stdout, exitcode
- Important change to OAuth Plugin, add refresh token properties to make scope and redirect uri not required added properties oauth.refresh.grant.type, oauth.refresh.scope, oauth.refresh.redirect.uri, oauth.refresh.client.id, oauth.refresh.client.secret. May need to set the scope and redirect uri if upgrading from previous version, consult with your oauth provider to see if it's required.

Look and Feel

- New style options, rename default style to classic, change content security policy
- Better arrangement in field styler
- Menu Item Dashboard Styles renamed to Dashboard Style
- Updated login page color scheme
- Reduced whitespace on dynamic dm option on edit datamodel window

Misc

- Optimized GUID generation
- Optimize logging of AmiOne.amilog Partition Stats
- Added Powered by 3forge signature on all emails that are sent from 3forge.
- Optimization: only flush messages once queue is empty
- Support variable colors in chart formulas
- Add more security/authentication options

If you have any questions about these new features or would like a demo, please email info@3forge.com.