November 2023 VERSION 4.2.2

Active>360

Active>360 Release Notes

AB INITIO SOFTWARE LLC 201 Spring St. Lexington MA 02421 Voice +1 781.301.2000 support@abinitio.com

Intellectual Property Rights & Warranty Disclaimer

COPYRIGHTS

CONFIDENTIAL & PROPRIETARY

PATENT NOTICE

Patent https://www.abinitio.com/en/legal/ip-guidelines.html

RESTRICTED RIGHTS LEGEND

WARRANTY DISCLAIMER

TRADEMARKS

The following are worldwide trademarks or service marks of or licensed to Ab Initio (those marked ® are registered in the US Trademark Office, and may be registered in other countries):

- >®
- > (stylized)®
- Ab Initio®
- Ab Initio (stylized)®
- Abinitio.com®
- Ab Initio I>0®
- Acquire>It®
- Active>360™
- Active>Data®
- BRE®
- Business Rules Environment®
- Conduct>It®
- Continuous>Flows®
- Continuous Flows®
- Control>Center®
- Co>Operating®
- Co>Operation®
- Cooperating Enterprise®
- Co>Operating System®
- Co>Operative®
- Co>Ordinate®
- Co>Relate®
- Data>Catalog™
- Data>Profiler®
- Data Quality Assessment™
- Data Quality Environment™
- DQA™
- Dynamic Data Mart®
- Easy>Data®
- EME®
- Enterprise Meta>Environment®
- Express>It®
- From First Principles™
- From the Beginning®
- GDE®
- Graphical Development Environment™
- I>0®
- INIT®
- Making Things Simple Is Hard Work®
- Metadata Hub®
- Metadata Portal®
- Query>It®
- Re>Posit®
- Re>Source®
- Semantic Discovery™
- Shop For Data®
- Spec-To-Graph™
- Test Data Management™
- The Company Operating System®

Certain product, service, or company designations for companies other than Ab Initio are mentioned in this documentation for identification purposes only. Such designations are often claimed as trademarks or service marks. In instances where Ab Initio is aware of a claim, the designation appears in initial capital or all capital letters. However, readers should contact the appropriate companies for more complete information regarding such designations and their registration status.

Contents

About this book

These release notes describe the most recent version of Ab Initio Active>360 software. They include information related to installation, upgrades, new features, and other changes.

This document describes only new features and enhancements in the most recent version of Active>360. For a list of fixed issues and security updates since the previous version of Active>360, see the *Active>360 Patch Notes*.

Audience

These release notes are for system administrators, developers, business users, and others who are responsible for installing, upgrading,, configuring, or customizing Active>360 software.

Getting assistance

General information — On the Ab Initio Forum (https://forum.abinitio.com), you can find online and PDF product documentation, training courses, informative videos, and user discussions and reusable solutions.

To send feedback about documentation, navigate to the pertinent web page in the Ab Initio Help Library and click the topic's Send Feedback button  . You can also send email to **documentation@abinitio.com**.

To send feedback about videos, send email to **videonotes@abinitio.com**.

Assistance from Ab Initio Support — To contact Ab Initio Support, send email to **support@abinitio.com** or call +1 781-301-2100.

When reporting a problem, include the following:

- The Co>Operating System® version and, if applicable, other Ab Initio product software versions
- The platform (operating system and version) that your Co>Operating System® software is running on
- The complete error message (if any)
- A description of what you were doing when the error message (if any) appeared
- For database issues:
  - The type and version of the database (for example, Oracle 12c version 12.1.0.1.0)
  - The platform the database server is on
  - The JDBC driver version (if applicable)

1

What's new in Active>360 4.2.2

This chapter describes highlights of this version, and provides a complete list of new features and changes. It contains the following sections:

Highlights of this version

- **Major new features**
  - **Support for fast-moving reference data** — Active>360 now supports an optional Active>Data cache to enable real-time handling of changes to reference data that occur between batch rebuilds of the reference data lookup file. For more information, see "Support for fast-moving reference data".
  - **Support for calculating cycle date and time zone with override expressions** — Active>360 now supports the calculation of cycle-date and time-zone attributes with ruleset override expressions. For more information, see "Support for calculating cycle date and time zone with override expressions".
  - **Improved support for cross-entity links** — Links (entity keys) that are used in cross-entity aggregates can now be used in the Active>Portal expression editor.
  - **More variations for scaled binary decimals** — Active>360 now supports a wider range of variations of scaled binary decimals in the Active>Portal. For more information, see "More variations for scaled binary decimals".
  - **Reference data support in technical triggers** — Active>360 now supports reference data lookups in technical triggers.
  - **New simulation preview mode (experimental)** — Active>360 now includes a basic simulation preview mode hidden under an Easter egg. For more information, see "New simulation preview mode (experimental)".
  - **Improved support for inventories (experimental)** — This release includes improvements to the experimental inventories feature that was introduced in Active>360 Version 4.2.1. For more information, see "Improved support for inventories (experimental)".
- **Important upgrade and compatibility changes**
  - **Kubernetes support** — Active>360 can now be installed in Kubernetes, including in the more restrictive RedHat OpenShift environment. For more information, see "Kubernetes support".

Complete list of new features and changes

This version of Active>360 contains the following new features and changes:

- Support for fast-moving reference data
- Support for calculating cycle date and time zone with override expressions
- Improved support for cross-entity links
- More variations for scaled binary decimals
- Reference data support in technical triggers
- New simulation preview mode (experimental)
- Improved support for inventories (experimental)
- Kubernetes support

Support for fast-moving reference data

Active>360 now supports an optional Active>Data cache to enable real-time handling of changes to reference data that occur between batch rebuilds of the reference data lookup file.

A reference data file is a lookup file that is periodically rebuilt in a batch job. You can now make quick updates to a reference data file between scheduled rebuild jobs — for example, adding or replacing a record — by using a REST API call to write an *update record*.

Update records are cached in Active>Data memory and are used preferentially when the lookup is accessed. The update records are subsequently written to the lookup file and removed from the Active>Data cache when the lookup file is rebuilt in the next batch job.

Depending on the environment, the memory cost can be significant when using Active>Data caching because the cache is replicated to every data node in the Active>Data cluster.

Support for calculating cycle date and time zone with override expressions

Active>360 now supports the calculation of cycle-date and time-zone attributes with ruleset override expressions.

Note the following terminology:

- *Cycle date* is an account-specific or customer-specific day of the month when monthly aggregates roll over.
- *Time zone* is an account-specific or customer-specific day-divide that determines when daily aggregates roll over — for example, midnight in the time zone where a particular account is based.

For example, previously, a mobile phone provider had to populate cycle-date and time-zone attributes in an entity-specific ingested attribute file. Now, the provider can set the cycle date and time zone for a device using a lookup file that is keyed by an account rather than by the device.

Improved support for cross-entity links

Links (entity keys) that are used in cross-entity aggregates can now be used in the Active>Portal expression editor.

More variations for scaled binary decimals

Active>360 now supports a wider range of variations — up to 9 decimal places — of scaled binary decimals in the Active>Portal. This improvement is critical for some financial applications; for example, those that work with cryptocurrencies.

The specific number of variations of scaled binary decimals that are visible in the Active>Portal is site-specific, and is configured by the Active>360 **act.portal.scaled_binary_type_decimal_places** installation configuration parameter.

Reference data support in technical triggers

Active>360 now supports reference data lookups in technical triggers.

New simulation preview mode (experimental)

Active>360 now includes a basic simulation preview mode hidden under an Easter egg. This new simulation mode is still under development, but currently enables you to do the following in the simulation environment only (not in production):

- Make various changes to entities through the Active>Portal; for example, create watchers, define aggregates, and modify ruleset logic.
- Deploy your changes.
- Run historical events through your new logic to see what happens.

The simulation environment generates Data Profiler results for all aggregates and calculations, and provides a count of notifications that would have been sent in a non-simulation environment.

Improved support for inventories (experimental)

This release includes improvements to the experimental inventories feature that was introduced in Active>360 Version 4.2.1. This feature is still hidden under an Easter Egg, but is now available in the Active>Portal expression editor and the Active>360 Reference Application (RFA).

Kubernetes support

Active>360 can now be installed in Kubernetes, including in the more restrictive RedHat OpenShift environment. This new support is sufficient for development deployments but is not yet recommended for production deployments.

2

New features in earlier versions of Active>360

This chapter describes new features in earlier versions of Active>360. It contains the following sections:

Active>360 Version 4.2.1

This section describes highlights of this version and provides a complete list of new features and changes. It contains the following sections:

Highlights of this version

- **Major new features**
  - **Technical triggers** — Technical triggers are similar to watchers in that they look for interesting events or changes in entity profile data and send out corresponding notifications. However, technical triggers are more extensible and flexible than watchers, and they are intended for more technical users. For more information, see "Technical triggers".
  - **Reference data files** — You can now define reference data files for use in rule and condition expressions and in technical triggers in the Active>Portal web application. For more information, see "Reference data files".
  - **Computable notification fields** — You can now add fields to a notification (similar to event definitions). When you create a watcher that sends a given notification, you can define a ruleset that contains logic to calculate values for every field in that notification. For more information, see "Computable notification fields".
  - **Co>Operating System version upgrade** — For more information, see "Co>Operating System version upgrade".
  - **New Variations windowed aggregate type for date ranges** — A new windowed aggregate type, **Variations**, lets you obtain aggregate values in multiple sliding windows with potentially overlapping date ranges. For more information, see "New Variations windowed aggregate type for date ranges".
  - **New windowed aggregate subtypes for consecutive days** — Three new windowed aggregate subtypes, **Average Consecutive Days**, **Minimum Consecutive Days**, and **Maximum Consecutive Days**, lets you capture the average, minimum, or maximum numbers of consecutive days in which a particular defined event occurs for a given windowed aggregate. For more information, see "New windowed aggregate subtypes for consecutive days".
  - **Improved lineage and dependency calculations** — To support the new object types and subtypes in this version of Active>360, and to improve overall efficiency and reliability, improvements have been made to the underlying code for performing lineage and dependency calculations. For more information, see "Improved lineage and dependency calculations".
  - **Active>Portal usability enhancements** — To improve overall usability, several enhancements have been made to the Active>Portal web application. For more information, see "Active>Portal usability enhancements".
  - **New Data>Catalog integration parameters** — To improve Data>Catalog integration capabilities, a number of installation configuration parameters have been added to the Active>360 installer. For more information, see "New Data>Catalog integration parameters".
  - **Improved support for the scaled binary decimal data type** — For more information, see "Improved support for scaled binary decimal data type".

- **Improved CIP integration support** — CIP integration support has been improved in several areas. For more information, see "Improved CIP integration support".
- **Vectors in entities (experimental)** — A hidden experimental feature now lets you define an *inventory* of subrecords inside of an entity. In this context, an inventory is a vector of subrecords, and each subrecord can contain a number of fields. For more information, see "Vectors in entities (experimental)".

- **Important upgrade and compatibility changes**
  - **Grafana removed from observability stack** — The Grafana product is no longer included in the Active>360 observability stack. You can install Grafana separately and connect it to the Prometheus implementation that is still included with Active>360. As part of this change, the Active>Portal no longer includes any Grafana-based metrics dashboards.

Complete list of new features and changes

This version of Active>360 contains the following new features and changes:

Technical triggers

Technical triggers are similar to watchers in that they look for interesting events or changes in entity profile data and send out corresponding notifications. However, technical triggers are more extensible and flexible than watchers, and they are intended for more technical users.

Technical triggers let you do the following:

- Define a DML record that will be saved in the shared variable collection and can be both read and written by the technical trigger. This lets you implement complex logic when it is necessary to remember state between invocations of the technical trigger.
- Define a DML transform function that determines whether the technical trigger should be executed (returns a value of **true** or **false**). Typically such a transform would be defined for events, but it can also be defined for any access. This is similar to the expression associated with a watcher, but it is implemented in DML instead of in the Active>Portal expression editor.

- Define a DML transform function that is executed if the trigger expression returns a value of **true**. This lets you do the following inside a function:
  - Modify the technical trigger state.
  - Invoke one or more notifications, including calculating the values for the fields in those notifications.
  - Change the value of any calculation in the entity profile.

Technical trigger logic has access to the current state, the event (if any), the current state of the entity profile and any reference data files.

Active>360 includes a batch graph that you can use to develop and test the logic for technical triggers from the GDE before copying the logic into the Active>Portal.

Reference data files

Reference data files are lookup files that are made available inside Active>360 for use in rule and condition expressions and in technical triggers in the Active>Portal web application.

Similar to defining events and notifications, you define reference data files by creating the file metadata and then defining the structure of folders and fields in the Active>Portal. Reference data files must have a key (multipart keys are also supported), which lets you access the reference data file using a key that is not just the entity ID. You can also access the reference data by using ingested attributes.

When you define a reference data file, you indicate which entities it can be used with, and what the default value should be for each key. The reference data file will be displayed in the Active>Portal expression editor when using the corresponding entity as a lookup file. You can reference the data by using the default key values or by specifying overrides using standard ruleset notation.

Active>360 includes a batch pset for loading reference data files, similar to the pset for loading ingested attributes.

Computable notification fields

You can now add fields to a notification (similar to event definitions). When you create a watcher that sends a given notification, you can define a ruleset that contains logic to calculate values for every field in that notification. The ruleset has access to the entity profile and to the event that triggered the watcher.

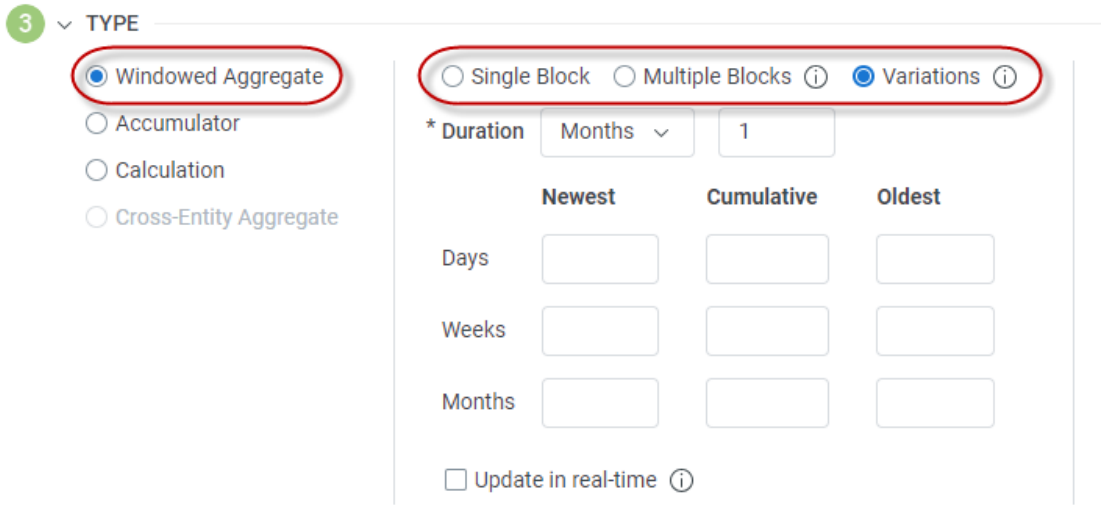
Co>Operating System version upgrade

The Co>Operating System package that is bundled with Active>360 has been upgraded to Version 4.2.1.3.

New Variations windowed aggregate type for date ranges

A new windowed aggregate type, **Variations**, lets you obtain aggregate values in multiple sliding windows with potentially overlapping date ranges. For example, you could have a 90-day (3-month) windowed aggregate for which you want to obtain values for today, yesterday, the day before that, the previous week, the previous two weeks combined, the previous three weeks combined, and so forth.

In the Active>Portal, the new **Variations** windowed aggregate type is available in the Type pane on Windowed Aggregate pages. For example, if you create a new windowed aggregate, the choices in the Type pane are now **Single Block**, **Multiple Blocks**, and **Variations**, as shown below:



Note the following about the **Variations** aggregate type:

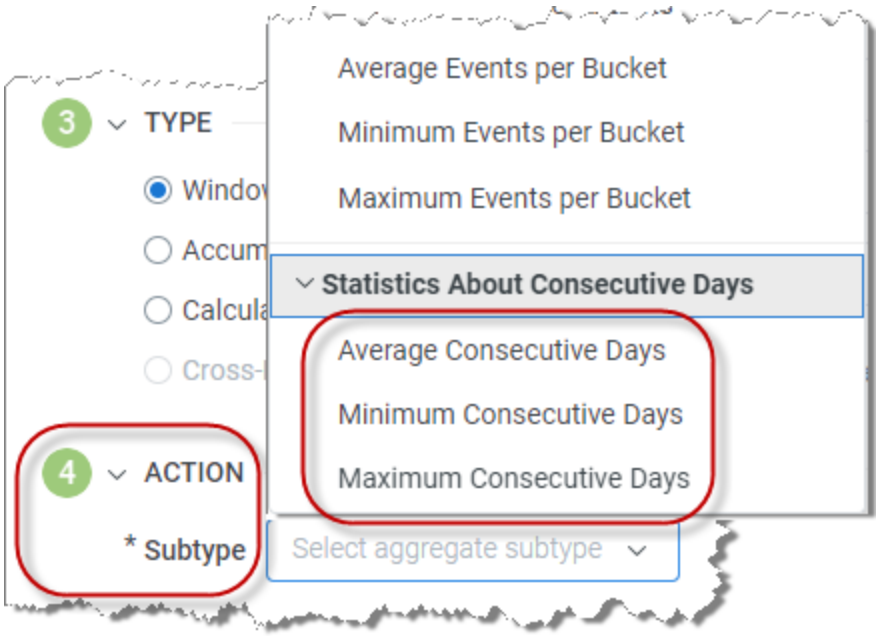
- The **Variations** type is only supported for sliding window aggregates; calendar aggregates are not supported.
- Each variation returns its results as a separate aggregate value.
- All variation values are dynamic, meaning that the variations themselves are not saved in the shared variable collection.
- Variation values are computed in real-time from partial values saved in memory or in the real-time aggregate history file.
- The name of the variation is derived from the name of the underlying aggregate followed by a suffix that depends on the variation definition; for example, **_d1** for the current day, **_w2** for the previous week.
- Individual variations can be separately included or excluded in subsets.

New windowed aggregate subtypes for consecutive days

Three new windowed aggregate subtypes, **Average Consecutive Days**, **Minimum Consecutive Days**, and **Maximum Consecutive Days**, lets you capture the average, minimum, or maximum numbers of consecutive days in which a particular defined event occurs for a given windowed aggregate.

In the Active>Portal, the new windowed aggregate subtypes are available on Windowed Aggregate pages in the Action pane, under the **Statistics About Consecutive Days** heading in the **Subtype** drop-down list. For example, if you create a new windowed aggregate, the choices in the **Subtype** drop-down list in the

Action pane now include **Average Consecutive Days**, **Minimum Consecutive Days**, and **Maximum Consecutive Days**, as shown below:



Improved lineage and dependency calculations

To support the new object types and subtypes in this version of Active>360, and to improve overall efficiency and reliability, improvements have been made to the underlying code for performing lineage and dependency calculations.

These improvements are reflected in the following areas:

- Support for the new object types and subtypes introduced in this product release
- Calculations of usage and lineage information
- Preventing objects that are in use from being deprecated
- Ensuring all dependent objects are included in a deployment

Active>Portal usability enhancements

To improve overall usability, the following enhancements have been made to the Active>Portal web application:

- Lineage is now available in all Details views.
- Lineage views are now more coarse-grained to make them easier to understand.
- Child fields for reference data and inventory attributes are now hidden by default. You can still access field-level lineage by clicking on the Details view for the individual field.
- In Details views, the name of the Usage section has been changed to References to better reflect the information shown in the section.
- A **Technical Triggers** link has been added to the Outputs pane on the Active>Portal home page.

New Data>Catalog integration parameters

To improve Data>Catalog integration capabilities, the following installation configuration parameters have been added to the Active>360 installer:

Parameter	Description
data_catalog.export_dir	The full directory path on the primary Active>360 runtime host to which the Active>360 batch subsystem can export Data>Catalog files.
data_catalog.import_dir	The full directory path on the primary Active>360 runtime host from which the Active>360 batch subsystem can import Data>Catalog files.
data_catalog.integrated	A Boolean value that indicates whether this Active>360 installation is integrated with an existing Data Catalog Services deployment.
data_catalog.queryit.cart_composite_id	During Active>360 deployment of daily Data>Catalog unload jobs, the composite ID of the data cart to which new catalog datasets are added.
data_catalog.queryit.dataspace	The name of the data space that the Active>360 installer configures for the Query>It service instance.
data_catalog.queryit.host_data_dir	The data directory on the Query>It host for the Query>It service instance used by Active>360.
data_catalog.queryit.instance	The name of the Query>It service instance created by the Active>360 installer for the Data Catalog Services deployment used by Active>360.
data_catalog.queryit.project	The name of the dataset common project created by the Active>360 installer for the Query>It service instance.
data_catalog.queryit.schema	The name of the data source schema for the Query>It service instance.
data_catalog.queryit.schema_kind	The schema kind for the Query>It schema specified by the **data_catalog.queryit.schema** parameter.
data_catalog.url	The URL for the Data Catalog Services deployment, if any, used with Active>360.
data_catalog.username	The username used by Active>360 to join the Data Catalog Services deployment.

For more information, see the *Active>360 Installation Guide*.

Improved support for scaled binary decimal data type

The following improvements have been made to Active>360 support for the **scaled binary decimal** data type:

- The supported precision has been increased to 9 places; earlier releases supported only 5 places.
- The Active>360 Configurator now supports a comma-separated integer list of precision values; earlier releases supported only a single integer value.
- The Active>Portal web application now supports multichoice lists of precision values, as appropriate for a given calculation or aggregate.

Improved CIP integration support

The following improvements have been made to CIP integration support:

- It is now possible to map **decimal** data type fields in imported entity data to different data types.
- Subsets extracted during import of CIP inbound and outbound profile data now include subset usage attributes (queries, enriched events, notifications).

Vectors in entities (experimental)

A hidden experimental feature now lets you define an *inventory* of subrecords inside of an entity. In this context, an inventory is a vector of subrecords, and each subrecord can contain a number of fields.

NOTE: For instructions on enabling this hidden feature, contact Ab Initio Support.

When this feature is enabled, you can work with inventories in either of the following ways:

- Define an inventory by creating a folder list in an event with the same structure as the inventory you want to define, then specifying an event condition that will cause that folder list to be copied into the inventory (using replace, not append).
- Arbitrarily modify an inventory by using a DML function inside of a technical trigger.

Inventories are accessible inside the Active>Portal expression editor as folder lists, and can be queried using standard list expression functions. For example, you can use inventories to compute other calculations, or in watchers and technical triggers. Inventories are also visible in **get_entity** calls.

Grafana removed from observability stack

The Grafana product is no longer included in the Active>360 observability stack. You can install Grafana separately and connect it to the Prometheus implementation that is still included with Active>360. As part of this change, the Active>Portal no longer includes any Grafana-based metrics dashboards.

3

List of new Active>360 features and changes by version

This chapter lists, by version, the features and changes introduced in previous releases (beginning with Version 4.2.1). It includes the following topics:

Installation and configuration

Version	Feature
4.2.2	Kubernetes support
4.2.1	Improved CIP integration support

Active > Portal

Version	Feature
4.2.2	Support for calculating cycle date and time zone with override expressions
4.2.2	Improved support for cross-entity links
4.2.2	More variations for scaled binary decimals
4.2.1	Active>Portal usability enhancements
4.2.1	Improved support for scaled binary decimal data type
4.2.1	New Variations windowed aggregate type for date ranges
4.2.1	New windowed aggregate subtypes for consecutive days

Active > Data

Version	Feature
4.2.2	Support for fast-moving reference data
4.2.2	Support for calculating cycle date and time zone with override expressions
4.2.1	Reference data files

Triggers, watchers, and notifications

Version	Feature
4.2.2	Reference data support in technical triggers
4.2.1	Technical triggers
4.2.1	Computable notification fields

Aggregates, calculations, and lineage

Version	Feature
4.2.1	Improved lineage and dependency calculations

Ingestion, Data > Catalog, and data types

Version	Feature
4.2.1	New Data>Catalog integration parameters

General enhancements

Version	Feature
4.2.2	New simulation preview mode (experimental)
4.2.2	Improved support for inventories (experimental)
4.2.1	Co>Operating System version upgrade
4.2.1	Grafana removed from observability stack
4.2.1	Vectors in entities (experimental)