MORNINGSTAR Wealth

Introduction to ByAllAccounts Connect Account Linking

(Formerly known as Custom Element for Account Setup, Version 2)

MCondoc-3.12-20250627

ByAllAccounts

Account Linking version: 3.12

Document Version: MCondoc-3.12-20250627

Guide generated on: 6/26/2025

Please report incorrect or missing information.

Telephone: 1-866-856-4951 Option 1

Byallaccounts-support@morningstar.com

Web: http://byallaccounts.morningstar.com

Table of Contents

About this document

This document provides an overview of the ByAllAccounts Connect Account Linking component which provides access to the ByAllAccounts^{SM} (BAA) aggregation service.

This document describes the Account Linking component, steps for embedding it in a web application, methods of authentication, and ways to customize it.

Use this document if you are using ByAllAccounts DataConnect API for single sign on (SSO). If you are using Morningstar Authentication for single sign on (SSO), please refer to Introduction to ByAllAccounts Connect Component, with Morningstar Authentication Instructions.

If you are implementing Account Linking with Account Summary

If you are implementing Account Linking with the Account Summary component, you need a different guide. Please refer to one of the following guides, depending on the authentication method you are using:

- ByAllAccounts Connect Suite - Account Summary with Account Linking if you are using ByAllAccounts DataConnect API for single sign on (SSO).
- ByAllAccounts Connect Suite - Account Summary with Account Linking using Morningstar Authentication if you are using Morningstar Authentication for single sign on (SSO).

Changes in this version

This document is evolving along with the components. The changes in this release are described in Appendix A: Cumulative changes along with important information about other recent changes.

About Account Linking

The Account Linking component provides access to the ByAllAccounts^{SM} aggregation service. It is a W3C custom HTML element that implements the account setup functionality for aggregation within another application.

Account Linking supports adding and maintaining accounts within the aggregation system. The aggregated account data, including balance, holding, and transaction information, are available via ByAllAccounts APIs.

Account Linking can be fully customized to integrate into your parent page, with customizations to terminology, styles, and features.

To fully integrate Account Linking you will need to work with a ByAllAccounts Implementation Manager who will help with the internal ByAllAccounts setups required including creating a Firm for you in our system and providing an administrator login and password to perform API calls, etc.

See it in action

To see Account Linking in action, click here.

Known limitations

There are currently no known limitations. The capabilities, interfaces, and behaviors of the ByAllAccounts components are subject to change as we make refinements based on customer feedback.

User types

The components can only be used with BAA Investor users and those users must:

- have full Read-Write permission to their own data
- be Single Sign On (SSO) users
- be in a BAA Firm that is licensed to use Account Linking

Browser support

Up-to-date versions of thses browsers are fully supported:

- Chrome (including incognito mode), Firefox, Microsoft Edge, and Safari

Authentication

Authentication is handled by the parent page before it invokes Account Linking.

Authentication consists of invoking a DataConnect API method (<SESSIONAUTHRQ> or <SAMLAUTHRQ>) to authenticate the investor and obtain values for *jsessionId* and *csrfToken*. The *jsessiolid* and *csrfToken* values must be passed to the component using the *auth-context* attribute.

For details you can refer to the Single Sign-on Operations section in the DataConnect V4 Ultra User Guide.

Note: If the parent page passes in read-only authorization for the user through *auth-context*, the Account Linking component cannot be invoked.

Permissions to allow framing

The Account Linking component uses a Content Security Policy (CSP) that controls which domains are allowed to frame it. In order to allow a domain to frame the component, the parent page's domain(s) must be configured in the component's whitelist within the ByAllAccounts service. For details see "Whitelisting requirements" on page 2. Your ByAllAccounts Implementation Manager will assist you with this configuration.

Whitelisting requirements

For your domains to be whitelisted, they must meet the following requirements for ByAllAccounts Connect whitelist entries:

- Must be all lower case
- Must not have a trailing slash
- Can include a port number

- Cannot be “localhost”
- Must be https (not http)

Additionally

- Asterisk (*) wildcards are supported

Whitelisting and your development environment

In production, the ByAllAccounts components absolutely require proper whitelisting. However you may perform development on a system with a domain name that does not meet the whitelisting requirements (e.g. localhost). To work within these constraints, you can:

- Use a mechanism to provide a domain name that meets the requirements, such as a proxy server or a third-party platform.
- Consider available browser options to relax the browser security checking. This modification would be appropriate only for a development environment.

Supported workflows

Account Linking provides the following workflows, initiated by setting the route attribute as shown:

- adding new accounts to the system using *route* with */add-account*
- editing credentials that have already been entered using *route* with */edit-credential*
- adding more accounts to an existing credential using *route* with */discover-accounts*

If you do not provide a value for the *route* attribute then the default behavior is */add-account*. The *route* attribute is shown in the code example in "Example setting all attributes" on page 6 and explained in the table in "About attributes" on page 7.

Download and install the Account Linking component

The *mstar-aggregation-consumer-accountsetup* component is built with Angular and can be used in frameworks such as React, and Vue, or in vanilla JS applications.

These instructions are for acquiring, customizing, and incorporating Account Linking. To fully implement it, work with your ByAllAccounts Implementation Manager.

Step 1

Download the packaged file for Account Linking from https://www.byallaccounts.net/WebPortfolio/mstar-aggregation-consumer-accountsetup.tgz.

Step 2

Install the package by running the following command in your development environment:

```
npm install mstar-aggregation-consumer-accountsetup.tgz
```

Step 3

Confirm the *mstar-aggregation-consumer-accountsetup* folder is under the *node_modules* directory. The folder contains:

- *mstar-aggregation-consumer-accountsetup.js*, which is the JavaScript bundle for the Account Linking component.
- *assets* folder, which contains various configuration files and images for the component.
- *package.json* file that describes the npm package.

The build system should ensure that the *assets* folder and the main JavaScript file (*mstar-aggregation-consumer-accountsetup.js*) of the component are placed under the application's distribution (*/dist*) folder from where it is being served.

Step 4

Customize the user interface. See the "Customize the user interface" on page 4.

Step 5

Import the necessary JavaScript files into your HTML parent page and create an instance of the *mstar-aggregation-consumer-accountsetup* tag element.

Step 6

Incorporate Account Linking into your parent page. See "Incorporate the component into your parent page" on page 5.

Customize the user interface

There are multiple types of customizations you can optionally provide for the component to seamlessly blend its appearance into your parent page. Apply these customizations using the files provided in the *assets* folder.

- **Terminology** – the i18n/en.json translation file contains all the static text displayed on pages, dialogs, and buttons within the interface. Edit this file to customize text and terminology. Configure the component to use your custom terminology file by setting the *translate-file-path* attribute.
- **Styles** – define an optional CSS file to override the default Morningstar Design System (MDS) styles and adjust fonts, colors, and backgrounds for elements in the display. Configure the component to use your custom CSS file by setting the *override-css-file* attribute. To use fonts other than the default MDS font, you can add a global entry like this to your CSS file:

  ```
  * {
    font-family: YourFont, Verdana, sans-serif !important;
    }
  ```

- **Fonts** – define an optional fonts file to override the default Morningstar Design System (MDS) fonts for elements in the display. Configure the component to use your custom fonts by setting the *custom-fonts* attribute. These new fonts are then available to be set using the CSS file specified using the *override-css-file* attribute.

- **Optional features** – the config/ui-config.json file contains options that enable you to turn on/off UI features in Account Linking. Configure Account Linking to user your custom configuration file by setting the *ui-config-file* attribute. Only a subset of the options currently applies to the component. These options are:
  - "fiLogosVisible" - controls whether a page of popular financial institution (FI) logos is shown in the FI selection step. If true, the logos are shown; if false, then only a search control and FI list are shown.
  - "includeURLinSearch" - controls whether the FI URLs are included in the search criteria when the user is searching for an institution.

  For example:

```
"cui-fi-select": {
   "fiLogosVisible": false,
   "includeURLinSearch": false
}
```

Applying the user interface customizations

Use these instructions to customize terminology, styles, fonts, and optional features.

1. Edit the files provided in the *assets* folder. These files and edits are described in "Customize the user interface" on page 4.
2. Modify your web server to allow our origin (www.byallaccounts.net) to access those files. We suggest the following:
   - Header set Access-Control-Allow-Origin "www.byallaccounts.net"
   - Header set Access-Control-Allow-Headers "Cache-Control,Content-Type,Accept,Referer,User-Agent,Sec-Fetch-Dest"
   - Header set Access-Control-Allow-Methods "GET,OPTIONS"

   If this access is not allowed, then the component will not be able to access the customization files and will have to use its own version of those files.
3. Set the appropriate attribute to identity each custom file. For attribute information see "About attributes" on page 7.

Incorporate the component into your parent page

This section provides examples for incorporating the component in your parent page. This code:

- Loads the component.
- Instantiates the component in the parent page.

- Sets attributes. Minimally sets *auth-context*, which is required.
- Makes the component visible in the parent page.

The section includes:

- Descriptions of the attributes for the component
- A simple code example showing the minimum requirements.
- A more detailed code example showing all the attributes.

Additionally, you must determine how to handle the events described in "About events" on page 11.

Minimum example

This code example shows launching Account Linking with minimum required attributes.

You must define the *auth-context* attribute for authentication, as shown here. For details about attributes, refer to "About attributes" on page 7.

```
<!-- Load the main JavaScript bundle file of the Account Linking component -->
<script src="./mstar-aggregation-consumer-accountsetup.js"></script>
<script>
  mstarWebComp = document.createElement(“mstar-aggregation-consumer-accountsetup”);
  mstarWebComp.setAttribute(“auth-context”, JSON.stringify(
   { jsessionId: “212764322EB9DBEBED880425DE3216EB.s1a”,
     csrfToken: “DBC63BDE361C192B3CEF641B4C551DD8AC27B4A0276E91”
   }));
  content.appendChild(mstarWebComp);
 </script>
```

Example setting all attributes

This section shows a code example for launching Account Linking with all applicable attributes.

This sample integration code:

- Loads the Account Linking component.
- Instantiates the component in the page.
- Sets all common attributes. See "About attributes" on page 7 for detailed information about attributes.
- Makes Account Linking visible in the page.

Attributes are available to:

- customize the user interface

  attributes: *ui-config-file*, *translate-file-pat*h, *override-css-file*, *custom-fonts*

  These attributes typically do not change during the lifetime of the parent page. We recommend setting these attributes first.

- provide authentication

  attribute: *auth-context*

  Authentication can change during the life of the parent page when authentication expires or a new user is authentication. If you reset the *auth-context* attribute, you must then reset the *route* attribute.

- initiate a workflow

  attribute: *route*

  The *route* attribute specifies which workflow in Account Linking to initiate. If *auth-context* is changed for then the *route* attribute should be set again after that change.

In this example, *hostname* should be replaced with your host name for those files.

```
<!-- Load the main JavaScript bundle file of the Account Linking component -->
<script src=“./mstar-aggregation-consumer-accountsetup.js”></script>
<script>
  mstarWebComp = document.createElement(“mstar-aggregation-consumer-accountsetup”);
  mstarWebComp.setAttribute(“ui-config-file”,  “https://hostname/customassets/config/ui-
config.json”);
  mstarWebComp.setAttribute(“translate-file-path”,  “https://hostname/customassets/i18n/”);
  mstarWebComp.setAttribute(“override-css-
file”,  “https://hostname/customassets/css/corporatestyle.css”);
  mstarWebComp.setAttribute(“custom-
fonts”,  “https://fonts.googleapis.com/css?family=Lobster”);
  mstarWebComp.setAttribute(“auth-context”, JSON.stringify(          { jsessionId:
“212764322EB9DBEBED880425DE3216EB.s1a”,            csrfToken:
“DBC63BDE361C192B3CEF641B4C551DD8AC27B4A0276E91”          }));
  mstarWebComp.setAttribute(“route”, “/edit-credential;credentialId=21911”);
  content.appendChild(mstarWebComp);
</script>
```

About attributes

The following table describes the attributes for the Account Linking component. The only required attribute is *auth-context*. The only requirement for order of attributes is that if you use *route* for Account Linking, you must set it after setting *auth-context*. Refer to the table below for details.

Any of these attributes can be set (and reset) at any time but *translate-file-path*, *override-css-file*, and *ui-config-file* typically do not change during the lifetime of a parent page. An example of when *auth-context* and *route* might be reset during the lifetime of a parent would be handling of an error event.
For information about events, see "About events" on page 11.

Attribute	Req'd?	Parameters and Description
auth-context	Yes	**Possible Parameters** Must contain one of: ■ *{ jsessionId: <jsessionIdValue>, csrfToken: <csrfTokenValue> }* ■ *{}*

Attribute	Req'd?	Parameters and Description
		Description The *auth-context* attribute is used for user authentication. It is required and may be reset any number of times during the lifetime of the parent page. For example, it would need to be reset after a timeout. It must contain the parameter pair jsessionId/csrfToken or an empty object. When the user is authenticated via SSO, the jsessionId and csrfToken pair must be included, using the form: *{ jsessionId: <jsessionIdValue>, csrfToken: <csrfTokenValue>}* To prevent further use of the component by the user, use the empty value to clear the authentication data from the component: *{}*
route	No	**Possible Parameters** ■ */add-account* ■ */edit-credential;credentialId=<credentialIdValue>* ■ */discover-accounts;credentialId=<credentialIdValue>* ■ */edit-account;accountId=<accountIdValue>* In each case, <*credentialIdValue*> is the unique internal identifier for the Account Credential. Note: Although existing implementations of Account Linking may still be using the *showProgressStep=false* parameter with any of the possible parameters listed above, **it is being deprecated**. See the description section for more information. The */add-account* parameter can additionally have: ■ *fiId=<FIIdValue>* ■ *fiId=<FiIdValue>;newCredential=true* ■ *allowMultipleAdds=false* ■ *credentialId=<credentialIdValue>* The */edit-credential;credentialId=<credentialIdValue>* parameter can additionally have: ■ *viewFirst=true* The */edit-account;accountId=<accountIdValue>* parameter can additionally have: ■ *viewFirst=true* **Description** If you set the *route* attribute, you must set it after the *auth-context* attribute because user authentication must be set before the component can attempt the operation. If you

Attribute	Req'd?	Parameters and Description
		do not set the *auth-context* before the *route* attribute, a badAuthentication event will occur. The *route* attribute is set with a parameter to specify the workflow to initiate. A route may also have optional parameters to further control the behavior. When you specify multiple parameters for a route, the parameters can be listed in any order. The *route* attribute may be set any number of times during the life of the parent page. When you set the *route* attribute it initiates a new or different workflow in Account Linking. For example, you would set a route to add an account, then another route to edit credentials. When a workflow completes, the component tells the parent page that it is done. If you want to bring it back and use it, set another *route*. Use the following parameters. ■ */add-account* Sets the *route* to adding a new account. When no *route* is set, Account Linking starts with the add accounts workflow by default. This workflow takes the user through the required steps for provisioning new accounts in the aggregation system. ■ */add-account;fiId=<FIIdValue>* Sets the *route* to launch directly into adding an account at a specific financial institution (FI), where *FIIdValue* is the unique internal identifier for the FI you want Account Linking to use when adding an account. ■ */add-account;fiId=<FIIdValue>;newCredential=true* Defaults to *false*. When set to *true*, sets *route* to launch directly into adding an account to a new credential on the specified financial institution without checking for existing credentials. If the *fiId* parameter is not set, *newCredential* is ignored. ■ */add-account;credentialId=<credentialIdValue>* Sets the route to launch directly to the specified credential in a new Connect Accounts view where the user is prompted to either link more accounts to the specified credential or to create new credentials. Note: If the *fiId* attribute is included along with the *credentialId*, *fiId* will be ignored. ■ */edit-credential;credentialId=<credentialIdValue>* Sets the *route* to go directly to editing the Account Credential specified by *<credentialIdValue>*. ■ */edit-credential;credentialId=<credentialIdValue>;viewFirst=true* Defaults to false. When set to true, sets *route* to launch into a Pending Request view for the credential specified by *<credentialIdValue>*. From that view, users can chose to close, withdraw the request, or edit details. ■ */discover-accounts;credentialId=<credentialIdValue>*

Attribute	Req'd?	Parameters and Description
		Sets *route* to display the account discovery step for the Account Credential specified by <*credentialIdValue*> so they can be added to the aggregation system. ■ */edit-account;accountId=<accountIdValue>;viewFirst=true* Defaults to false. When set to true, sets route to launch into an Account Details view for the account specified by <accountIdValue>. From that view, users can chose to close, edit the account, or navigate to the Remove Account view. In each case, *credentialIdValue* is the unique internal identifier for the Account Credential. Optionally, to control whether multiple adds are allowed on an */add-account* route, set *allowMultipleAdds=false* on the */add-account* parameter. When not set, or set to true, Account Linking will provide an option for the user to loop to the FI Select step to enable adding another credential. When set to false, only one credential and its accounts can be added. For example: */add-account:allowMultipleAdds=false*. Note: Existing implementations using the optional *showProgressStep=false* parameter to suppress the final progress step in the workflow take note that **it is being deprecated** because there is no longer a progress step. When this parameter was previously used with */add-account* and set to false it had an additional effect, which was to limit the Add to one credential and its accounts. It will still prevent looping back to the FI Selection step. Explicitly setting *allowMultipleAdds* to true or false will override this *showProgressStep* handling. Existing implementations should transition as soon as possible to using *allowMultipleAdds*.
translate-file-path	No	**Possible Parameters** The path to the custom en.json file. **Description** The *translate-file-path* attribute identifies the URL that is the path to the *en.json* file that you can create to provide custom terminology for certain text and labels. The path is relative to the base *href* of the deployed application.
ui-config-file	No	**Possible Parameters** Path and file name to your custom user interface (UI) configuration file. **Description** The *ui-config-file* attribute identifies the URL that is the path to and the name of the file that contains options for turning on/off UI sections in the Account Linking component. The path is relative to the base *href* of the deployed application.
override-css-file	No	**Possible Parameters** The path and file name for your custom cascading style sheet (CSS) file.

Attribute	Req'd?	Parameters and Description
		Description The *override-css-file* attribute identifies the URL that is the path to and file name of a custom CSS file containing overrides to the default Morningstar Design System (MDS) styles.
custom-fonts	No	**Possible Parameters** The path and name for your font definitions. **Description** The *custom-fonts* attribute identifies the URL that is the path to and file name of your custom font definition file that will override to the default Morningstar Design System (MDS) fonts.
iframe-style	No	**Possible Parameters** Standard CSS inline style values. For example, iframe-style="height:50vh; border:dashed;" **Description** The *iframe-style* attribute applies CSS inline styles to the iframe of the component. Clearing this attribute after it has been set reverts the iframe style of the component back to its original state.

About events

The Account Linking component (*mstar-aggregation-consumer-accountsetup*) can issue two types of events, which are caused by a trigger action.

How an event is resolved depends on the type of event:

- Critical events must be monitored for and managed by the parent page. These events signal that there is an error that the component cannot resolve or that the user has completed their work.
- Standard (informational) events provide information for the parent page. Informational events may be monitored, but are managed by the component. For example, Account Linking handles any problems that occur when the credentials entered for a financial institution cannot be used to successfully authenticate at that institution.

"Monitoring for events" on page 13 provides a code sample for monitoring for a specific event. "Managing critical events" on page 14 describes ways in which the parent page may need to handle the critical events. "Managing standard events " on page 15 describes managing the informational events.

Critical events

The following critical events must be monitored in Account Linking.

Trigger	Name	Event Detail String	Description
A request to the server finds user's authentication context is not valid (resulted in a 401 Unauthorized)	badAuthentication	"Unauthorized access"	User's authentication context is either not valid, the session timed out, or their authentication expired. The user will no longer be able to interact with the component after a *badAuthentication* (401) event until the component receives changes to the *auth-context* attribute that provide valid authentication data. The user's authentication could end at any time. Whenever the parent page receives a *badAuthentication* event it must re-authenticate and invoke a *setAttribute auth-context* with the refreshed header information before the user can proceed with the operation they had started. For information about managing this event, refer to "badAuthentication" on page 14.
Account Linking is not licensed	badConfiguration	"User not licensed"	If the component is not licensed at initiation, there will be a blank area in the parent page where the component is expected. If the configuration changes while the component is in use, the component will be in a blocked state and greyed out on the parent page. For information about managing this event, refer to "badConfiguration" on page 14.
A request to the server resulted in an error other than 401 or an internal code error occurred	internalError	One of: "Forbidden access" "Service unavailable" "Unknown server error" "Unknown error" "Resource not found"	An unexpected error occurred during the user's interaction with the component. The parent page must remedy the problem and possibly display a message to the user. Some causes include: Forbidden access: User does not have the necessary permissions; user may have read only permissions. Service unavailable: Possibly an intermittent service interruption. Resource not found: Could be a misnamed path or file. For information about managing this event, refer to "internalError" on page 14.

Trigger	Name	Event Detail String	Description
User completes action and/or exits the operation	userExit	“User exit”	User has done one of the following: completed the account add, completed the account edit, canceled out of the workflow, or declined the user agreement. For information about managing this event, refer to "userExit " on page 14.

Standard events

The following standard events can occur in Account Linking. These events that may be monitored by the parent page but are handled by the Account Linking component

Trigger	Name	Event Detail String	Description
User makes a change to credential or account	dataChanged	“Data changed”	User makes a change to a credential or account that may result in synchronous changes to any or all of the following: authentication status, accounts linked to the credential, and/or financial data for those accounts.
User initiates an aggregation operation	dataChangedAsync	“Async Data change”	User initiates the service to run the asynchronous process of composite “aggregate” which can be discover/add/aggregate, add/aggregate, or just aggregate.
User initiates an authenticate operation	dataChangedAsyncAuthenticate	“Async Authenticate Data change”	User initiates an authentication operation, which causes the service to run the asynchronous process of attempting to authenticate.
User declines user agreement	userDeclinedAgreement	“User declined user agreement”	User does not accept the terms of the latest user agreement when prompted.

Monitoring for events

This code sample shows how to listen for a specific event. Both components must be monitored for critical events.

```
mstarWebcomp.addEventListener('userExit', function (event) {
   console.log(' @@@@ userExit event is called. detail: ' + event.detail);
});
```

Managing critical events

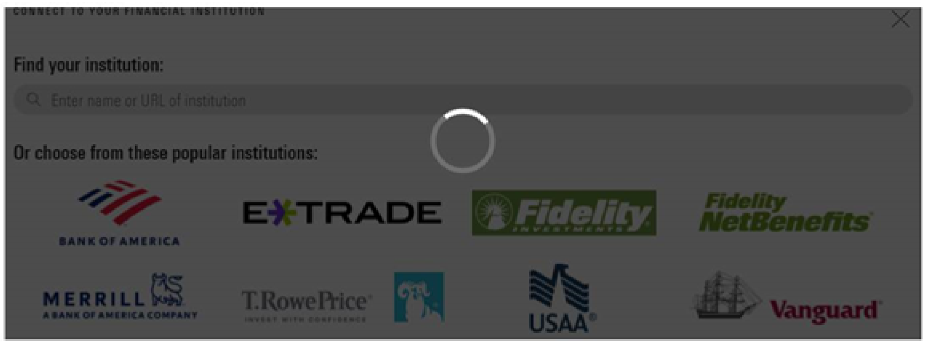
This section describes managing the critical trigger events listed in "About events" on page 11. These critical events must be monitored and managed by the parent page.

badAuthentication

A badAuthentication event can be caused by:

- timeout: the user stops interacting with a component for a period of time and the session for the *auth-context* times out.
- authentication ended: the authorized connection ended.

For badAuthentication events, the parent page must reauthenticate the user and set a new *auth-context*. Until the reauthentication happens, the component is in a blocked state and is greyed out.



When the authentication is reestablished, the user may proceed from where they left off.

badConfiguration

A badConfiguration event is caused by an unlicensed component.

If the component is not licensed at initiation, there will be a blank area in the parent page where the component is expected. If the configuration changes while the component is in use, the component will be in a blocked state and greyed out as it is for a badAuthentication event. The parent page needs to recognize that error and remedy the problem.

internalError

The internalError event can happen for several reasons. For example, a “Resource not found” error would occur if the parent page provided an invalid credential identifier when requesting a *route* to */edit-credential*.

When an internalError event occurs, the user interface for the component remains static. The parent page needs to recognize that error and remedy the problem. The parent page can pass a message to the user, describing the type of error.

userExit

Applies to Account Linking only.

A component sends the userExit event when the user exits it, such as by completing adding an account, completing editing an account, canceling out of the workflow, or declining the user agreement.

Upon an userExit event, the parent page could hide Account Linking or activate it again. To activate it again, if the authorization context is still valid, the parent page can set the *route* attribute to reopen Account Linking at a set point in the workflow.

Managing standard events

This section describes managing the standard (informational) trigger events listed in "About events" on page 11. These events that may be monitored by the parent page but are handled by the Account Linking component. You may want to monitor standard events to update your own back end.

dataChanged

The dataChanged event lets the parent page know something has changed in Account Linking. The parent page should refresh any display of account, credential, position, and transaction data. The event is emitted each time one of these data changes occurs, so the parent page could receive this event multiple times during a single session of the Account Linking component.

dataChangedAsync

The dataChangedAsync event lets the parent page know the user submitted a composite “aggregate” in Account Linking that triggered the service to run the complex and long-running process asynchronously. The composite "aggregate" can be discover/add/aggregate or just aggregate. When the parent page receives this event it should poll for asynchronous activity completion and then query the API to obtain the changed data.

dataChangedAsyncAuthenticate

The dataChangedAsyncAuthenticate event lets the parent page know that an authentication was initiated in Account Linking. The asynchronous process can be complex and long-running. When the parent page receives this event, it can poll for asynchronous activity completion and then query the API to obtain the authentication status.

userDeclinedAgreement

A userDeclinedAgreement event occurs when a user does not accept the terms of a user agreement when it is presented in Account Linking.

Appendix A: Cumulative changes

This appendix keeps a running list of released versions and a summary of the changes in each version starting with version 2.11 for Account Linking.

WARNING: When a release is required, you *must* use the latest version of the component. If you do not, any project you have that uses the component will fail because the new version has changes to the underlying API.

Date	Version	Required?	Summary of Changes
March 24, 2022	2.11	No	This release is not required but is recommended if you are doing customizations. The change in this release removes the ability to add the sidebar feature from all pages, except for the financial institution selection page. Redeploying will provide refined configuration controls in the ui-config.json file and refined text in the en.json file.
April 14, 2022	2.11	No	Documentation update to clarify some details.
May 4, 2022	2.11	No	Documentation fixes in code examples.
June 2, 2022	2.11	No	The *showProgressStep* parameter is being deprecated and will be phased out over time. Existing implementations that use it with */add-account* should transition to using use the new *allowMultipleAdds* parameter as soon as possible.
July 22, 2022	3.0.3	No	Where there are buttons on the page, they now stay in view at the bottom of the page as the page is sized. Morningstar Connect must be upgraded to Version 3.0.3 to enable this change.
August 4, 2022	3.1	No	▪ Internet Explorer (IE) 11 is no longer supported. ▪ The component is now built with Angular 14.
August 25, 2022	3.1	No	Information about known long-term issues at financial institutions is now displayed.
October 6, 2022	3.1	No	The service cannot aggregate data for financial institutions that require the user to

Date	Version	Required?	Summary of Changes
			authenticate, such as by a text message code, every time they log in. For that allow users to disable that feature, Account Linking will return the following message so the user can disable the feature and try again: "Cannot connect. The institution requires you to login on their website and turn off the setting that requires identity verification at every login."
October 27, 2022	3.1	No	An *add-account;fiId=<FIId>;newCredential=true* parameter was added to the *route* attribute to enable accounts to be added to a new credential on a given financial institution without checking for existing credentials.
November 17, 2022	3.2	No	■ The */edit-credential;credentialId=<credentialIdValue>* parameter for the *route* attribute can now include *viewFirst=true* to display a Pending Request. ■ M&T Bank (Banking) can now be found using the search query. ■ A number of changes have been implemented to be compliant with Web Content Accessibility Guidelines (WCAG 2.1).
December 8, 2022	3.3	No	■ An error when using Chrome or Edge with MetaMask enabled has been fixed. ■ A number of changes have been implemented to be compliant with Web Content Accessibility Guidelines (WCAG 2.1).
December 22, 2022	3.3	No	Added the */edit-account;accountId =<accountIdValue* > parameter for the *route* attribute, which can include *viewFirst=true* to display an Account Details view.
January 26, 2023	3.3	No	The margin settings for the component's iframe can now be adjusted. With this

Date	Version	Required?	Summary of Changes

			change, the component's iframe body margin value is 8px by default. Customers who want to have a 0 margin on the outer container can override the CSS class with this value.
			`.cui-wizard-body`
			`{ margin: 0px; }`

February 9, 2023	3.3	No	■ Added */credentialId=<credentialIdValue>* for */add-account* on the *route* parameter. When */add-account;credentialId=<credentialIdValue>* is set, the *route* launches directly to the specified credential in a new Connect Accounts view where the user is prompted to either link more accounts to the specified credential or to create new credentials. ■ Additional changes have been implemented to be compliant with Web Content Accessibility Guidelines (WCAG 2.1).
March 2, 2023	3.3	No	■ Additional changes have been implemented to be compliant with Web Content Accessibility Guidelines (WCAG 2.1). ■ Documentation corrections: version numbers from November 2025 on were corrected in this table.
April 17, 2023	Account Linking 3.4 Account Summary 1.1	Yes *	This release introduces the Account Summary component. ByAllAccounts Connect component has been renamed to Account Linking. * Implementations continuing to run the existing Account Linking component alone can continue to run it without changes. However, anyone implementing the Account Summary component with it must download a new version of the Account Linking

Date	Version	Required?	Summary of Changes
			component. The Account Summary component is implemented in conjunction with Account Linking as described in Morningstar Connect Suite - Account Summary with Account Linking and Morningstar Connect Suite - Account Summary with Account Linking using Morningstar Authentication .
May 25, 2023	Account Linking 3.4	No	Updates for Account Linking: ■ Several status information messages have been updated. ■ An unnecessary scrollbar that appeared when running in Firefox has been removed.
July 27, 2023	Account Linking 3.5	No	Added the *iframe-style* attribute to customize styling for the iframe of the component.
August 17, 2023	Account Linking 3.6	No	Upgraded to the underlying Angular technology. Minor usability improvements.
September 7, 2023	Account Linking 3.6	No	Updates for Account Linking: ■ The language on the User Agreement page and button has been updated. ■ On the landing page, the financial institutions (FIs) displayed are filtered based on whether or not the Firm is enabled for open authorization (OAuth). The default FI list includes Open Banking FIs only for the firms enabled for OAuth and the FI list for the firms not enabled for OAuth has been modified. ■ The list of popular financial institutions (FIs) has changed. Goldman Sachs - Private Wealth Management FTP replaces Goldman Sachs Private Wealth Management - Quicken. For firms that are enabled for open authorization (OAuth), Fidelity Open Banking and Empower Open

Date	Version	Required?	Summary of Changes
			Banking have been added to the popular FIs list.
September 28, 2023	Account Linking 3.6	No	There are a few financial institutions (FIs) that will require a license to use going forward. If your Firm does not have a license for those financial institutions, then you will see the following message when interacting with accounts or credentials for them: "If you are interested in connecting to {FI Name}, please contact your aggregation service provider."
November 9, 2023	Account Linking 3.6	No	Web Content Accessibility Guidelines (WCAG 2.1) compliance changes are complete for Account Summary, so the Morningstar Connect components, including Account Linking, are now fully compliant.
November 30, 2023	Account Linking 3.6	No	Buttons and fonts have been enlarged for design consistency.
December 21, 2023	Account Linking 3.6	No	Changes in this release were made for maintenance purposes.
January 18, 2024	Account Linking 3.6	No	Changes in this release were made for maintenance purposes.
February 8, 2024	Account Linking 3.6	No	Changes to Account Linking support different Referrer Policy settings in customer environments.
February 29, 2024	Account Linking 3.7	No	Changes in this release were made for maintenance purposes.
March 21, 2024	Account Linking 3.7	No	Changes in this release were made for maintenance purposes.
May 2, 2024	Account Linking 3.7	No	Documentation was updated to clarify details about whitelisting.
June 13, 2024	Account Linking 3.7	No	Wait times for open authorization (OAuth) have been improved for the Account Linking component.
August 15, 2024	Account Linking 3.8	No	Changes in this release were made for maintenance purposes.

Date	Version	Required?	Summary of Changes
September 5, 2024	Account Linking 3.8	No	Changes in this release were made for maintenance purposes.
September 26, 2024	Account Linking 3.9	No	The underlying Angular technology has been upgraded.
October 17, 2024	Account Linking 3.9	No	Changes to the component in this release were made for maintenance purposes. Documentation was updated to reflect that names are now prefixed with ByAllAccounts.
November 7, 2024	Account Linking 3.10	No	Changes to the component in this release were made for maintenance purposes.
December 12, 2024	Account Linking 3.10	No	Changes to the component in this release were made for maintenance purposes.
February 20, 2025	Account Linking 3.10	No	Changes to the component in this release were made for maintenance purposes.
April 24, 2025	Account Linking 3.11	No	The underlying Angular technology has been upgraded to Angular 19.
May 15, 2025	Account Linking 3.11	No	The following changes were made in this release. ▪ When users made a change that caused a new form to display, the prior form momentarily flashed before the new one appeared. This issue has been corrected. ▪ When a user exit event was sent to the parent application, any currently displayed form stayed until the parent application removed the widget from the display. Now the form is emptied of content until the parent application removes the widget. ▪ The font and logo have been refreshed.
June 5, 2025	Account Linking 3.11	No	Changes in this release were made for maintenance purposes.
June 27, 2025	Account Linking 3.12	No	Changes in this release were made for maintenance purposes.