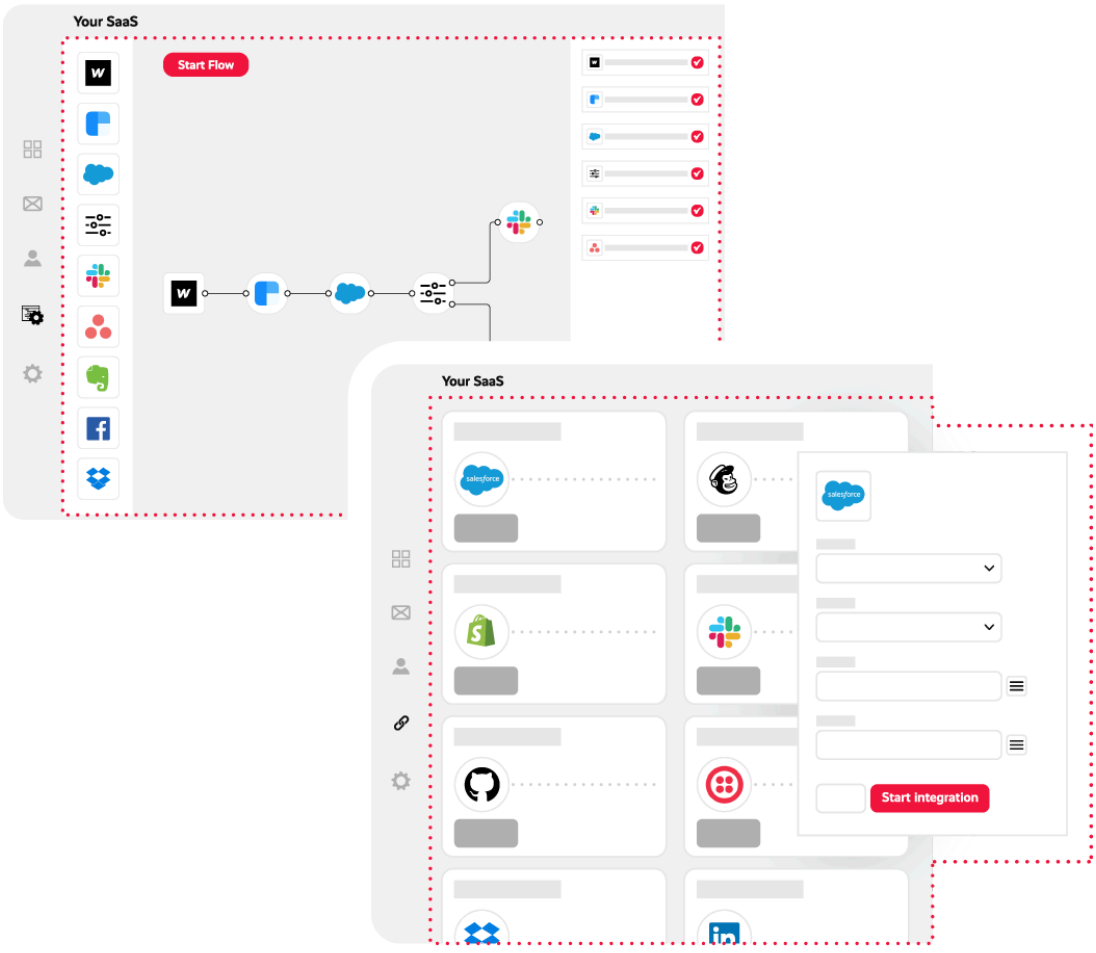
appmixer




# Release Notes

June 9, 2023

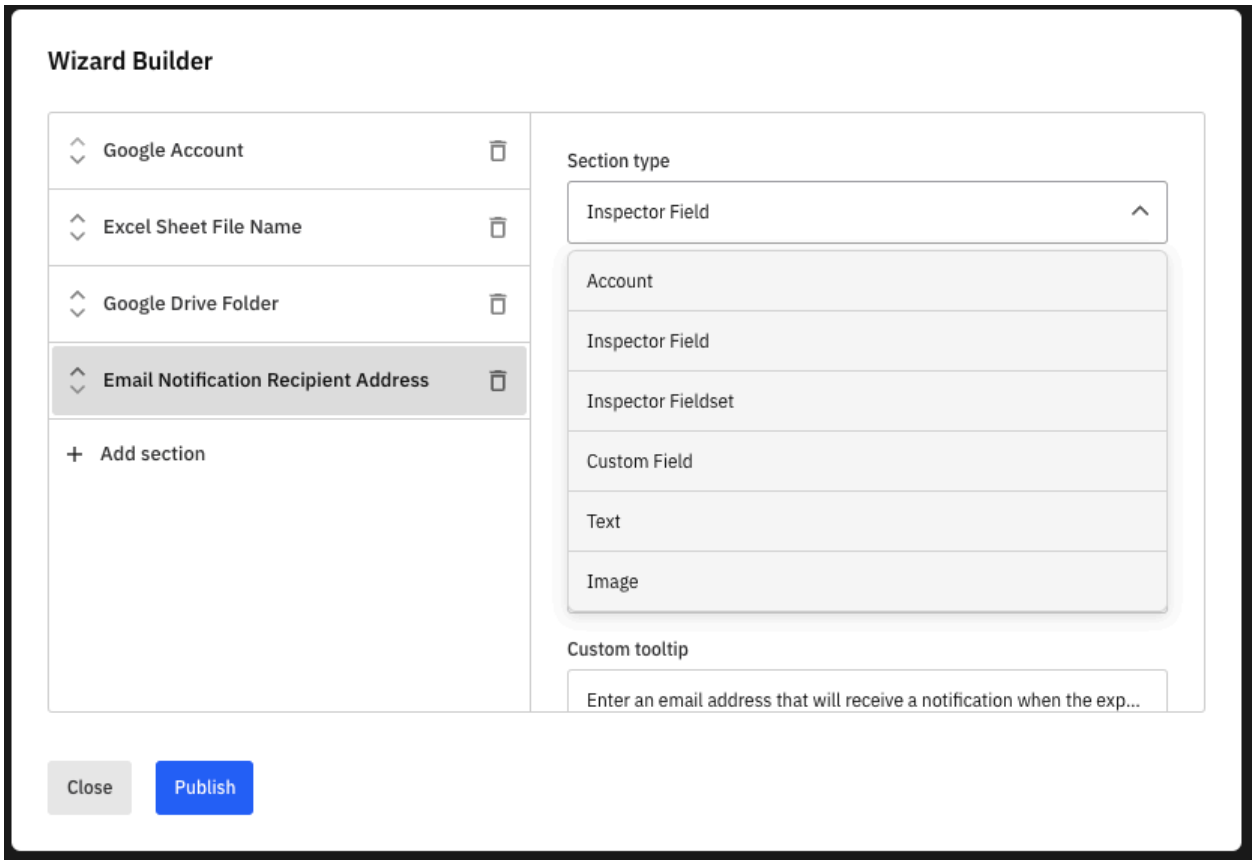
**Version: 5.0**

**New in 5.0** **1**
UI/SDK 1
Core 7
API 11
Backoffice 11
CLI 12
**Documentation** **13**

# New in 5.0

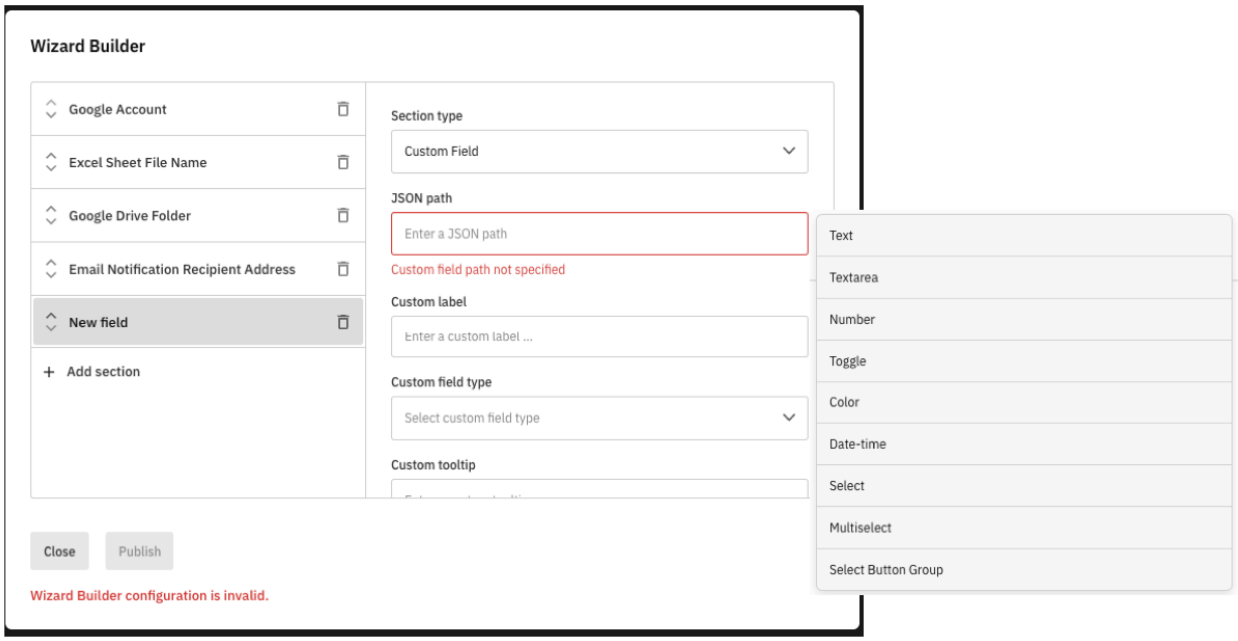
## UI/SDK

- The Wizard configuration now offers enhanced flexibility, allowing the inclusion of the following new sections (as opposed to just inspector fields which was the case in the previous versions of Appmixer):

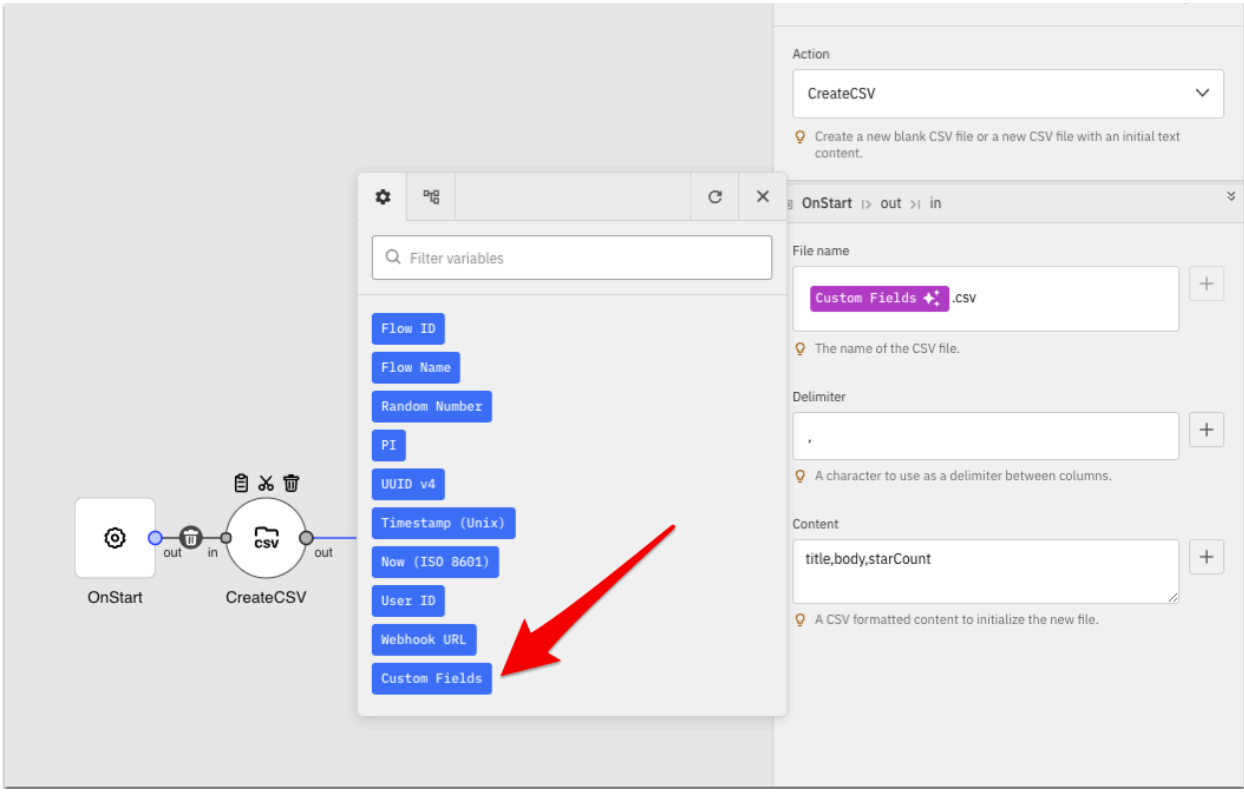



- Account: allows you to add an account field manually. Previously, account fields were automatically added which made it easier to configure wizards but did not allow you to: add different account sections for different components (imagine you have a flow with 2 GDrive components to upload files. Previously, you could only create a wizard that uploads to one GDrive account. Newly, you can add 2 account sections allowing the end-user to upload files to two different GDrive accounts.
- Inspector Field: same as before, i.e. adding a field that represents a configuration field in the Inspector.
- Inspector Fieldset: allows you to add aLL fields of a component automatically. In other words, you don't need to add fields manually if you want the end-user to configure the entire component. This also applies to dynamic fields that you previously could not add to the Wizard (e.g. fields for adding all column values of the google.spreadsheets.CreateRow component - which are loaded dynamically based on the sheet/worksheet selected by the user and therefore are not known at design time).
- Custom Field: allows you to add a completely custom field of any type (Text, Textarea, Number, Toggle, Color, Date-time, Select, Multiselect, Select button group) to collect your own custom data from the end-user:

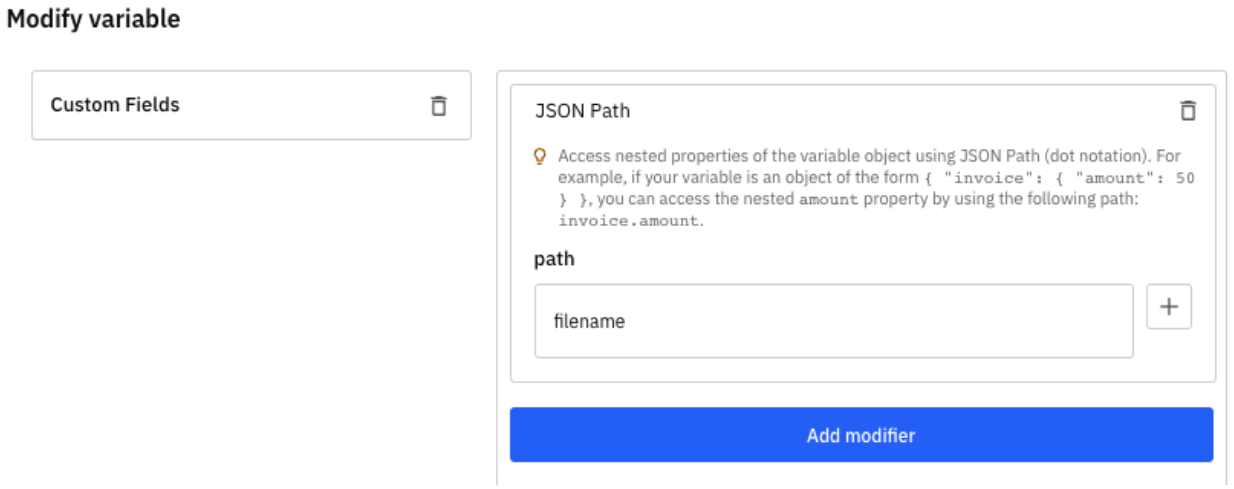
© 2023 client IO




You name your field in the “JSON path” input and then refer to it by adding the “Custom Fields” variable in your inspector fields:




And applying the “JSON path” modifier to it to select the field that you defined in the Wizard builder:




- Text: add header/subheader or a paragraph of custom text to your Wizard. This is great to give additional textual information to your end-users.
- Image: add any custom image to your wizards. This is great to give your end-users additional information about the integration, usually in the form of icons or infographics.

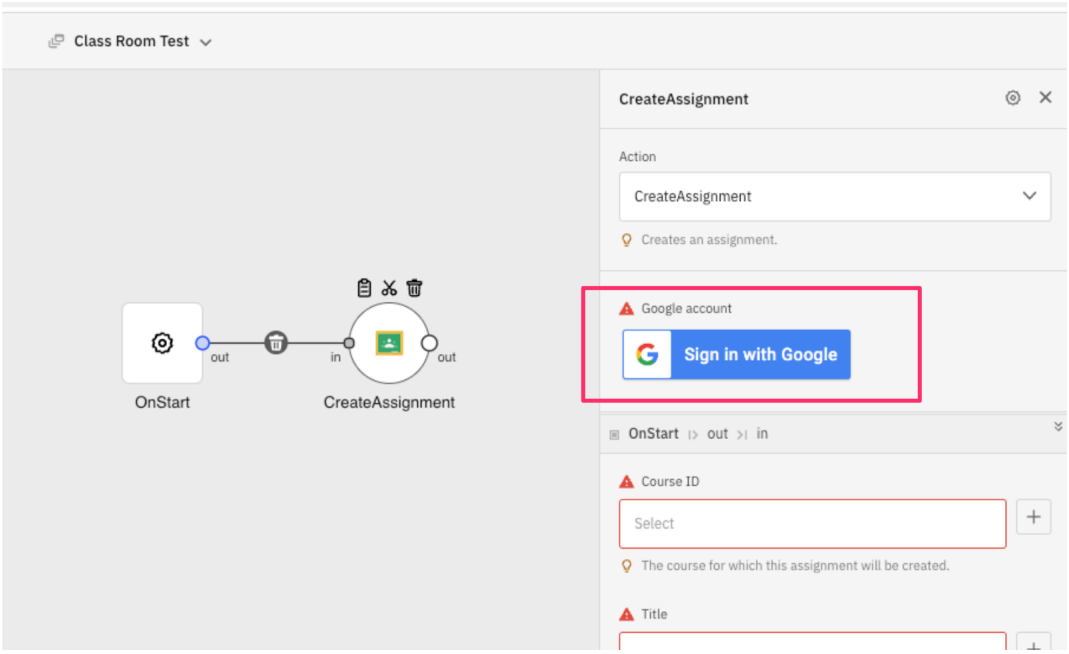
- Connecting new accounts uses a minimal scope. For example, the *appmixer.google.gmail* module has several components. The *NewEmail* component requires Google authentication scopes:
  - "https://mail.google.com/",
  - "https://www.googleapis.com/auth/gmail.readonly"

  And the SendEmail component requires scopes:
  - "https://mail.google.com/",
  - "https://www.googleapis.com/auth/gmail.compose",
  - "https://www.googleapis.com/auth/gmail.send"

  In the previous version of Appmixer, when any Google Gmail component was authenticated, the system joined all the Google Gmail scopes together. That means that even if the user used only NewEmail and not the SendEmail component, the system asked for permission to send emails. Newly, only the scope for the component that is being authenticated will be used. This allows the user to give Appmixer only the permissions it needs to execute the particular flow/integration.
  Note that if the user wants to use multiple components from the same module with different scopes, they will have to authenticate them all, and the system will create multiple tokens with the appropriate scopes.

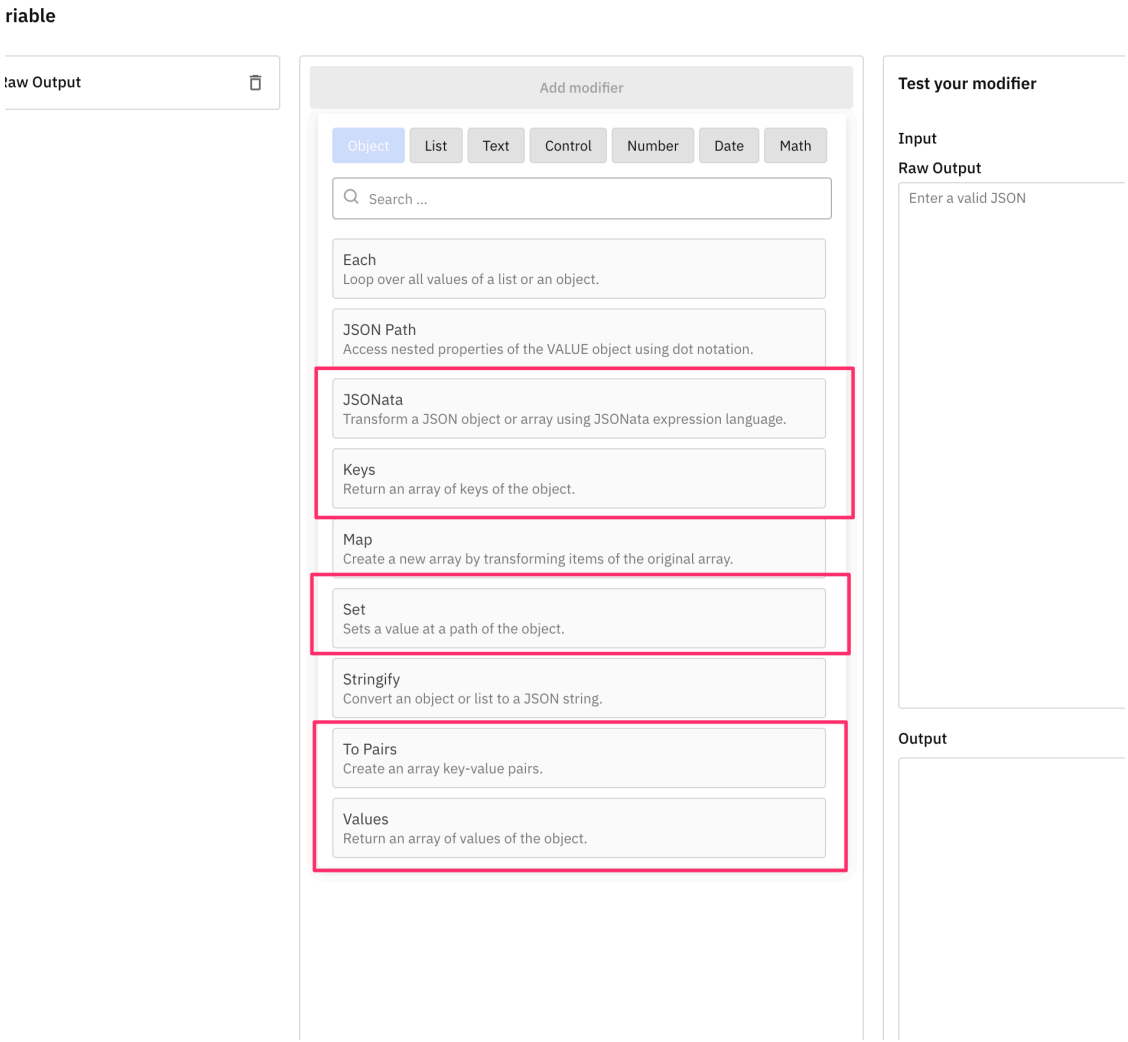
© 2023 client IO

- Custom "Connect account" button in Inspector/Wizard: allows you to customize the button to authenticate to components. This is especially useful to make sure that connectors follow the developer guidelines of the vendors of the 3rd party apps, therefore making it easier for your own apps to pass the vendor's verification procedures.

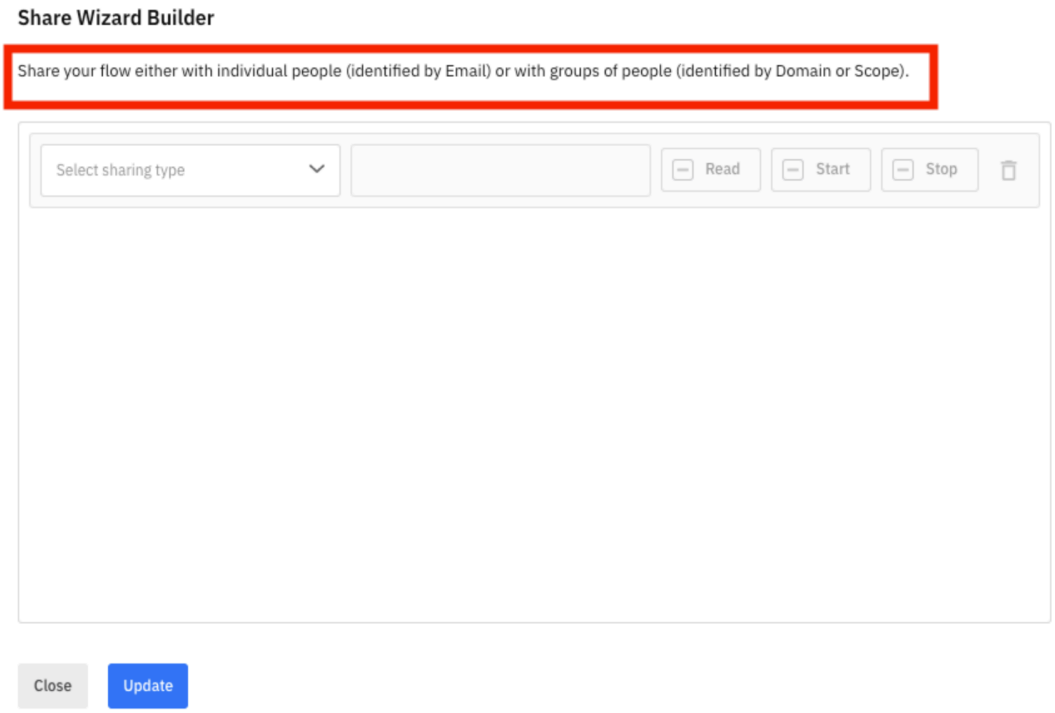



https://docs.appmixer.com/appmixer/v/5.0/component-definition/authentication#custom-connect-account-button

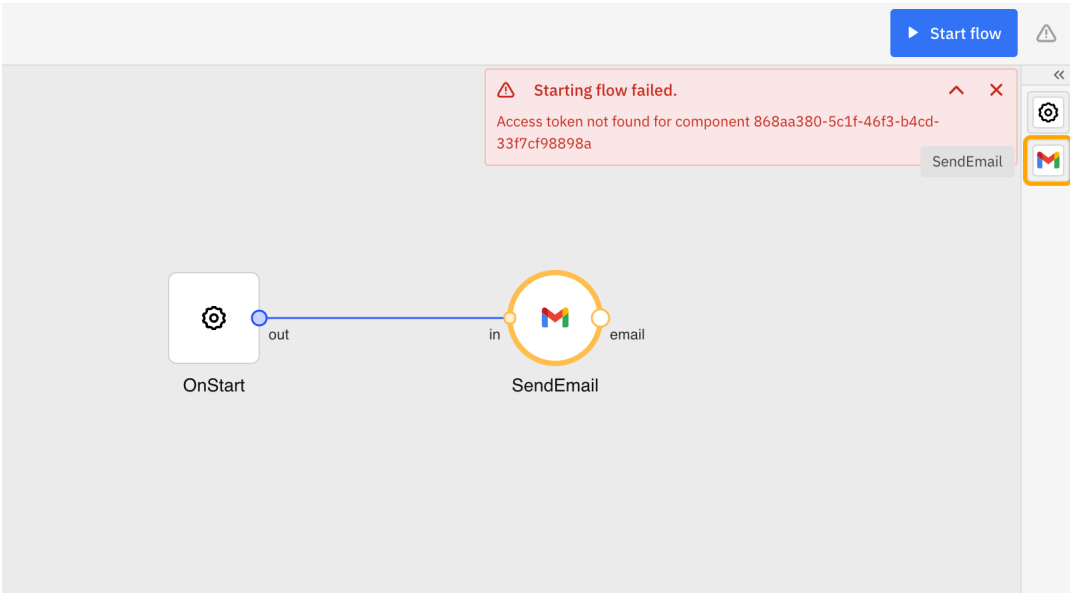
- New Object Modifiers:




- The JSON Path modifier now supports all JSON Path expressions. (https://github.com/dchester/jsonpath)
- The *Share* dialog text changed:




- Error message boxes now contain more information about the reason of the errors:




- Authentication popups can be fully customized. https://docs.appmixer.com/appmixer/v/5.0/customizing-ui/custom-auth-popups

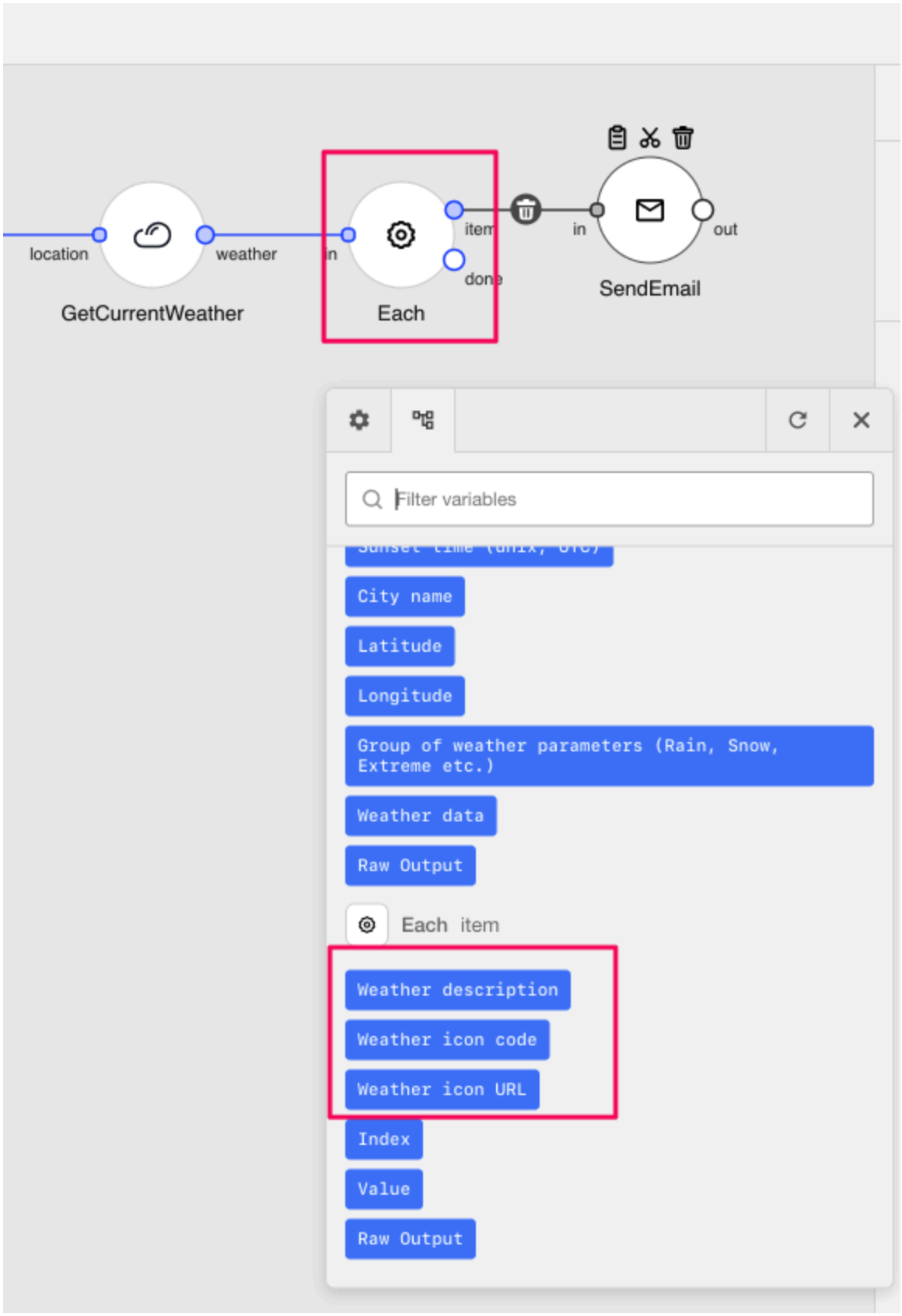
# Core

- Upgraded to Node 18 (LTS). This also means that all the components need to be compatible with Node 18.

- New function context.replaceFileStream(fileId, content). https://docs.appmixer.com/appmixer/v/5.0/component-definition/behaviour#async-context.replacefilestream-fileid-stream
- New System Configuration options (https://docs.appmixer.com/appmixer/v/5.0/appmixer-backoffice/system-configuration):
    - COMPONENT_RECEIVE_TIMEOUT (defaults to 23 minutes). The receive() function should return within that timeout. Otherwise, an error will be thrown and the input message will be retried.
    - COMPONENT_FACTORY_TIMEOUT (defaults to 5 minutes). If the engine fails to create a component instance within this time, an error will be thrown. The input message will be retried.
    - BROKER_MESSAGE_ACK_TIMEOUT
- *Textarea* input type added to the connector authentication popup (Used in MongoDB connector, for example).
- Output port schema options. Newly, the component output options can be defined by using JSON schema. This is especially useful if your components return arrays and you want your end-users to be able to easily work with the properties of the array item objects. Before, there was no easy way for end-users to know what properties that can point to (e.g. by using the JSON Path modifier) inside array items. With the JSON schema defined for array items, the user can newly easily select the array item properties either in the variables picker of components following the “Each” utility (i.e. when looping over the array items) or in the modifiers (Map, JSON Path, …).

© 2023 client IO

```
{
    "outPorts": [
        {
            "name": "weather",
            "options": [
                { "label": "Temperature", "value": "main.temp" },
                { "label": "Pressure", "value": "main.pressure" },
                { "label": "Humidity", "value": "main.humidity" },
                { "label": "Sunrise time (unix, UTC)", "value": "sys.sunrise" },
                { "label": "Sunset time (unix, UTC)", "value": "sys.sunset" },
                { "label": "City name", "value": "name" },
                {
                    "label": "Weather data",
                    "value": "weather",
                    "schema": {
                        "type": "array",
                        "items": {
                            "type": "object",
                            "properties": {
                                "description": { "type": "string", "title": "Weather de
                                "icon": { "type": "string", "title": "Weather icon code
                                "iconUrl": { "type": "string", "title": "Weather icon U
                            }
                        }
                    }
                }
            ]
        }
    ]
}
```

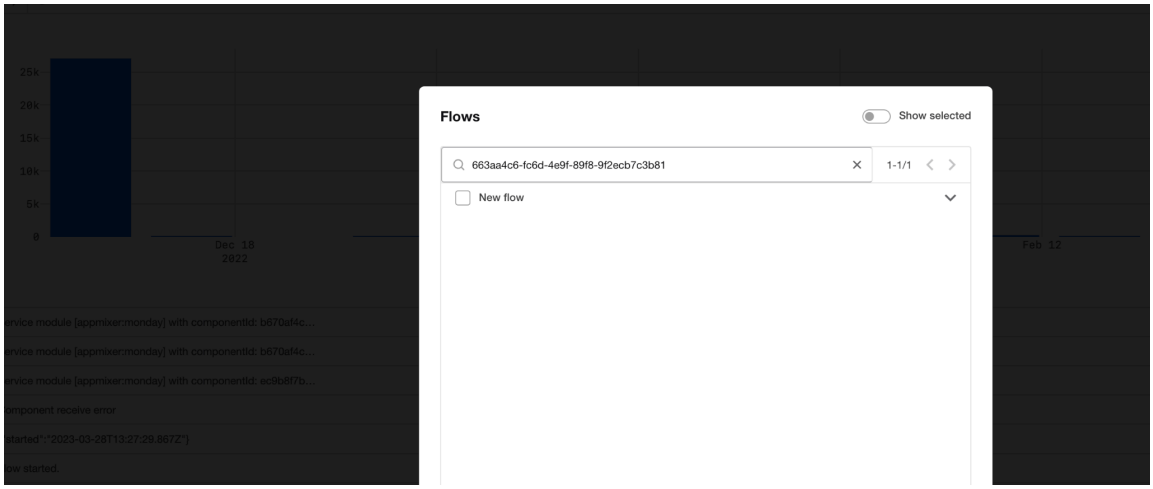

location
weather
GetCurrentWeather
in
item
done
Each
in
out
SendEmail
Filter variables
City name
Latitude
Longitude
Group of weather parameters (Rain, Snow, Extreme etc.)
Weather data
Raw Output
Each item
Weather description
Weather icon code
Weather icon URL
Index
Value
Raw Output

https://docs.appmixer.com/appmixer/v/5.0/component-definition/manifest/outports#json-schema

- Error log from a component is by default limited to 51200 bytes. This can be newly controlled by the LOG_COMPONENT_ERROR_MAX_BYTES system configuration option.

# API

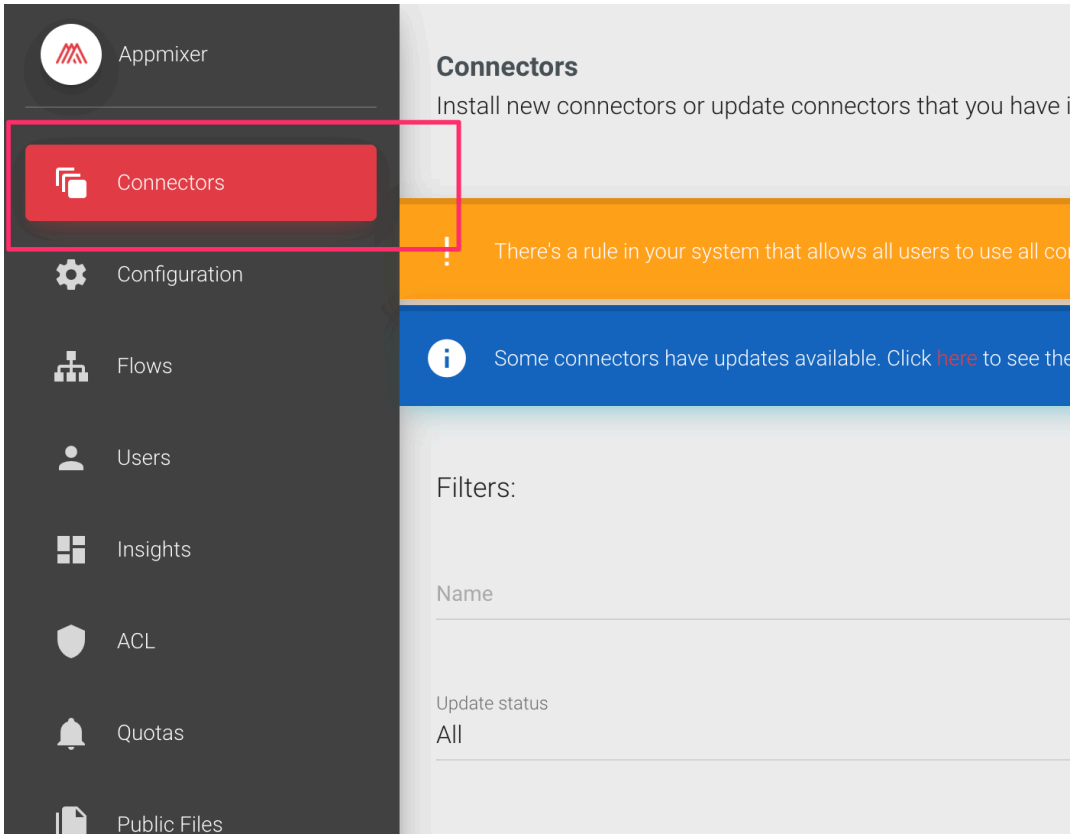
- GET /auth/flow/{flowId} response object structure has changed. More information in https://docs.appmixer.com/appmixer/v/5.0/migration-from-4.5#get-auth-flow-flowid
- The GET /flows and GET /flows/count *pattern* option searches through *flowId* as well now:



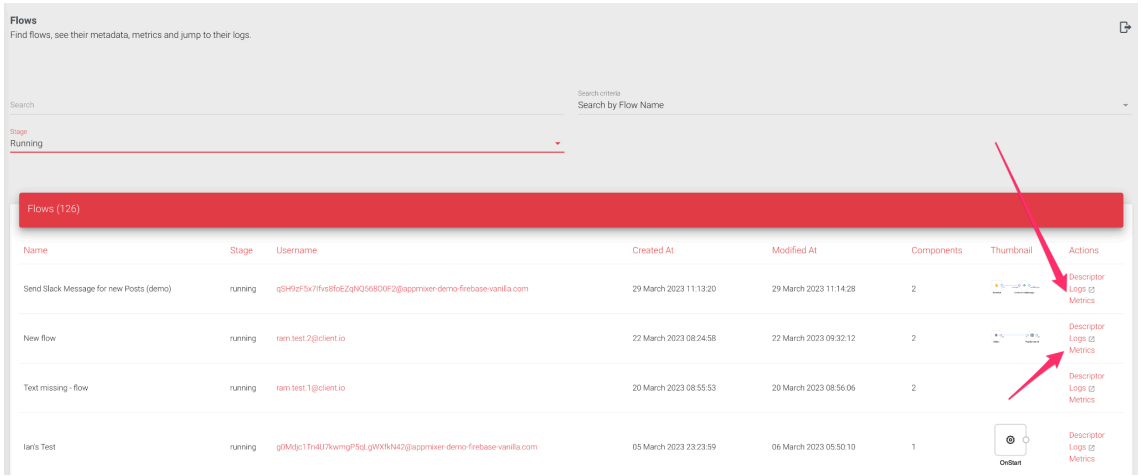

More in https://docs.appmixer.com/appmixer/v/5.0/api/flows

# Backoffice

- Modules were renamed Connectors.




- New *Actions* for Flows. *Logs* link to the *Insights* page in the Appmixer Studio with the flow preselected making it possible for admin users to explore the flow logs and therefore provide better customer support for end-users. *Metrics* will show the number of messages this flow generated.




# CLI

- Upgraded to Node 18

- context.setTimeout() supported
- context.lock() supported

# Documentation

Please visit https://docs.appmixer.com for full documentation.

*The material in this document is for informational purposes only and is subject to change without notice. While reasonable efforts have been made in the preparation of this document to assure its accuracy, client IO makes no representation or warranty expressed, implied, or statutory, as to its completeness, accuracy, or suitability, and assumes no liability from the use of the information contained herein.*

*This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by patent, copyright, trademark, and other intellectual property laws. Except as expressly provided in any written license agreement from client IO, the furnishing of this document does not give you any license to these patents, copyrights, trademarks, or other intellectual property.*

*© 2023 client IO. All rights reserved.*