Tonestream

BAI Parser Release Notes Copyright © 2025 OneStream Software LLC. All rights reserved.

Any warranty with respect to the software or its functionality will be expressly given in the Subscription License Agreement or Software License and Services Agreement between OneStream and the warrantee. This document does not itself constitute a representation or warranty with respect to the software or any related matter.

OneStream Software, OneStream, Extensible Dimensionality, and the OneStream logo are trademarks of OneStream Software LLC in the United States and other countries. Microsoft, Microsoft Azure, Microsoft Office, Windows, Windows Server, Excel, Internet Information Services, Windows Communication Foundation and SQL Server are registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. DevExpress is a registered trademark of Developer Express, Inc. Cisco is a registered trademark of Cisco Systems, Inc. Intel is a trademark of Intel Corporation. AMD64 is a trademark of Advanced Micro Devices, Inc. Other names may be trademarks of their respective owners.

Table of Contents

Release PV8.0.0 SV104	 1
Prerequisites	1
Fixed Issues	1
Contact Support	1
Release PV8.0.0 SV103	 2
Prerequisites	 2
Fixed Issues	 2
Release PV800 SV102	3
Prerequisites	3
New Features	3
Enhancements	3
Fixed Issues	 3
Known Issues	 4
Release PV800 SV101	7
Prerequisites	 7
New Features	 7
Enhancements	7
Fixed Issues	 7
Known Issues	 8

Table of Contents

Release PV800 SV100	1	1
Prerequisites	1	1
Known Issues	1	1

Release PV8.0.0 SV104

The theme of this Hotfix release is bug fixes.

NOTE: Only the Release Notes were updated for this release. The BAI Parser Solution Guide has not been updated except to change the release number and copyright if needed.

Prerequisites

- This solution is compatible with OneStream Platform v8.
- This release requires a minimum 8.0 Platform release.

Fixed Issues

 The BAI Parser Viewer does not freeze and processes all files accordingly regardless of the fund type of the accounts within a file. (MP3-6887)

Contact Support

Contact OneStream Support by registering at:

Support - OneStream Software

Release PV8.0.0 SV103

The theme of this Hotfix release is bug fixes.

NOTE: Only the Release Notes were updated for this release. The BAI Parser Solution Guide has not been updated except to change the release number and copyright if needed.

Prerequisites

- This solution is compatible with OneStream Platform v8.
- This release requires a minimum 8.0 Platform release.

Fixed Issues

 The BAI parser will successfully map files with fund type 'D'. All related formatting, including customer references and transaction detail values, will be mapped in the file accordingly. (MP3-6809)

Release PV800 SV102

The theme of this release are enhancements.

NOTE: Only the Release Notes were updated for this release. The BAI Parser Solution Guide has not been updated except to change the release number and copyright if needed.

Prerequisites

- This solution is compatible with OneStream Platform v8.
- This release requires a minimum 8.0 Platform release.

New Features

None

Enhancements

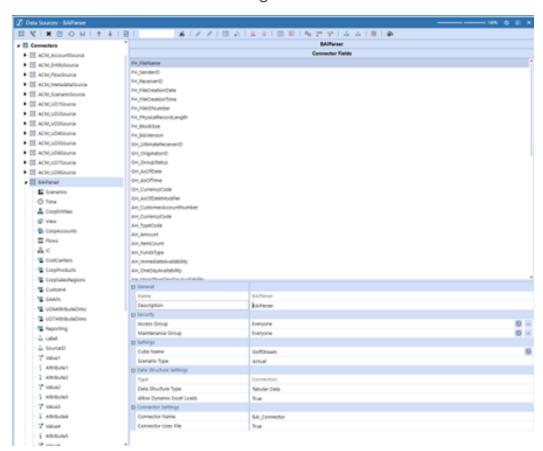
- When imported, files with warnings will still display all available information so you can review your data. (MP3-6350)
- When Account Identifier and Fund Type fields are not valid on imported files, the file will still be parsed with warnings. (MP3-6328, MP3-6348)

Fixed Issues

None

Known Issues

- Automation of the data load is impacted by the Connector Uses File setting.
 To automate a file, follow the steps below:
 - 1. Navigate to the Data Source using the BAI Parser Solution.
 - 2. Set the Connector Uses File setting to False.



- 3. Create a copy of the BAI_Connector Connector Business Rule.
- 4. Comment out line 42.

```
DIX O BIMIV VIDI &
► in Finance
                                   Properties Formula
Farser
                                                # # % | Œ # | ¥ 60
   ACM_AccountSource
                                              private System.Text.StringBuilder string * new System.Text.StringBuilder();
private string strBRName * "BAI_Connector";
    ACM_EntitySource
                                                  public object Main(SessionInfo si, 8MGlobals globals, Transformer api, ConnectorArgs args) {

√x ACM_FlowSource

    ACM_MetadataSource
                                                      string.AppendLine($"(str@RName) - Working on (args.ActionType)");
    ACM_ScenarioSource
                                                      var bai = new Morkspace.8AI.Assembly_BAI.BusinessRule.DashboardDataSet.8AI_MelperQueries.MainClass();
    ACM_UD1Source
    ACM_UD2Source
    ACM_UDBSource
    ACM_UD4Source

√A ACM_UDSSource

                                                              case ConnectorActionTypes.GetFieldList:
                                                                return bal.GetBAIColMeader();
)
    ACM_UD6Source
    ACM_UD7Source
    ACM_UD8Source
                                                              case ConnectorActionTypes.GetData:
    √ BAI_Connector
                                                                     bai.SI = si;
DataTable dt = bai.GctDataFromFile(api.FileInfo.SourceFileFath);
    √x BiBlendFayment
                                                                      api.Parser.ProcessDataTable(si, dt, false, api.ProcessInfo);
    CPP CapitalPlanningData
    CPP_CapitalPlanningData_EntityTi
    √K CPP_CapitalPlanningData_EntityTc
    CSP_CashPlanningData
    MME ConnectorTemplate
    MME_ConnectorTemplate

GSP_DataUnitConnector

√ RaymentMomtHouston

                                                              case ConnectorActionTypes.GetOrillSack:
```

NOTE: If you have an automated process that moves files to the Harvest Folder, you must add a step that also copies the file to a new folder in the OneStream File Explorer. This step must occur before the connector rule runs, so it is recommended to place this code in an extensibility rule.

Add code to the new connector rule that reads the file from your chosen
 OneStreamFile Explorer folder.

NOTE: Below is an example of how you can access the folder structure under a specific workflow name:

string IncomingFolder = BRApi.Utilities.GetFileShareFolder (si,FileShareFolderTypes.ApplicationIncoming, api.WorkflowProfile);

6. After defining the file path, use this code to process the file:

DataTable dt = bai.GetDataFromFile(SourceFilePath);

Release PV800 SV101

The theme of this release is bug fixes.

NOTE: Only the Release Notes were updated for this release. The BAI Parser Solution Guide has not been updated except to change the release number and copyright if needed.

Prerequisites

- This solution is compatible with OneStream Platform v8.
- This release requires a minimum 8.0 Platform release.

New Features

None

Enhancements

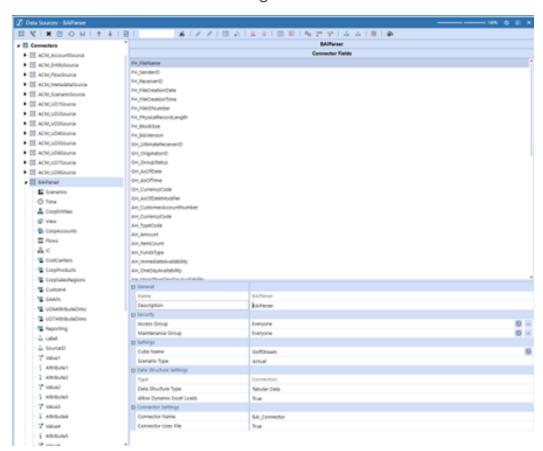
None

Fixed Issues

- BAI files uploaded without type codes parse based on all other codes available in the document. (MP3-4736)
- BAI files uploaded with empty, optional fields will parse without error. (MP3-4735)

Known Issues

- Automation of the data load is impacted by the Connector Uses File setting.
 To automate a file, follow the steps below:
 - 1. Navigate to the Data Source using the BAI Parser Solution.
 - 2. Set the Connector Uses File setting to False.



- 3. Create a copy of the BAI_Connector Connector Business Rule.
- 4. Comment out line 42.

```
DIX ORIMIV VIDIP
► in Finance
                                   Properties Formula
Farser
                                                # # % | Œ # | ¥ 60
    ACM_AccountSource
                                              private System.Text.StringBuilder string * new System.Text.StringBuilder();
private string strBRName * "BAI_Connector";
    ACM_EntitySource
                                                  public object Main(SessionInfo si, 8MGlobals globals, Transformer api, ConnectorArgs args) {

√x ACM_FlowSource

    ACM_MetadataSource
                                                      string.AppendLine($"(str@RName) - Working on (args.ActionType)");
    ACM_ScenarioSource
                                                       var bai = new Morkspace.8AI.Assembly_BAI.BusinessRule.DashboardDataSet.8AI_MelperQueries.MainClass();
    ACM_UD1Source
    ACM_UD2Source
    ACM_UDBSource
    ACM_UD4Source

√A ACM_UDSSource

                                                              case ConnectorActionTypes.GetFieldList:
                                                                 return bal.GetBAIColMeader();
)
    ACM_UD6Source
    ACM_UD7Source
    ACM_UD8Source
                                                              case ConnectorActionTypes.GetData:
    √ BAI_Connector
                                                                      bai.SI = si;
DataTable dt = bai.GctDataFromFile(api.FileInfo.SourceFileFath);
    √x BiBlendFayment
                                                                      api.Parser.ProcessDataTable(si, dt, false, api.ProcessInfo);
    CPP CapitalPlanningData
    CPP_CapitalPlanningData_EntityTi
    √K CPP_CapitalPlanningData_EntityTc
     CSP_CashPlanningData

    ✓ OSP_DataUnitConnector
    ✓ PaymentMgmtHouston

    MME ConnectorTemplate
                                                              case ConnectorActionTypes.GetOrillSack:
```

NOTE: If you have an automated process that moves files to the Harvest Folder, you must add a step that also copies the file to a new folder in the OneStream File Explorer. This step must occur before the connector rule runs, so it is recommended to place this code in an extensibility rule.

Add code to the new connector rule that reads the file from your chosen
 OneStreamFile Explorer folder.

NOTE: Below is an example of how you can access the folder structure under a specific workflow name:

string IncomingFolder = BRApi.Utilities.GetFileShareFolder (si,FileShareFolderTypes.ApplicationIncoming, api.WorkflowProfile);

6. After defining the file path, use this code to process the file:

DataTable dt = bai.GetDataFromFile(SourceFilePath);

Release PV800 SV100

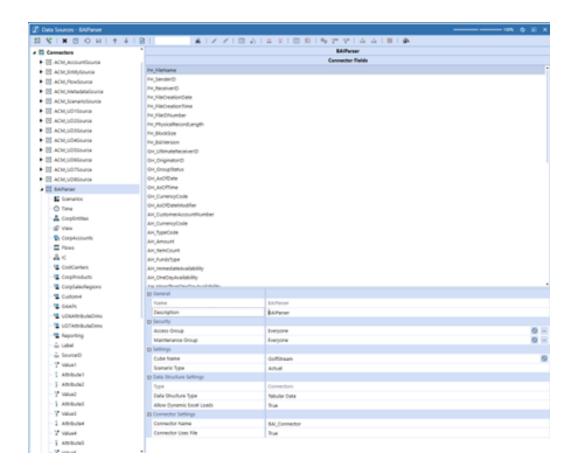
This is the initial release of the BAI Parser Solution and documentation for installation guidance.

Prerequisites

- This solution is compatible with OneStream Platform v8.
- This release requires a minimum 8.0 Platform release.

Known Issues

- Automation of the data load is impacted by the Connector Uses File setting.
 To automate a file, follow the steps below:
 - 1. Navigate to the Data Source using the BAI Parser Solution.
 - 2. Set the Connector Uses File setting to False.



- 3. Create a copy of the BAI_Connector Connector Business Rule.
- 4. Comment out line 42.

```
DIX ORIMIV VIDIP
► in Finance
                                  Properties Formula
Farser
                                               # # % | Œ # | ¥ 60
   ACM_AccountSource
                                             private System.Text.StringBuilder string * new System.Text.StringBuilder();
private string strBRName * "BAI_Connector";
    ACM_EntitySource
                                                 public object Main(SessionInfo si, 8MGlobals globals, Transformer api, ConnectorArgs args) {

√x ACM_FlowSource

    ACM_MetadataSource
                                                     string.AppendLine($"(str@RName) - Working on (args.ActionType)");
    ACM_ScenarioSource
                                                     var bai = new Morkspace.8AI.Assembly_BAI.BusinessRule.DashboardDataSet.8AI_MelperQueries.MainClass();
    ACM_UD1Source
    ACM_UD2Source
    ACM_UDBSource

√A ACM_UD45ource

√A ACM_UDSSource

                                                             case ConnectorActionTypes.GetFieldList:
                                                               return bal.GetBAIColMeader();
)

√S ACM_UD6Source

√€ ACM_UD7Source

    ACM_UD8Source
                                                             case ConnectorActionTypes.GetData:
    √ BAI_Connector
                                                                    bai.SI = si;
DataTable dt = bai.GctDataFromFile(api.FileInfo.SourceFileFath);
    √x BiBlendFayment
                                                                    api.Parser.ProcessDataTable(si, dt, false, api.ProcessInfo);
    CPP CapitalPlanningData
    CPP_CapitalPlanningData_EntityTi
    √K CPP_CapitalPlanningData_EntityTc
    CSP_CashPlanningData
    MME ConnectorTemplate

√ RaymentMomtHouston

                                                             case ConnectorActionTypes.GetOrillSack:
```

NOTE: If you have an automated process that moves files to the Harvest Folder, you must add a step that also copies the file to a new folder in the OneStream File Explorer. This step must occur before the connector rule runs, so the code needs to be placed in an extensibility rule.

- Add code to the new connector rule that reads the file from your chosen OneStreamFile Explorer folder.
- 6. After defining the file path, use this code to process the file:

```
DataTable dt = bai.GetDataFromFile(SourceFilePath);
```

NOTE: This is an example of how to access the folder structure under a specific workflow name:

string IncomingFolder = BRApi.Utilities.GetFileShareFolder

(si,FileShareFolderTypes.ApplicationIncoming, api.WorkflowProfile);