

Inhaltsverzeichnis

- Allgemein** 1
- Was tut das Plugin?** 1
- Funktionen** 1
- Unterstützte Inhalte** 1
- Zweck & Anwendung** 2
 - Zweck 2
 - Anwendung 2
- Installation** 2
 - Anforderungen 2
- Konfiguration** 2
 - Konfigurationsdatei verwenden 2
 - Struktur 2
 - Servers-Element 3
 - Server-Element 3
 - Trigger-Element 5
 - File-Element 6
 - Bindings-Element 7
 - Binding-Element 8
 - Node Query Expression 9
 - Benutzung 10
 - Synchronisation 10
 - Beispiel Konfigurationsdatei 10
 - Beispiel Konfigurationsschema-Datei 11
- Fehlerdiagnose** 15
- Entities** 15
- Ordner & Dateien** 16
 - Ordner 16
 - Dateien 16
- Versionsinformation** 16
 - Dieses Dokument 16
 - Plugin 16
 - Assembly 16

CSV Exchange Plugin

Allgemein

Das CSV Exchange Plugin stellt einen Datenaustauschmechanismus zwischen CoDaBix® und CSV (comma-separated values) -basierten Dateiformaten bereit.

Was tut das Plugin?

Das CSV Exchange Plugin definiert eine Speicherstruktur für CSV-Dateien. Innerhalb dieser Struktur können ein oder mehrere Datensets definiert werden. Jedes Datenset repräsentiert eine Verbindung zwischen den Spalten im CSV-Speicher und den Nodes in CoDaBix®. Über diese Verbindung liest und schreibt das Plugin die CoDaBix® Nodewerte als Datenset in die CSV-Datei und andersherum.

Funktionen

- Überwachung von Dateiveränderungen
- Dateien nach Beendigung verwerfen
- Dateien nach Beendigung löschen
- Benutzerdefinierte CSV Separatorenkonfiguration
- Relatives Nodeverbindung in DataSet
- Zugriff auf Dateien im lokalen Dateisystem oder über ein SSH-basiertes Transferprotokoll (SCP oder SFTP)

Unterstützte Inhalte

- Microsoft Excel CSV Separator (Standard)
- Benutzerdefinierte CSV Separatoren

Zweck & Anwendung

Zweck

Die konfigurierten CSV-Dateien können einfach zum Lesen und Schreiben von Daten in und aus CoDaBix® verwendet werden. Dies erlaubt einfache Datenimport-/exportszenarios unter Verwendung von kommaseparierten Werten (comma-separated values = CSV). Auch können andere CoDaBix® Teilnehmer durch das CSV Exchange Plugin in einer CSV-Datei gespeicherte Informationen abliefern und abholen.

Anwendung

- Automatische Datensynchronisation zwischen CoDaBix® und Fremd(unter)systemen zur Datenübertragung
- Datenimport zur Archivierung von Prozessprotokollen
- Datenexport zur nachverarbeitung von CoDaBix®-produzierten und -gespeicherten Informationen

Installation

Dieses Plugin ist Bestandteil des CoDaBix® Setups. Bitte konsultieren Sie [CoDaBix® Setup und erster Start](#) für weitere Informationen darüber, wie dieses Plugin installiert und deinstalliert werden kann.

Anforderungen

- Standardanforderungen von CoDaBix®
- Lese-/Schreibzugriff auf die konfigurierten CSV-Dateien

Konfiguration

Konfigurationsdatei verwenden

Struktur

Das CSV Exchange Plugin definiert die Wurzel seines Elementenbaums über das PluginSettings-Element, wie in [Plugin Konfiguration - Verwenden einer Konfigurationsdatei](#)

beschrieben, und setzt seinen XML-Baum durch Benutzen des Servers Elements fort.

Servers-Element

Das Servers-Element dient als Container für ein oder mehrere Server-Elemente. Dieses Element verwaltet alle Dateiserver, die dem Exchange Plugin zugeordnet sind.

Das Servers-Element kann wie folgt aussehen:

```
<Servers>
  <!-- 0-n Server elements -->
</Servers>
```

Server-Element

Das Server-Element dient als Container für das Element File sowie für ein oder mehrere Trigger-Elemente, falls das SyncMode-Attribut auf SystemToFile gesetzt ist. Ein Server identifiziert eine CSV-Datei, zu der die Serverkonfiguration gehört. Die Konfiguration beinhaltet auch die Aktionen, die vor, während und nach dem Synchronisationsprozess ausgeführt werden müssen. Diese Information wird daher vom CSV Exchange Plugin genutzt, um CSV-Daten mit CoDaBix® auszutauschen.

Jedes Server-Element stellt folgende Liste an Attributen zur Verfügung:

	Verpflichtend	Typ	Zweck
IsEnabled	nein	Boolean	Zeigt den Wert true, wenn der Datenaustausch für den Dateiserver aktiv ist; andernfalls zeigt es den Wert false (Standard: true).
SyncMode	nein	CSV Sync Mode	Die Richtung, in die die Synchronisierung durchgeführt werden soll.
Zusätzliche Attribute für SyncMode="FileToSystem"			
SyncTrigger	nein	CSV Sync Trigger	Die Auslösebedingung, die die Synchronisierung einleitet.
BeforeSyncAction	nein	CSV Sync Action	Die Aktion, die ausgeführt werden muss, bevor der Synchronisationsprozess beginnt.
AfterSyncAction	nein	CSV Sync Action	Die Aktion, die ausgeführt werden muss, nachdem der Synchronisationsprozess geendet hat.

	Verpflichtend	Typ	Zweck
BeforeSyncMoveFileTo (falls BeforeSyncAction="MoveFile")	ja	Node Query Expression	Gibt den neuen vollqualifizierten Datei- oder Verzeichnispfad an, in den die Datei verschoben/umbenannt werden soll. Wenn Sie einen Verzeichnispfad angeben (der mit / oder \ endet), wird der ursprüngliche Dateiname beibehalten, wenn die Datei in das angegebene Verzeichnis verschoben wird. Falls in diesem Fall bereits eine Datei mit dem gleichen Namen existiert, wird an den Dateinamen ein Suffix wie (1), (2) usw. angehängt.
AfterSyncMoveFileTo (falls AfterSyncAction="MoveFile")	ja	Node Query Expression	Siehe oben.

CSV Sync Mode

Die folgenden Werte sind gültig für Attribute dieses Typs:

Wert	Beschreibung
"FileToSystem"	Die CSV-Datei wird mit den gebundenen CoDaBix® Nodes synchronisiert.
"SystemToFile"	Die gebundenen CoDaBix® Nodes werden mit der CSV-Datei synchronisiert. In diesem Fall können Sie über ein oder mehrere Trigger-Elemente Trigger definieren, die angeben, wann die Daten in CoDaBix gesammelt und in die CSV-Datei geschrieben werden.

CSV Sync Trigger

Die folgenden Werte sind gültig für Attribute dieses Typs:

Wert	Beschreibung
"FileChanged"	Der Synchronisationsprozess soll angestoßen werden, sobald die Datei verändert wird. Das bedeutet, dass die Synchronisation jedes Mal dann aufgerufen wird, wenn das Datum sich ändert, an dem die Datei zuletzt geschrieben wurde.

CSV Sync Action

Die folgenden Werte sind gültig für Attribute dieses Typs:

Wert	Beschreibung
"KeepFile"	Die Datei soll vor/nach der Synchronisierung behalten werden.
"TruncateFile"	Der Inhalt der Datei soll vor/nach der Synchronisierung entfernt werden.
"DeleteFile"	Die Datei soll vor/nach der Synchronisierung gelöscht werden.

Wert	Beschreibung
"MoveFile"	Die Datei soll vor/nach der Synchronisierung verschoben oder umbenannt werden. Für diese Aktion muss das BeforeSyncMoveFileTo- bzw. AfterSyncMoveFileTo-Attribut angegeben werden (siehe oben).

Das Server-Element kann wie folgt aussehen:

```
<Server SyncMode="FileToSystem"
  SyncTrigger="FileChanged"
  AfterSyncAction="DeleteFile">
  <!-- 0-n Trigger elements -->
  <!-- File element -->
</Server>
```

Trigger-Element

Das Trigger-Element dient (falls SyncMode="SystemToFile") zur Definition eines Triggers, der angibt, wann die Daten von CoDaBix® über einen synchronen Lesevorgang gesammelt und dann in die CSV-Datei geschrieben werden sollen, und stellt folgende Attribute zur Verfügung:

	Verpflichtend	Typ	Zweck
Type	ja	String	Gibt den Typ des Triggers an. "ValueChange": Der Trigger löst aus, wenn ein neuer Wert in einen bestimmten Node geschrieben wird und der neue Wert sich vom bisherigen Wert des Nodes unterscheidet. "Edge": Der Trigger löst aus, wenn ein bestimmter Wert (angegeben durch das EdgeValue-Attribut) in einen bestimmten Node geschrieben wurde (während der vorherige Nodewert ein anderer war). Optional kann ein Wert festgelegt werden, der nach dem synchronen Lesevorgang in den Node zurückgeschrieben werden soll. "Interval": Der Trigger löst in einem regelmäßigen Intervall aus, das durch das interval-Attribut spezifiziert wird.
Node (falls Type="Edge" oder Type="ValueChange")	ja	String	Gibt den vollqualifizierten Pfad zur Node an, die überwacht werden soll.
EdgeValue (falls Type="Edge")	ja	String	Der Wert, auf den der Trigger überprüft. Der Trigger löst aus, wenn dieser Wert auf einen Node geschrieben wird und der bisherige Nodewert ein anderer war.
ChangeBackValue (falls Type="Edge")	nein	String	Falls angegeben, schreibt das CSV Exchange Plugin diesen Wert in den Node, nachdem der Trigger ausgelöst hat und der Datensatz über einen synchronen Lesevorgang eingesammelt wurde.

	Verpflichtend	Typ	Zweck
ChangeBackNode (falls Type="Edge")	nein	String	Falls angegeben und falls ChangeBackValue angegeben ist, schreibt das CSV Exchange Plugin den Wert in den Node, der durch dieses Attribut angegeben ist, anstatt in den Node, der durch das Node-Attribut angegeben ist.
Interval (falls Type="Interval")	ja	Int32	Gibt das Trigger-Intervall in Millisekunden an. Beispielsweise bedeutet ein Wert von 2000, dass der Trigger alle zwei Sekunden auslöst.

Das Trigger-Element kann wie folgt aussehen:

```
<Trigger Type="Edge" Node="/Nodes/Line1/TriggerNode" EdgeValue="1" ChangeBackValue="0" />
```

File-Element

Das File-Element dient als Container für das Element Bindings und definiert die Attribute, um die für den Austausch benötigte Datei aufzusetzen. Diese Attribute konfigurieren die Datei und ihr zum Synchronisieren benutztes Format.

Das File-Element stellt folgende Liste an Attributen zur Verfügung:

	Verpflichtend	Typ	Zweck
Path	ja	Node Query Expression	<p>Der vollqualifizierte zugreifbare Pfad zur zu synchronisierenden CSV-Datei.</p> <p>Bei Verwendung des SyncMode SystemToFile werden die in der Node Query Expression angegebenen Nodes in den synchronen Lesevorgang der Column-Nodes mit eingeschlossen, und dann beim Schreiben der Datei ausgewertet.</p> <p>Bei Verwendung des SyncMode FileToSystem werden die angegebenen Nodes beim Starten des Channels einmalig ausgewertet, und zusätzlich können Sie einen Filter wie XY*.csv verwenden, wodurch das Plugin alle Dateien, die dem Filter entsprechen, überwacht und verarbeitet.</p> <p>Beachten Sie: Für Dateien, auf die über das lokale Dateisystem zugegriffen wird, unterliegt dieser Pfad den Beschränkungen der File Access Security, die in den CoDaBix® Einstellungen definiert wurden.</p>
Separator	nein	Char	Der CSV Separator in der Datei, der zum Trennen der Werte voneinander benutzt wird (Standard: ;).

	Verpflichtend	Typ	Zweck
HasHeader	nein	Bool	Gibt an, ob die erste Zeile in der CSV-Datei Spaltenüberschriften enthält. Gültige Werte: True, False Bei Angabe von True bedeutet dies für den SyncMode FileToSystem, dass die erste Zeile der CSV-Datei bei der Synchronisierung übersprungen wird. Für den SyncMode SystemToFile bedeutet dies, dass bei der Synchronisierung bei einer leeren CSV-Datei zuerst eine Header-Zeile mit einer Überschrift für jede Spalte geschrieben wird, die in den <Binding>-Elementen über das HeaderName-Attribut festgelegt werden kann.
Type	nein	String	Legt den Typ des Dateiübertragungsmechanismus fest. File (Standard): Auf die Datei wird über das lokale Dateisystem zugegriffen. Scp: Auf die Datei wird über einen SSH-Server (SCP-Protokoll) zugegriffen. Sftp: Auf die Datei wird über einen SSH-Server (SCP-Protokoll) zugegriffen.
Zusätzliche Attribute, wenn Type="Scp" oder Type="Sftp" :			
Hostname	ja	String	Der Hostname des SSH-Servers.
Username	ja	String	Der Benutzername für den SSH-Server.
Password	ja	String	Das verschlüsselte Passwort für den SSH-Server. Um das verschlüsselte Passwort zu erhalten, öffnen Sie bitte CoDaBix und klicken in der Webkonfiguration auf  Password Security . Dort können Sie das Originalpasswort eingeben und dieses verschlüsseln, sodass es in dieses Attribut eingefügt werden kann.

Das File-Element kann wie folgt aussehen:

```
<File Path="MachineSetup.csv">
  <!-- Bindings element -->
</File>
```

oder beim Benutzen von SCP:

```
<File Path="/directory/MachineSetup.csv" Type="Scp"
Hostname="192.168.0.20" Username="user1" Password="password1">
  <!-- Bindings element -->
</File>
```

Bindings-Element

Das Bindings-Element agiert als Container für ein oder mehrere Binding-Elemente. Dieses Element verwaltet alle Bindungen (Bindings), die der Datei zugeordnet sind.

Das Bindings-Element stellt folgende Liste an Attributen zur Verfügung:

	Verpflichtend	Typ	Zweck
BaseNode	nein	String	Der Nodepfad, der als Basis für relative Nodepfade der folgenden Node-Bindings verwendet wird.

Das Bindings-Element kann wie folgt aussehen:

```
<Bindings BaseNode="/Nodes/Line 1/Tools/CuttingTool">
  <!-- 0-n Binding elements -->
</Bindings>
```

Binding-Element

Das Binding-Element bindet eine Spalte in der CSV-Datei an einen CoDaBix® Node. Diese Information wird daher vom CSV Exchange Plugin genutzt, um die kommasetrennten Werte (comma-separated values = CSV) mit CoDaBix® Nodes zu verbinden.

Jedes Binding-Element stellt folgende Liste an Attributen zur Verfügung:

	Verpflichtend	Typ	Zweck
ColumnIndex	ja	Int32	Der null-basierte Index (Index der ersten Spalte ist null) der zu bindenden CSV Spalte.
Node	ja (falls SystemValueType nicht angegeben)	String	Der (relative, falls BaseNode benutzt wird) Nodepfad des zu bindenden Nodes.
ValueType (falls Node angegeben)	nein	Value Type	Falls angegeben, erzwingt die Interpretation des Spaltenwerts als angegebener Typ. Sie können die Formatierung über das ValueFormat-Attribut festlegen.
ValueFormat (falls ValueType angegeben)	nein	String	Gibt die Formatierung für den Wert an.
SystemValueType	ja (falls Node nicht angegeben)	System Value Type	Falls angegeben, wird statt des Werts eines Nodes ein Systemwert für diese Spalte benutzt. Siehe die untere Tabelle für die möglichen Systemwert-Typen.
SystemValueFormat (falls SystemValueType angegeben)	nein	String	Gibt die Formatierung des Systemwerts an (z.B. Datumsformat).
HeaderName	nein	Node Query Expression	Wenn HasHeader im <File>-Element auf True gesetzt ist, gibt dies die Überschrift der Spalte an. Falls nicht angegeben, wird ein Standardname wie „Column1“ verwendet.

Value Type

Die folgenden Werte sind gültig für Attribute dieses Typs:

Wert	Beschreibung
"DateTime"	Der Wert der Spalte soll als DateTime-Wert interpretiert werden. Sie können die Formatierung über eine .NET Datums-/Uhrzeit-Formatzeichenfolge (standard oder benutzerdefiniert) festlegen, z.B. yyyy-MM-dd HH:mm:ss.
"TimeSpan"	Der Wert der Spalte soll als TimeSpan-Wert interpretiert werden. Sie können die Formatierung über eine .NET TimeSpan-Formatzeichenfolge (standard oder benutzerdefiniert) festlegen, z.B. dd\ .hh\ :mm\ :ss.

System Value Type

Die folgenden Werte sind gültig für Attribute dieses Typs:

Wert	Beschreibung
"TriggerTimestamp" (falls SyncMode="SystemToFile")	In die Spalte wird der Zeitstempel geschrieben, wann der Trigger ausgelöst hat. Sie können die Formatierung über eine .NET Datums-/Uhrzeit-Formatzeichenfolge (standard oder benutzerdefiniert) festlegen, z.B. yyyy-MM-dd HH:mm:ss.
"CreationTimestamp" (falls SyncMode="FileToSystem")	Der Zeitstempel, der in der Spalte steht, wird als CreationTimestamp für die Nodewerte verwendet, die in CoDaBix geschrieben werden sollen. Sie können die Formatierung über eine .NET Datums-/Uhrzeit-Formatzeichenfolge (standard oder benutzerdefiniert) festlegen, z.B. yyyy-MM-dd HH:mm:ss.

Das Binding-Element kann wie folgt aussehen:

```
<Binding ColumnIndex="0" Node="Depth"/>
<Binding ColumnIndex="1" Node="Last Refresh" ValueType="DateTime"
ValueFormat="yyyyMMdd-HHmss"/>
<Binding ColumnIndex="2" SystemValueType="TriggerTimestamp"
SystemValueFormat="yyyyMMdd-HHmss"/>
```

Node Query Expression

Bestimmte Attribute (z.B. Dateipfad oder Spaltenüberschrift) ermöglichen es, einen Nodepfad anzugeben, was bedeutet, dass dieser Teil durch den Nodewert ersetzt wird. Eine Query Expression hat die Form $\${<Nodepfad>|<ID>|<Guid>}$ (der Nodepfad selbst darf kein }-Zeichen enthalten). Wenn Sie ein Dollarzeichen (\$) direkt verwenden möchten, müssen Sie dieses als zwei Dollarzeichen (\$\$) escapen.

Wenn eine Expression ausgewertet wird, wird ein synchroner Lesevorgang initiiert. Dies ermöglicht Ihnen, den Wert in einem Script, welches einen `codabix.NodeReader` für die entsprechende Node registriert, bereitzustellen.

Beispiele (angenommen Node /Nodes/A hat Wert „First“, Node /Nodes/B hat Wert „Second“):

Node Query Expression	Resultierende Ausgabe/Dateipfad
C:\File-\$/Nodes/A}.csv	C:\File-First.csv
\$\$Header_\$/Nodes/A}\$\$/Nodes/B}\$\$	\$Header_FirstSecond\$

Benutzung

Es wird empfohlen, einen professionellen XML Editor zur manuellen Bearbeitung der Konfigurationsdatei zu benutzen. Um auch von der am Ende dieses Artikels erwähnten XML-Schemadefinition zu profitieren, müssen Sie sich auf das Schema beziehen, indem Sie das `xsi:noNamespaceSchemaLocation`-Attribut wie folgt in der Dokumentenwurzel `PluginSettings` anwenden (die XSD-Datei muss neben der XML-Datei platziert sein):

```
<?xml version="1.0" encoding="utf-8" ?>
<PluginSettings xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Codabix.CsvExchangePlugin.xsd">
  <!-- child elements -->
</PluginSettings>
```

Unabhängig davon, ob Sie die Konfigurationsdatei manuell oder automatisiert bearbeiten/erstellen, muss der oben dokumentierte Elementenbaum erfüllt sein, um eine gültige, wohlgeformte und benutzbare Konfigurationsdatei zu erzeugen.

Synchronisation

Sobald das Plugin vom CoDaBix® Plugin Manager geladen und gestartet wird, wird seine Konfigurationsdatei vom Plugin mit den entsprechenden CoDaBix® Entities gelesen und synchronisiert.

Falls sich die Konfigurationsdatei ändert, informiert der Plugin Manager das Plugin. Durch diese Benachrichtigung startet das Plugin neu und benutzt die aktuellsten Konfigurationsänderungen beim Hochfahren.

Beispiel Konfigurationsdatei

[Codabix.CsvExchangePlugin.Settings.xml](#)

```
<PluginSettings
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Codabix.CsvExchangePlugin.Settings.
xsd">
  <Servers>
    <Server SyncMode="FileToSystem"
      SyncTrigger="FileChanged"
      AfterSyncAction="MoveFile"
      AfterSyncMoveFileTo="ProcessedFiles\ ">
      <File Path="SampleFile*.csv">
        <Bindings BaseNode="/Nodes/Line 1/Tools/CuttingTool">
          <Binding ColumnIndex="0" Node="Depth"/>
          <Binding ColumnIndex="1" Node="Speed"/>
          <Binding ColumnIndex="2" Node="Direction/X"/>
          <Binding ColumnIndex="3" Node="Direction/Y"/>
        </Bindings>
      </File>
    </Server>
  </Servers>
</PluginSettings>
```

```

        <Binding ColumnIndex="4" Node="Date"
        ValueType="DateTime" ValueFormat="yyyyMMdd-HHmss" />
    </Bindings>
</File>
</Server>

<Server SyncMode="SystemToFile">
    <Trigger Type="Edge" Node="/Nodes/Line
2/Feedback/TriggerNode" EdgeValue="1" ChangeBackValue="0" />
    <File Path="/home/user/${/Nodes/Line2/CsvFileName}"
        Type="Sftp"
        Hostname="192.168.0.20"
        Username="user1"
        Password="encrypted-password"
        HasHeader="True">
        <Bindings BaseNode="/Nodes/Line 2/Feedback">
            <Binding ColumnIndex="0" Node="CurrentDepth"
            HeaderName="My Column 1"/>
            <Binding ColumnIndex="1" Node="CurrentSpeed"
            HeaderName="My Column 2"/>
            <Binding ColumnIndex="2"
            SystemValueType="TriggerTimestamp" SystemValueFormat="yyyyMMdd-
            HHmss" HeaderName="My Column 3"/>
        </Bindings>
    </File>
</Server>
</Servers>
</PluginSettings>

```

Beispiel Konfigurationsschema-Datei

Codabix.CsvExchangePlugin.Settings.xsd

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema id="Codabix.CsvExchangePlugin.Settings"
    elementFormDefault="qualified"
    xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:simpleType name="SystemValueTypeType">
        <xs:restriction base="xs:string">
            <xs:enumeration value="CreationTimestamp" />
            <xs:enumeration value="TriggerTimestamp" />
        </xs:restriction>
    </xs:simpleType>

    <xs:simpleType name="ValueTypeType">
        <xs:restriction base="xs:string">
            <xs:enumeration value="Integer" />
            <xs:enumeration value="FloatingPoint" />
            <xs:enumeration value="DateTime" />
        </xs:restriction>
    </xs:simpleType>

```

```
<xs:enumeration value="TimeSpan" />
</xs:restriction>
</xs:simpleType>

<xs:complexType name="BindingType">
  <xs:attribute name="ColumnIndex" use="required">
    <xs:simpleType>
      <xs:restriction base="xs:integer">
        <xs:minInclusive value="0" />
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="HeaderName" use="optional">
    <xs:simpleType>
      <xs:restriction base="xs:string" />
    </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="Node" use="optional">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:minLength value="1" />
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="SystemValueType"
type="SystemValueTypeType" use="optional" />
  <xs:attribute name="SystemValueFormat" type="xs:string"
use="optional" />
  <xs:attribute name="ValueType" type="ValueTypeType"
use="optional"/>
  <xs:attribute name="ValueFormat" type="xs:string"
use="optional" />
</xs:complexType>

<xs:complexType name="BindingsType">
  <xs:sequence>
    <xs:element name="Binding" type="BindingType" minOccurs="1"
maxOccurs="unbounded" />
  </xs:sequence>

  <xs:attribute name="BaseNode" type="xs:string" use="optional"
/>
</xs:complexType>

<xs:simpleType name="FileTypeType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="File" />
    <xs:enumeration value="Scp" />
    <xs:enumeration value="Sftp" />
    <xs:enumeration value="file" />
    <xs:enumeration value="scp" />
  </xs:restriction>
</xs:simpleType>
```

```
    <xs:enumeration value="sftp" />
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="HasHeaderType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="True" />
    <xs:enumeration value="False" />
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="FileType">
  <xs:sequence>
    <xs:element name="Bindings" type="BindingsType"
minOccurs="1" maxOccurs="1" />
  </xs:sequence>

  <xs:attribute name="Path" type="xs:string" use="required" />
  <xs:attribute name="Type" type="FileTypeType" use="optional"
/>
  <xs:attribute name="Hostname" type="xs:string" use="optional"
/>
  <xs:attribute name="Username" type="xs:string" use="optional"
/>
  <xs:attribute name="Password" type="xs:string" use="optional"
/>
  <xs:attribute name="MonitoringInterval" type="xs:integer"
use="optional" />

  <xs:attribute name="HasHeader" type="HasHeaderType"
use="optional" />

  <xs:attribute name="Separator" use="optional" default=";">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:length value="1" />
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
</xs:complexType>

<xs:simpleType name="TriggerTypeType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Edge" />
    <xs:enumeration value="ValueChange" />
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="TriggerType">
  <xs:attribute name="Type" type="TriggerTypeType"
use="required" />
```

```
<xs:attribute name="Node" type="xs:string" use="optional" />
<xs:attribute name="EdgeValue" type="xs:string" use="optional"
/>
<xs:attribute name="ChangeBackValue" type="xs:string"
use="optional" />
<xs:attribute name="ChangeBackNode" type="xs:string"
use="optional" />
</xs:complexType>

<xs:simpleType name="SyncModeType">
<xs:restriction base="xs:string">
<xs:enumeration value="FileToSystem" />
<xs:enumeration value="SystemToFile" />
</xs:restriction>
</xs:simpleType>

<xs:simpleType name="SyncTriggerType">
<xs:restriction base="xs:string">
<xs:enumeration value="FileChanged" />
</xs:restriction>
</xs:simpleType>

<xs:simpleType name="SyncActionType">
<xs:restriction base="xs:string">
<xs:enumeration value="KeepFile" />
<xs:enumeration value="TruncateFile" />
<xs:enumeration value="DeleteFile" />
<xs:enumeration value="MoveFile" />
</xs:restriction>
</xs:simpleType>

<xs:complexType name="ServerType" mixed="true">
<xs:sequence>
<xs:element name="Trigger" type="TriggerType" minOccurs="0"
maxOccurs="unbounded" />
<xs:element name="File" type="FileType" minOccurs="1"
maxOccurs="1" />
</xs:sequence>

<xs:attribute name="IsEnabled" type="xs:boolean"
use="optional" />
<xs:attribute name="SyncMode" type="SyncModeType"
use="optional" />
<xs:attribute name="SyncTrigger" type="SyncTriggerType"
use="optional" />
<xs:attribute name="BeforeSyncAction" type="SyncActionType"
use="optional" />
<xs:attribute name="AfterSyncAction" type="SyncActionType"
use="optional" />
<xs:attribute name="BeforeSyncMoveFileTo" type="xs:string"
use="optional" />
```

```
<xs:attribute name="AfterSyncMoveFileTo" type="xs:string"
use="optional" />
</xs:complexType>

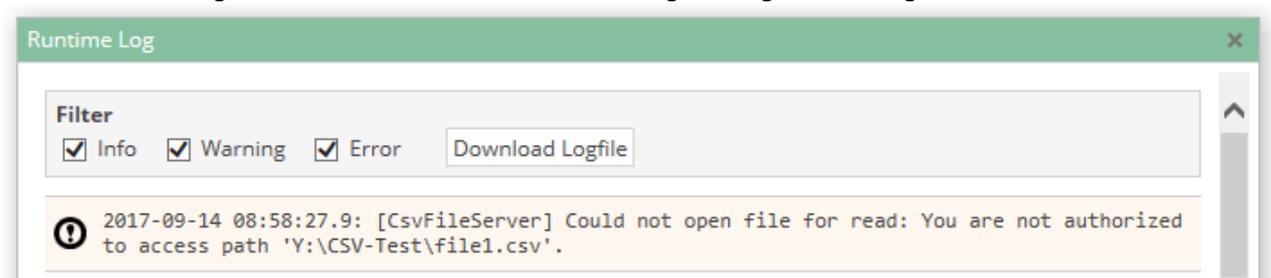
<xs:complexType name="ServersType">
  <xs:sequence>
    <xs:element name="Server" type="ServerType" minOccurs="0"
maxOccurs="unbounded" />
  </xs:sequence>
</xs:complexType>

<xs:complexType name="PluginSettingsType">
  <xs:sequence>
    <xs:element name="Servers" type="ServersType" minOccurs="1"
maxOccurs="1" />
  </xs:sequence>
</xs:complexType>

<xs:element name="PluginSettings" type="PluginSettingsType" />
</xs:schema>
```

Fehlerdiagnose

Das CSV Exchange Plugin loggt Ereignisse ins CoDaBix® Runtime Log, die mit [CsvFileServerPlugin] oder [CsvFileServer] anfangen. Sie können das Runtime Log in der Webkonfiguration öffnen, um die letzten Logeinträge anzuzeigen:



Entities

Das CSV Exchange Plugin verwendet das CoDaBix® Entity Model nicht, da es über eine XML-Konfigurationsdatei (Codabix.CsvExchangePlugin.Settings.xml) konfiguriert wird und daher keine Entities zur Verfügung stellt.

Ordner & Dateien

Ordner

Name	Pfad	Zweck/Anwendung
AssemblyFolder	<CodabixInstallDir>/plugins/CsvExchangePlugin/	Beinhaltet die Plugin Assemblydatei.
ConfigFolder	<CodabixDataDir>/plugins/CsvExchangePlugin/	Beinhaltet die Plugin Konfigurationsdatei.
LoggingFolder	<CodabixDataDir>/log/	Beinhaltet die Plugin Logdateien.

Dateien

Typ	Pfad	Zweck/Anwendung
Assembly	[AssemblyFolder]/Codabix.CsvExchangePlugin.dll	Die Plugin Assemblydatei.
Config File	[ConfigFolder]/Codabix.CsvExchangePlugin.Settings.xml	Die Konfigurationsdatei.

Versionsinformation

Dieses Dokument

Datum	2019-04-17
Version	1.4

Plugin

Name	CSV File Server
Node	/System/Plugins/Exchange/CSV
Version	1.1.0

Assembly

Name	Codabix.CsvExchangePlugin.dll
Datum	2019-04-17
Version	1.1.0.0

From:

<https://codabix.de/> - **CoDaBix**®

Permanent link:

<https://codabix.de/de/plugins/exchange/csvexchangeplugin>

Last update: **2021/07/30 13:39**