

Inhaltsverzeichnis

v1.2	1
v1.1	3
v1.0	7
0.23	15
0.22	16
0.21	18
0.20	19
0.19	22
0.18	24
0.17	29
0.16	33
0.15	35
0.14	36
0.12	39
0.11	41
0.10	42
0.9	45

Versionshistorie

Dieses Dokument listet die Änderungen in den freigegebenen Versionen von Codabix auf.

Eine Liste bekannter Probleme in der aktuellen Version finden Sie unter [Bekannte Probleme](#).

v1.2

v1.2.0 (freigegeben am 29.09.2021) (Änderungen verglichen mit **v1.1.6**)

- **Inkompatible Änderung: [Windows]** Codabix benötigt nun **Windows 8** bzw. **Windows Server 2012** oder höher.
- **Inkompatible Änderung: [Linux]** Unter Debian (oder Derivaten wie Raspberry Pi OS) benötigt Codabix nun **Debian 10** oder höher, und unter Fedora benötigt Codabix nun **Fedora 33** oder höher.
- Die **.NET-Runtime** wurde auf Version **5.0.10** upgedatet.
- **[Windows]** Ein neuer Build **Codabix für Windows (ARM64)** ist nun verfügbar, welcher auf ARM-basierten PCs, wie dem Surface Pro X oder Windows VMs auf einem Apple-Silicon-basierten Mac, läuft.
Beachten Sie: In diesem Build sind derzeit das **Melsec QJ Device Plugin** und das **H1 Device Plugin** nicht enthalten.
- **Codabix Host:**
 - Unterstützung für die NO_COLOR-Umgebungsvariable wurde zur Codabix-Shell-Anwendung hinzugefügt, sodass virtuelle Terminalsequenzen („ANSI Color“) deaktiviert werden, wenn diese Variable vorhanden ist.
- **Codabix Webkonfiguration:**
 - Die Script-Editor-Umgebung wurde auf Monaco Editor v0.25.2 und TypeScript v4.3 upgedatet.
- **Codabix Engine:**
 - Eine neue Option **History Value Mode** can in den Projekteinstellungen gesetzt werden, die es ermöglicht, eine separate Tabelle pro Node für historische Werte (anstatt einer einzigen Tabelle) für verbesserte Performance beim Lesen von historischen Werten zu verwenden.
- **Allen-Bradley Device Plugin:**
 - Der Treiber wurde auf **libplctag v2.3.8** upgedatet.
- **Beckhoff Device Plugin:**
 - Das Beckhoff Device Plugin wurde hinzugefügt, mit dem TwinCAT-Maschinen (unter Verwendung von TwinCAT 3.1.4024 oder höher) angebunden werden können.
- **EUROMAP Device Plugin:**
 - Unterstützung für Events (Alarms und Changes) wurde hinzugefügt.
 - Unterstützung für den Dateizugriff über FTP wurde hinzugefügt.
 - Das Verhalten beim Akzeptieren einer Response wurde verbessert, wenn die

- Command Response Mode-Einstellung auf MayLinger gesetzt ist.
- Das Logging auf Debug-Ebene wurde für Diagnosezwecke verbessert.
- **Modbus Device Plugin:**
 - Die Endianness (Bytereihenfolge) für numerische Werte kann nun konfiguriert werden.
- **OPC UA Client Device Plugin:**
 - **Inkompatible Änderung:** Verbindungen zu **OPC Classic** Servern sind aus Sicherheitsgründen **standardmäßig gesperrt**. Um diese zu erlauben, aktivieren Sie die Option „Allow context-sensitive credentials“ in den Codabix Access Security Einstellungen.
 - Unterstützung für Event-Subscriptions wurde über den Alarms + Events-Folder hinzugefügt.
 - Eine neue Sektion **Classic** wurde zu den Settings hinzugefügt, die es ermöglicht, OPC-Classic-Verbindungen genauer zu konfigurieren.
 - Beim Lesen von Variant-Nodes (mit einer Datatype ID von `i=24`) werden nun eingebaute strukturierte Datentypen wie `DataValue`, `LocalizedText`, `Range` vollständig als JSON serialisiert.
 - Bei Method-Nodes können Sie nun bei deren OUT-Argument-Nodes den Variant-Datentyp (`i=24`) im Path angeben (JSON-Format), damit eingebaute strukturierte Datentypen wie `DataValue`, `LocalizedText`, `Range` als JSON serialisiert werden.
- **RFC-1006 Device Plugin:**
 - Es wurde ein Problem behoben, durch das eine falsche TPDU-Size beim Verbindungshandshake im Server-Modus gesendet wurde.
- **Serial Port Device Plugin:**
 - Das Serial Port Device Plugin wurde hinzugefügt, mit dem Sie eine Verbindung zu seriellen Geräten, wie beispielsweise Barcode-Scanner, herstellen können.
- **SQL Exchange Plugin:**
 - Unterstützung für Stored Procedures und Funktionen bei Verwendung von **MySQL/MariaDB** wurde hinzugefügt.
 - Zusätzlich zu den Nodes im Folder `Columns` gibt es nun einen Folder `ColumnsSet`, der ebenfalls `Column-NOdes` nach dem Browsen der Datenbank/Tabelle enthält, jedoch mit Array-Typen. Dies ermöglicht Ihnen, ein komplettes Result-Set (d.h. mehrere Zeilen) aus der Tabelle zu lesen, wobei bei jedem Node das Array dann die Werte der einzelnen Zeilen für diese Spalte enthält, die gelesen wurden.
 - Columns mit nicht unterstützten Werttypen können nun als Stringwert gelesen werden.
 - Es wurde ein Problem behoben, das verhinderte, dass eine `CHAR(36)`-Spalte beim Verwenden einer MySQL- oder MariaDB-Datenbank als String gelesen werden konnte.
- **OPC UA Server Interface Plugin:**
 - Es wurde ein Problem behoben, das dazu führte, dass beim Aufruf der `Read`-Methode auf File-Nodes ein falscher Werttyp (Byte-Array statt `ByteString`) zurückgegeben wurde.
- **REST Interface Plugin:**
 - Authentifizierungstoken verwenden nun einen projektspezifischen Key, anstatt den des aktuellen Prozesses.

- **SignalR Interace Plugin:**
 - Der HTTP-Header Access-Control-Allow-Credentials wird aus Sicherheitsgründen nicht mehr gesendet, da er für diese Anwendung nicht benötigt wird. Bei browserbasierten SignalR-Clients können Sie "withCredentials": false in dem von withUrl() verwendeten Objekt angeben.
 - Authentifizierungstoken verwenden nun einen projektspezifischen Key, anstatt den des aktuellen Prozesses.
- **Script Interface:**
 - Die Methode net.ping.sendAsync wurde hinzugefügt, die es ermöglicht, einen ICMP Echo Request zu senden.
 - Die Methode net.registerHttpRequestRoute() hat nun einen zusätzlichen options-Parameter, mit dem Benutzerauthentifizierung ermöglicht bzw. erzwungen werden kann. Wenn aktiviert, kann der Client ein Token über das REST Interface v2 erhalten und dieses in einem Authorization: Bearer <token> Header senden. Informationen über den authentifizierten User können dann über die net.HttpContext.user-Property abgerufen werden.
 - Die Eigenschaft expectContinue wurde zu net.httpClient.HttpRequestMessage hinzugefügt, mit der festgelegt werden kann, dass ein Expect: 100-continue-Header gesendet werden soll, um auf eine Response vom Server zu warten, bevor der Request-Body gesendet wird.
 - **[Windows]** Die Genauigkeit von Timeouts, die in timer.delayAsync angegeben, wurde bei niedrigen Werten verbessert.

v1.2.1 (freigegeben am 14.10.2021)

- Die **.NET-Runtime** wurde auf Version **5.0.11** upgedatet (enthält **Sicherheitsverbesserungen**).
- **Codabix Host:**
 - **[Windows]** Der eingebettete Webbrowser in der Codabix-Anwendung wurde auf Chromium v94.0.4606.71 upgedatet (enthält **Sicherheitsverbesserungen**).
- **Socket Device Plugin:**
 - Es wurde ein Synchronisierungsproblem behoben, das dazu führen konnte, dass ein Paket, das in die Transmit-Buffer-Node geschrieben wurde, in der nächsten Verbindung anstatt in der aktuellen Verbindung gesendet wurde.
- **REST Interface Plugin/SignalR Interace Plugin:**
 - Es wurde ein Problem behoben, das beim Lesen von historischen Werten zu einem Hängenbleiben führen konnte.

v1.1

v1.1.0 (freigegeben am 07.05.2021) (Änderungen verglichen mit v1.0.10)

- **Codabix Host:**
 - Eine Vorschauversion der neuen **Codabix Dashboards**-Anwendung wurde hinzugefügt und ist unter der URL /dashboards/ abrufbar.
- **Codabix Engine:**
 - **Inkompatible Änderung:** Der Name der Settings-Dateien, die von Plugins (CSV Exchange Plugin, Database Plugin, S7 Device Plugin) genutzt werden, wurde von

- CoDaBix.*.Settings.xml in Codabix.*.Settings.xml geändert. Wenn Sie Codabix von einer früheren Version upgraden, versucht Codabix automatisch, die Dateien umzubenennen; Sie sollten jedoch prüfen, ob Sie die früheren Dateinamen in anderen Anwendungen benutzen.
- **Inkompatible Änderung:** Die Precision-Eigenschaft bei Nodes wurde durch die Format-Eigenschaft ersetzt, die eine .NET-Formatzeichenfolge bei numerischen und 'DateTime/TimeSpan-Werttypen enthalten kann. Beim Upgrade von einer früheren Version werden die vorhandenen Precision-Werte automatisch in entsprechende Format-Werte konvertiert.
 - Ein neues Feld „**Login Name**“ wurde zu User-Objekten hinzugefügt, welches zum Einloggen (alternativ zu E-Mail und Phone Number) verwendet werden kann, beispielsweise in der REST API und in OPC UA Clients.
 - Ein neuer, unveränderlicher **Anonymous User** (ID -1) wurde hinzugefügt, der verwendet werden kann, um den anonymen Benutzerzugriff bei Plugins zu konfigurieren, die die anonyme Benutzerauthentifizierung unterstützen.
 - Die Wertkonvertierung wurde verbessert, sodass ein Wert beim Lesen aus einem Gerät vor dem Aufruf von Value Converters nicht in den Node-Werttypen konvertiert wird, damit ein Value Converter Zugriff auf den ursprünglichen Wert bekommt.
- **OPC UA Server Interface Plugin:**
 - **Anonyme Benutzerauthentifizierung** kann nun verwendet werden, indem der **Anonymous User** einer oder mehreren Benutzergruppen hinzugefügt wird.
 - Neue Transport-Settings **MaxMessageQueueSize** und **MaxNotificationsPerPublish** wurden hinzugefügt (die auch höhere Standardwerte verwenden), damit der Server eine größere Zahl an Notifications puffern kann.
 - **OPC UA Client Device Plugin:**
 - Die Unterstützung für OPC Classic wurde für bessere Stabilität in eine separate Komponente (**OPC UA Gateway**) refaktoriert.
Beachten Sie: Für die OPC-UA-Gateway-Komponente muss das .NET Framework 4.7.2 oder höher installiert sein.
 - **S7 Device Plugin:**
 - Es ist nun möglich, einen Kanal aus einer **S7 OPC UA & Mobile HMI**-Datenbank zu importieren (entweder in Codabix über den **Import**-Kontextmenüeintrag des Channels, oder im S7 Configurator).
 - **RFC-1006 Device Plugin:**
 - Die Implementierung wurde für bessere Stabilität und Effizienz überarbeitet.
 - Zusätzliche Werte werden nun in die ReceiveBuffer-Node bei Verbindungsereignissen geschrieben: Ein Null-Wert mit Status Good sobald die Verbindung hergestellt wurde, und ein Null-Wert mit Status Bad sobald die Verbindung geschlossen wurde.
 - TSAP-Strings verwenden nun **ISO-8859-1** statt **Windows-1252** als Kodierung (aber Stringwerte in den ReceiveBuffer- und TransmitBuffer-Nodes verwenden weiterhin UTF-8).
 - Wenn ein leerer String für einen der TSAP-Werte angegeben wird, wird dieser nicht RFC-1006 CR (Connection Request)-Frame angegeben.
 - Die maximale Größe einer Nachricht (**TSDU**) beim Empfangen ist derzeit auf 20 MiB limitiert.

- **EUROMAP Device Plugin:**
 - Eine neue Einstellung `ReadValueMode` wurde hinzugefügt, die angibt, wie der `REPORT`-Befehl zum Lesen von Werten generiert wird.
 - Eine neue Einstellung `OperationTimeout` wurde hinzugefügt, die angibt, wie lange auf Lese-/Schreiboperationen gewartet wird, bevor diese abgebrochen werden.
- **CSV Exchange Plugin:**
 - Die Konfiguration ist nun über den Node-Tree möglich anstatt über eine XML Settings-Datei.
Eine bereits vorhandene XML Settings-Datei wird (beim Starten des Plugins, oder wenn die Datei während des Betriebs erstellt wird) automatisch in die Node-Tree-Konfiguration migriert und in den „backup“-Ordner kopiert. Anschließend wird die Datei gelöscht, um zu vermeiden, dass die Konfiguration im Node-Tree beim erneuten Start des Plugins überschrieben wird.
- **SQL Exchange Plugin:**
 - Die Performance beim Ausführen eines `SELECT`-Kommandos zum Lesen von Column-Nodes wurde durch die Verwendung einer `LIMIT 1`-Klausel (für MySQL/PostgreSQL/SQLite) bzw. `TOP (1)`-Klausel (für MSSQL) verbessert, da nur die erste zurückgegebene Zeile tatsächlich verwendet wird.
- **REST Interface Plugin:**
 - Das neue REST Interface v2 wurde hinzugefügt, das unter der URL `/api/rest` erreichbar ist. Es bietet eine RESTful API, um auf eine Codabix-Instanz zuzugreifen und diese zu managen.
- **SignalR Interace Plugin:**
 - Das SignalR Interface Plugin wurde hinzugefügt, welches eine Echtzeit-Web-API bietet, um unter Verwendung des Microsoft SignalR Core Frameworks auf eine Codabix-Instanz zuzugreifen und diese zu managen.
- **SINUMERIK Device Plugin [UNSTABLE]:**
 - Eine experimentelle Version des SINUMERIK Device Plugins wurde hinzugefügt, mit dem eine Verbindung zu Sinumerik Powerline und Solution Line-Maschinen hergestellt werden kann.
Beachten Sie: Verwenden Sie dieses Plugin **nicht** in Produktionsumgebungen, da es möglicherweise nicht richtig arbeitet und sogar einen Crash der Anwendung auslösen könnte.

v1.1.1 (freigegeben am 02.06.2021)

- Die **.NET-Runtime** wurde auf Version **5.0.6** upgedatet.
- **Codabix Host:**
 - **[Windows]** Der eingebettete Webbrowser in der Codabix-Anwendung wurde auf Chromium v90.0.4430.212 upgedatet (enthält **Sicherheitsverbesserungen**).
- **RFC-1006 Device Plugin:**
 - **[Linux]** Es wurde ein Problem behoben, durch das das Verbinden zu einem Server über einen Hostnamen (anstatt einer IP-Adresse) fehlschlagen konnte.
- **S7 Device Plugin:**
 - Es wurde ein Problem behoben, das zu einer einige Sekunden andauernden hohen

CPU-Auslastung führen konnte, wenn die SPS die Verbindung schließt.

- **SQL Exchange Plugin:**

- **MySQL**-Verbindungen setzen nun den SQL-Modus `NO_BACKSLASH_ESCAPES`, damit Backslashes innerhalb eines String-Literals nicht als Escape-Zeichen interpretiert werden, wodurch das Verhalten konsistent mit anderen Datenbanken wie PostgreSQL, SQLite und MSSQL wird.
Dies vereinfacht das Handling für Scripts, die die `ReadSqlExpression` oder `WriteSqlExpression` setzen und dabei ein String-Literal aus benutzerdefinierten Daten angeben müssen.

- **SignalR Interace Plugin:**

- Es wurde ein Problem behoben, das dazu führen konnte, das Codabix nicht mehr reagierte.

- **OPC UA Server Interface Plugin:**

- Ein `ByteString` wird nun immer korrekt in einen Codabix Blob-Wert konvertiert, und umgekehrt.
- Das `ArrayDimensions`-Attribut wird nun korrekt bei `Variable-Nodes` angegeben, die einen Arraywert repräsentieren.

- **OPC UA Client Device Plugin:**

- Ein Codabix Blob-Wert wird nun immer korrekt in einen `ByteString` konvertiert, und umgekehrt.

v1.1.2 (freigegeben am 11.06.2021)

- Die **.NET-Runtime** wurde auf Version **5.0.7** upgedatet (enthält **Sicherheitsverbesserungen**).

- **Codabix Engine:**

- Es wurde ein **Sicherheitsproblem** behoben, das dazu führte, dass Node Tokens (die beim REST Interface v1 für die Authentifizierung verwendet werden) nicht mit dem Password Key des Projekts verschlüsselt wurden.
- Es wurde ein Problem (bestehend seit v1.1.1) behoben, dass zu Fehlern bei der Backend-Datenbank führen konnte, wenn eine MySQL-Datenbank verwendet wird.

- **Codabix Host:**

- **[Windows]** Der eingebettete Webbrowser in der Codabix-Anwendung wurde auf Chromium v91.0.4472.77 upgedatet (enthält **Sicherheitsverbesserungen**).

- **S7 Device Plugin:**

- Es wurde ein Problem behoben, das dazu führte, dass das Standard-Encoding des Betriebssystems (z.B. UTF-8 unter Linux, Windows-1252 unter westeuropäischen Windows-Systemen) anstatt ISO-8859-1 als String-Encoding beim Lesen von Werten (jedoch nicht beim Schreiben) verwendet wurde.

- **OPC UA Client Device Plugin:**

- Es wurde ein Problem behoben, das zu einem Crash führen konnte, wenn eine `Method-Node` mit nicht-unterstützten Parametertypen aufgerufen wurde.

v1.1.3 (freigegeben am 26.07.2021)

- **Codabix Host:**

- **[Windows]** Der eingebettete Webbrowser in der Codabix-Anwendung wurde auf

Chromium v92.0.4515.107 upgedatet (enthält **Sicherheitsverbesserungen**).

v1.1.4 (freigegeben am 23.08.2021)

- Die **.NET-Runtime** wurde auf Version **5.0.9** upgedatet (enthält **Sicherheitsverbesserungen**).
- **Codabix Host:**
 - **[Windows]** Der eingebettete Webbrowser in der Codabix-Anwendung wurde auf Chromium v92.0.4515.159 upgedatet (enthält **Sicherheitsverbesserungen**).

v1.1.5 (freigegeben am 20.09.2021)

- **Codabix Host:**
 - **[Windows]** Der eingebettete Webbrowser in der Codabix-Anwendung wurde auf Chromium v93.0.4577.82 upgedatet (enthält **Sicherheitsverbesserungen**).

v1.1.6 (freigegeben am 29.09.2021)

- **[Linux]** Es wurde ein Problem behoben, durch das das Erstellen des Machinencodes mit dem Fehler „No valid disk could be found“ fehlschlagen konnte.
- **Codabix Host:**
 - **[Windows]** Der eingebettete Webbrowser in der Codabix-Anwendung wurde auf Chromium v94.0.4606.61 upgedatet (enthält **Sicherheitsverbesserungen**).

v1.1.7 (freigegeben am 14.10.2021)

- Die **.NET-Runtime** wurde auf Version **5.0.11** upgedatet (enthält **Sicherheitsverbesserungen**).
- **Codabix Host:**
 - **[Windows]** Der eingebettete Webbrowser in der Codabix-Anwendung wurde auf Chromium v94.0.4606.71 upgedatet (enthält **Sicherheitsverbesserungen**).

v1.0

v1.0.0 (freigegeben am 11.12.2020)

- **⚠ Inkompatible Änderung:** Nach dem Upgrade auf diese Version muss ein **neuer Lizenzschlüssel** eingegeben werden. Ohne die Eingabe eines neuen Lizenzschlüssels läuft Codabix wieder im Evaluationsmodus für 28 Tage (Windows) bzw. für drei Stunden (Linux/Raspberry Pi).
- **Wichtige Änderung: [Windows]** Unter **Windows 7 SP1** und **Windows Server 2008 R2 SP1** wird das Update **KB3063858** (x64 oder x86) benötigt. Außerdem muss, um Lizenzen mit einem Maschinencode verwenden zu können, das Windows Management Framework 3.0 (**KB2506143**) installiert sein.
- **Inkompatible Änderung: [Linux]** Unter Ubuntu benötigt Codabix nun **Ubuntu 18.04** oder höher, und unter Fedora benötigt Codabix nun **Fedora 32** oder höher.
- **Inkompatible Änderung:** Wenn MySQL als Backend-Datenbank verwendet wird, benötigt Codabix nun **MySQL v8.0** oder höher.
- **[Linux]** Ein neuer Build **Codabix für Linux (ARM64)** ist nun verfügbar, welcher z.B. auf einem Raspberry Pi 4 mit einem 64-Bit Raspberry Pi OS ausgeführt werden kann.
- **[Linux]** Der Build **Codabix für Raspberry Pi** wurde umbenannt in **Codabix für Linux**

(ARM32), um dem Schema der anderen Builds zu entsprechen.

- Die Schreibweise des Produkts wurde von **CoDaBix** in **Codabix** geändert.
- Das Codabix-Produktmodell wird nun in der About-Seite in der Codabix Webkonfiguration sowie in den Fenstertiteln angezeigt.
- Die **.NET-Runtime** wurde auf Version **5.0.1** für bessere Performance und Flexibilität für alle Codabix-Releases (inklusive Windows) upgedatet.
- **Codabix Host:**
 - **Inkompatible Änderung:** Der Name der ausführbaren Dateien wurde geändert, um sie auf der Kommandozeile leichter benutzen zu können:
 - **Codabix Shell**-Anwendung: Von `CoDaBix-Shell[.exe]` auf `codabix[.exe]`
 - **Codabix**-Anwendung (nur für Windows): Von `CoDaBix.exe` auf `codabix-ui.exe`
 - **Inkompatible Änderung:** Das Format der Kommandozeilen-Argumente in den **Codabix**- und **Codabix Shell**-Anwendungen wurde geändert, um der gängigen Praxis zu entsprechen. Sie können `codabix --help` ausführen, um eine Auflistung der möglichen Kommandos und Argumente zu erhalten. Beispielsweise wurde das Argument zum Verwenden eines festen Projektverzeichnisses von `--projectDirectory:<path>` auf `--project-directory=<path>` geändert.
 - **Inkompatible Änderung:** Das Kommandozeilen-Argument `--remoteHttp` wurde in der **Codabix Shell**-Anwendung entfernt. Stattdessen können Sie nun Remote HTTP-Bindings (inklusive `https`) über die Projekteinstellungen konfigurieren.
 - **Inkompatible Änderung: [Windows]** Der Web Server Mode Windows HTTP Server API (früher: `HTTP.sys`) wird nur noch unter Windows 10 Version 1607/Windows Server 2016 oder höher unterstützt. Unter älteren Windows-Versionen können Sie stattdessen `Embedded (Kestrel)` verwenden.
 - **Inkompatible Änderung: [Linux]** Bei der Verwendung eines Projekts, bei dem der Web Server Mode auf Windows HTTP Server API gesetzt ist (welcher nur unter Windows unterstützt wird), verweigert Codabix nun den Start, anstatt stillschweigend zu Kestrel zu wechseln. Sie können den Modus in den Projekteinstellungen in der **Codabix Shell**-Anwendung ändern.
 - **[Windows]** Der eingebettete Webserver (Kestrel) wird nun standardmäßig als Web Server Mode verwendet anstatt der Windows HTTP Server API, da Kestrel generell die beste Performance bietet.
 - **[Windows]** Der eingebettete Webbrowser in der Codabix-Anwendung wurde auf Chromium v87.0.4280.88 mit Fixes für **Sicherheitslücken** upgedatet.
 - **[Linux]** Die Webserver-Einstellung *Use Local HTTP Port* ist nun standardmäßig aktiviert, um sicherzustellen, dass Benutzer die Codabix Webkonfiguration von anderen Maschinen aus erreichen können.
 - **[Linux]** Wenn der Dienst über einen Prozess installiert wird, der als **root** ausgeführt wird (z.B. als Teil eines größeren Installationsroutine), versucht Codabix nun über die `SUDO_*`-Umgebungsvariablen die ursprüngliche Benutzer-ID zu ermitteln, um sicherzustellen, dass Dateien, die durch den Dienst erstellt werden, den korrekten Besitzer aufweisen.
 - Beim Erstellen eines neuen `https`-Bindings in den Projekteinstellungen wird nun automatisch ein selbstsigniertes SSL-Zertifikat für Testzwecke generiert.
 - Der obsoletere HTTP-Header `X-Frame-Options`, der bisher vom Webserver gesendet wurde, wurde durch einen `Content-Security-Policy`-Header mit einer `frame-ancestors`-Direktive ersetzt, die nur Seiten vom gleichen Ursprung

(Origin) erlaubt, Inhalte in Frames einzubetten.

Falls nötig, kann der Header zum Hinzufügen von zusätzlichen Quellen angepasst werden, indem in der `project.cbxproj`-Settingsdatei String-Einträge zum `ContentSecurityPolicyFrameAncestorSources`-Array hinzugefügt werden.

- Beim Wiederherstellen eines Backups unter Verwendung eines externen Datenbankservers als Backend-Datenbank (wie MySQL, MariaDB oder Microsoft SQL Server) werden nun statt des Löschens und Neuerstellens der ganzen Datenbank nur deren Inhalte (wie Tabellen oder Sequenzen) neu erstellt. Dies stellt sicher, dass die der verwendeten Datenbank zugewiesenen Benutzerrechte beim Wiederherstellen eines Backups nicht verloren gehen.
- Es wurden neue Kommandozeilenargumente zur Codabix-Shell-Anwendung hinzugefügt, mit denen Lizenzinformationen angezeigt und gesetzt werden können.
- Codabix muss nach dem Eingeben eines Lizenzcodes nicht länger neu gestartet werden.

• Codabix Webkonfiguration:

- **Inkompatible Änderung:** Internet Explorer 11 und ältere (EdgeHTML-basierte) Microsoft-Edge-Versionen werden nicht mehr unterstützt. Falls Sie zuvor einen dieser Browser verwendet haben, um die Codabix Webkonfiguration zu öffnen, installieren Sie bitte einen modernen Browser wie [Microsoft Edge](#), [Google Chrome](#) oder [Mozilla Firefox](#).
- Folder-Nodes haben nun einen neuen Kontextmenüeintrag „**Node Explorer**“, um ein separates Node-Explorer-Fenster zu öffnen, mit dem per **Drag'n'Drop** Links erstellt werden können.
- Der **Node-Tree** aktualisiert sich nun automatisch bei serverseitigen Änderungen. Durch diese Änderung wurden auch einige Bugs bezüglich dem Verschieben von Nodes im Tree behoben.
- Es ist nun möglich, in der Codabix Webkonfiguration **Backups zu erstellen und wiederherzustellen**, sogar dann, wenn Codabix als Dienst läuft.
- Der aktuelle **Lizenzstatus** wird nun angezeigt, falls eine Evaluierungslizenz benutzt wird oder Probleme mit der aktuellen Lizenz bestehen. Zusätzlich können nun erweiterte Lizenzinformationen über die „About“-Seite abgerufen werden.
- Das Styling wurde verbessert für ein besseres Look-and-Feel.
- Hotkeys wurden in der „Nodes“-Ansicht hinzugefügt, um den Node-Explorer nahezu komplett steuerbar über die Tastatur zu machen (die Hotkeys werden in den Tooltips der Toolbar-Buttons angezeigt).
- Die Toolbar hat ein neues Layout.
- Beim Anzeigen von Werten einer DateTime-Node (in lokaler Zeit) wird nun die Zeitzoneinformation mit ausgegeben, damit der Zeitpunkt eindeutig identifiziert werden kann.
- Die Script-Editor-Umgebung wurde auf Monaco Editor v0.21.2 und TypeScript v4.1 upgedatet.

• Codabix Engine:

- Unterstützung für strukturierte Datentypen wurde hinzugefügt.
- Die Control-Nodes `Action` und `StartMode` wurden zu einem einzigen Node **IsActive** vereint. Dies stellt sicher, dass nach einem Neustart von Codabix ein Modul bzw. Channel genau dann gestartet wird, wenn er beim Herunterfahren von Codabix gestartet war.
- Das Node-Ereignishandling wurde verbessert, um Ereignisse innerhalb von Node-Transaktionen auszulösen, anstatt nachdem die Transaktion committed wurde.

- Dies verbessert ebenfalls das Handling beim Duplizieren von Nodes in der Codabix Webkonfiguration sowie beim Importieren von Nodes über das Extension XML Plugin, sodass importierte/duplizierte Channel-Settings nicht länger vom Entity Model mit Standardwerten überschrieben werden.
- Das Node-Command-Handling wurde verbessert, um das **asynchrone Ausführen** von Commands zu erlauben. Dies stellt sicher, dass es während der Ausführung von langdauernden Methoden (z.B. Browsen eines OPC UA Client Device-Kanals) möglich ist, gleichzeitig andere Methoden (z.B. Browsen anderer Kanäle) aufzurufen sowie andere Kanäle zu starten und zu stoppen.
 - Beim Schreiben von Werten in Nodes, für die kein NodeWriter registriert ist, wird nun ein Good-Status als Ergebnis des Schreibvorgangs zurückgegeben, anstatt des Status des zu schreibenden Wertes. Dies ist darin begründet, da ein zu schreibender Wert den Status Bad haben kann, das Schreiben dieses Wertes in die Node jedoch tatsächlich funktioniert hat, und der zurückgegebene Status angeben soll, ob das Schreiben funktioniert hat.
 - Bei der Verwendung eines Projektverzeichnisses mit einer älteren Backend-Datenbank-Version führt Codabix nicht länger ein automatisches Upgrade durch, sondern fordert eine **Bestätigung des Upgrades**. Damit wird ein versehentliches Upgrade des aktuellen Projekts vermieden, da ein Projekt nach dem Upgrade nicht mehr mit älteren Codabix-Versionen verwendet werden kann.
 - Die maximale Länge des Nodenamens wurde von 100 auf 200 erhöht.
 - **[Windows]** Es wurde ein Problem behoben, das dazu führte, dass die eingebettete Datenbank (SQLite) nicht funktioniert, wenn als Projektverzeichnis ein **UNC-Pfad** verwendet wird.
- **S7 Device Plugin:**
 - **Inkompatible Änderung:** Der Device-Node wurde von S7 TCP-IP Device in S7 Device umbenannt. Falls Sie den Nodepfad /System/Devices/S7 TCP-IP Device z.B. in einem Script verwenden, müssen Sie diesen auf /System/Devices/S7 Device ändern.
 - **Inkompatible Änderung:** Die angegebenen **Rack- und Slot-Einstellungen** werden nun immer verwendet. Falls die Verbindung zu einer S7 nun mit dem Fehler The specified CPU could not be found fehlschlägt, müssen Sie möglicherweise die Rack-/Slot-Einstellungen auf die korrekten Werte anpassen.
 - Unterstützung für 64-Bit Typen LInt (Int64), LWord (UInt64) und LReal (Double) wurde hinzugefügt.
 - Unterstützung für den Import von S7 File Logger Projektdateien (. fgg) wurde hinzugefügt.
 - Einträge von mehreren Schreibaufträgen werden nun, falls möglich, in einen einzigen Schreibauftrag kombiniert (unter Beibehaltung der einzelnen Werte), um die Netzwerklatenz zu reduzieren und die Performance zu verbessern.
 - **Allen-Bradley Device Plugin:**
 - Der Treiber wurde auf **libplctag v2.1.17** upgedatet, welcher die Konnektivität zu SPSen verbessert, die über eine DH+-Brücke verbunden sind.
 - Unterstützung für Bit-Zugriff (auf Integer- oder BOOL-Typen) wurde hinzugefügt, zusammen mit einer neuen Pfad-Syntax, die es ermöglicht, den Bit-Bereich anzugeben.
 - **EUROMAP Device Plugin:**
 - Das EUROMAP Device Plugins wurde hinzugefügt, mit dem Maschinen angebunden werden können, die die EUROMAP-63-Spezifikation implementieren (dateibasierter

Zugriff).

Es unterstützt die folgenden Features:

- Browsen der Machine (durch Ausführen eines GETID-Befehls) zum automatischen Anlegen der verfügbaren Parameter
- Lesen von Werten durch einen manuellen Lesevorgang oder über eine Subscription (durch Generierung eines REPORT-Kommandos, das einen einzelnen Sample aufzeichnet)
- Schreiben von Werten durch Generierung von SET-Kommandos)
- Starten und Abbrechen von benutzerdefinierten Report-Definitionen (allerdings werden die aufgezeichneten Werte noch nicht wieder zurück in die Nodes geschrieben).

- **MQTT Device Plugin** [UNSTABLE]:

- Eine experimentelle Version des MQTT Device Plugins wurde hinzugefügt, mit dem eine Verbindung zu MQTT-Brokern hergestellt werden kann. Das Plugin kann optional unter Windows installiert werden.

Beachten Sie: Verwenden Sie dieses Plugin **nicht** in Produktionsumgebungen, da es möglicherweise nicht richtig arbeitet und sogar einen Crash der Anwendung auslösen könnte.

- **CSV Exchange Plugin:**

- **Inkompatible Änderung:** Der Name der Settings-Datei wurde von `CoDaBix.CsvFileServer.Settings.xml` in `CoDaBix.CsvExchangePlugin.Settings.xml` geändert. Wenn Sie Codabix von einer früheren Version upgraden, versucht Codabix automatisch, die Datei umzubenennen; Sie sollten jedoch prüfen, ob Sie den früheren Dateinamen in anderen Anwendungen benutzen.
- Es ist nicht mehr nötig, das `ColumnIndex`-Attribut für ein `<Binding>`-Element anzugeben, wenn der Spaltenindex gleich dem Elementindex ist.

- **SQL Exchange Plugin:**

- **Inkompatible Änderung:** ODBC-Verbindungen sind aus Sicherheitsgründen **standardmäßig gesperrt**. Um diese zu erlauben, aktivieren Sie die Option „Allow context-sensitive credentials“ in den Codabix Access Security Einstellungen.
- Unterstützung für **PostgreSQL** wurde hinzugefügt.
- **[Windows]** Unterstützung für die Verwendung von Windows-Authentifizierung beim Verbinden zu MSSQL- oder PostgreSQL-Server wurde hinzugefügt.

Beachten Sie: Die Verwendung von Windows-Authentifizierung ist aus Sicherheitsgründen **standardmäßig gesperrt**. Um diese zu erlauben, aktivieren Sie die Option „Allow context-sensitive credentials“ in den Codabix Access Security Einstellungen.

Beachten Sie: Wenn Codabix als Dienst installiert wird, läuft es standardmäßig unter einem anderen Benutzerkonto (`LocalSystem`) als der Benutzer, der den Dienst installiert hat. Damit die Windows-Authentifizierung funktioniert, müssen Sie das Benutzerkonto für den Dienst in der **Dienste**-Anwendung umstellen.

- **OPC UA Server Interface Plugin:**

- **Inkompatible Änderung:** Der Interface-Node wurde von `OPC-UA Server Interface` in `OPC UA Server Interface` umbenannt. Falls Sie den Nodepfad `/System/Interfaces/OPC-UA Server Interface` z.B. in einem Script verwenden, müssen Sie diesen auf `/System/Interfaces/OPC UA Server Interface` ändern.

- Die Einstellung `TransportProtocol` wurde hinzugefügt, welche es ermöglicht, HTTPS anstatt OPC UA TCP (`opc.tcp`) als Transportprotokoll zu verwenden.
 - Unterstützung für das Übergeben von Argumenten an Method-Nodes wurde hinzugefügt.
 - Das Anwendungszertifikat wird nun automatisch neu generiert, wenn der Domainname des aktuellen Zertifikats nicht mit dem Hostname des OPC UA Servers übereinstimmt.
- **OPC UA Client Device Plugin:**
 - **Inkompatible Änderung:** Der Device-Node wurde von OPC-UA Client Device in OPC UA Client Device umbenannt. Falls Sie den Nodepfad `/System/Devices/OPC-UA Client Device` z.B. in einem Script verwenden, müssen Sie diesen auf `/System/Devices/OPC UA Client Device` ändern.
 - Method-Nodes können nun gebrowst und aufgerufen werden.
 - Beim Browsen eines OPC UA-Servers werden nun Nodes mit einem Variant-Datentyp als String-Nodes anstatt als Null-Nodes in Codabix erstellt, um das Lesen von Werten zu ermöglichen.
 - **REST/JSON Interface Plugin:**
 - Die "ct"-Property (Category) bei Nodewerten wurde entfernt.
 - **Script Interface:**
 - **Inkompatible Änderung:** Der Parameter `doNotCreateSubscription` der Methode `codabix.Node.addValueChangedEventListener` wurde invertiert und in `createSubscription` umbenannt. Falls Sie für diesen Parameter zuvor `true` angegeben haben, um keine Subscription zu erstellen, sollten die das Argument auf `false` ändern.
 - **Inkompatible Änderung:** Ein neuer Parameter (`fields`) wurde im `codabix.Type`-Konstruktor nach `typeCode` hinzugefügt.
 - **Inkompatible Änderung:** Der Typ der Properties `codabix.Node.path` und `codabix.NodeStructure.path` wurde von `string | null` in `NodePathType` (entspricht `NodeValueType`) geändert, da es nun möglich ist, beliebige Typen, nicht nur String-Typen, als Path Type zu verwenden.
 - Die Methode `codabix.Node.registerCommand` wurde hinzugefügt, die es ermöglicht, ein Node Command (Callback) für Nodes vom Typ `Method` zu registrieren, inklusive Unterstützung für Argumente, die beim Aufruf der Methode übergeben werden.
Dies ermöglicht Ihnen, Handler für **Method-Nodes** zu implementieren, die z.B. durch einen OPC UA Client, über die Codabix Webkonfiguration oder durch ein anderes Script aufgerufen werden können.
 - Die Methode `net.httpClient.connectWebSocketAsync` wurde hinzugefügt, mit der eine Verbindung zu einem WebSocket-Server hergestellt werden kann.
 - Die Klassen `codabix.TypeMember` und `codabix.TypeField` wurden hinzugefügt.
 - Die Methode `codabix.Type.getFields`, die `codabix.TypeField`-Instanzen zurückgibt, wenn der Type den Object-Typecode hat, wurde hinzugefügt.
 - Die Klasse `codabix.Object`, die strukturierte Daten repräsentiert und in Nodes mit einem Objekt-Werttypen abgelegt werden kann, wurde hinzugefügt.
 - Der Rückgabotyp der Methode `codabix.NodeReader.readValuesAsync` wurde von `Promise<(NodeValue | null)[] | null | undefined>` in `Promise<void>` geändert, da die zurückgegebenen Werte seit Version 0.18 nicht verwendet werden. Existierende Scripts, die den vorherigen Rückgabotyp

- verwenden, funktionieren nach dem Update auf diese Version weiterhin.
- Es ist nun möglich, in einer HTTP-Handler-Methode, die mit `net.registerHttpRequest` registriert wurde, eine Ausnahme zu werfen, ohne dass dies dazu führt, dass das Script gestoppt wird.
- Die ES2015-Methode `Object.setPrototypeOf` wird nun unterstützt.
- Die Performance von Promise-basierten Methoden wurde verbessert.

v1.0.1 (freigegeben am 14.12.2020)

- **Codabix Webkonfiguration:**

- Es wurde ein Problem behoben, das dazu führen konnte, dass der „Login“-Button nicht anklickbar war.

- **Codabix Engine:**

- Es wurde ein Problem (bestehend seit v1.0.0) behoben, das dazu führen konnte, dass zwei gleiche Nodewerte als ungleich erkannt wurden.
Wenn Sie beispielsweise `e.isValueChanged` in einem `ValueChanged`-Eventlistener in einem Script verwendet haben, konnte dies `true` zurückgeben, selbst wenn der Wert gleich dem vorherigen Wert war.

v1.0.2 (freigegeben am 17.12.2020)

- **Codabix Webkonfiguration:**

- Es wurde ein Problem behoben, das dazu führen konnte, dass das Menü nicht richtig funktioniert, wenn das **Script Interface Plugin** nicht installiert oder nicht lizenziert ist.

- **Codabix Engine:**

- Es wurde ein Problem (bestehend seit v1.0.0) behoben, das dazu führen konnte, dass in Device-Variables, für die eine Subscription erstellt wird, kein Bad-Status geschrieben wird, wenn der Channel derzeit gestoppt ist.

v1.0.3 (freigegeben am 13.01.2021)

- Die **.NET-Runtime** wurde auf Version **5.0.2** upgedatet (enthält **Sicherheitsverbesserungen**).

- **Codabix Engine:**

- Das Verhalten von Scheduled Actions (Callbacks, die aufgerufen werden sollen, sobald der aktuelle Eventhandler verlassen wird) wurde verbessert, sodass eine solche Action immer ausgeführt wird, bevor die Methode, die das Event ausgelöst hat, zurückkehrt, auch dann wenn derzeit bereits auf einer niedrigeren Ebene des Aufrufstapels Scheduled Actions ausgeführt werden.
- Die `PropertyChanged`- und `ChildrenChanged`-Events werden nun korrekt beim Verlinken eines Nodes (sowie beim Entfernen der Verlinkung) ausgelöst.

- **Codabix Host:**

- **[Windows]** Der eingebettete Webbrowser in der Codabix-Anwendung wurde auf Chromium v87.0.4280.141 upgedatet (enthält **Sicherheitsverbesserungen**).

v1.0.4 (freigegeben am 05.02.2021)

- Es wird eine bessere Fehlermeldung angezeigt, wenn eine Lizenz nur in containerisierten Umgebungen verwendet werden kann.

- **Codabix Host:**

- **[Windows]** Der eingebettete Webbrowser in der Codabix-Anwendung wurde auf Chromium v88.0.4324.146 upgedatet (enthält **Sicherheitsverbesserungen**).

v1.0.5 (freigegeben am 10.02.2021)

- Die **.NET-Runtime** wurde auf Version **5.0.3** upgedatet (enthält **Sicherheitsverbesserungen**).

- **Codabix Host:**

- **[Windows]** Der eingebettete Webbrowser in der Codabix-Anwendung wurde auf Chromium v88.0.4324.150 upgedatet (enthält **Sicherheitsverbesserungen**).

- **OPC UA Server Interface Plugin:**

- Neue Transport-Settings **MaxMessageQueueSize** und **MaxNotificationsPerPublish** wurden hinzugefügt (die auch höhere Standardwerte verwenden), damit der Server eine größere Zahl an Notifications puffern kann.

v1.0.6 (freigegeben am 02.03.2021)

- Es ist nun möglich, subscriptionbasierte (nicht hardwaregebundene) Lizenzen auch in nicht-containerisierten Umgebungen zu verwenden.

- **Codabix Host:**

- **[Windows]** Der eingebettete Webbrowser in der Codabix-Anwendung wurde auf Chromium v88.0.4324.182 upgedatet (enthält **Sicherheitsverbesserungen**).

v1.0.7 (freigegeben am 10.03.2021)

- Die **.NET-Runtime** wurde auf Version **5.0.4** upgedatet (enthält **Sicherheitsverbesserungen**).

- **Codabix Host:**

- **[Windows]** Der eingebettete Webbrowser in der Codabix-Anwendung wurde auf Chromium v89.0.4389.72 upgedatet (enthält **Sicherheitsverbesserungen**).

v1.0.8 (freigegeben am 12.03.2021)

- Die **.NET-Runtime** wurde zurück auf Version **5.0.3** gesetzt aufgrund einer Regression im PropertyGrid, die die Codabix (Windows)-Anwendung betrifft.

v1.0.9 (freigegeben am 09.04.2021)

- Die **.NET-Runtime** wurde auf Version **5.0.5** upgedatet (enthält **Sicherheitsverbesserungen** verglichen mit der zuvor verwendeten v5.0.3).

- **Codabix Host:**

- **[Windows]** Der eingebettete Webbrowser in der Codabix-Anwendung wurde auf Chromium v89.0.4389.114 upgedatet (enthält **Sicherheitsverbesserungen**).

- **Script Interface:**

- Es wurde ein Problem mit der Methode `io.file.readAllTextAsync()` behoben, die fälschlicherweise ein UTF-8 BOM am Anfang einer zu lesenden Datei als ein U+FEFF-Zeichen als Teil des Strings zurückgab.

v1.0.10 (freigegeben am 07.05.2021)

- **Codabix Host:**
 - **[Windows]** Der eingebettete Webbrowser in der Codabix-Anwendung wurde auf Chromium v90.0.4430.85 upgedatet (enthält **Sicherheitsverbesserungen**).
- **S7 Device Plugin:**
 - Unterstützung für den S7 DTL-Datentyp (Variablentyp DateTimeLong) wurde hinzugefügt, der in Codabix in einer Variablen mit dem DateTime-Value-Type verwendet werden kann.
 - Es wurde ein Problem (bestehend seit in v1.0.0) beim Verwenden eines Character-Arrays wie Char[10] in einer String-Variable behoben, das dazu führte, dass nur ein einziges Zeichen gelesen wurde.
- **Script Interface:**
 - Es wurde ein Problem behoben, dass der interne Rekursionszähler nicht ausgelöst wurde, wenn nur native Funktionen (wie `codabix.writeNodeValueAsync()`) auf dem Stapel waren, wodurch es in bestimmten Fällen zu einem Absturz aufgrund eines Stapelüberlaufs kommen konnte.

0.23

0.23.0 (freigegeben am 25.06.2020)

- Das Verhalten der **Backup & Restore**-Funktionalität wurde verbessert, um das Risiko von versehentlichen Datenverlust zu minimieren:
 - Beim Wiederherstellen eines Backups wird nicht mehr das komplette **Projektverzeichnis** gelöscht, sondern es werden nur CoDaBix-definierte Ordner und Dateien gelöscht, bevor die entsprechenden Dateien und Ordner aus dem Backup wiederhergestellt werden. Dies beinhaltet folgende Elemente:
 - Ordner: `dashboard`, `log`, `plugins`, `userdata`, `webfiles`
 - Dateien: `codabixdb.db`, `codabixdb.db-*`, `codabixhistorydb.db`, `codabixhistorydb.db-*`, `project.cbxproj`, `project.cbxkeys`
 - Infolgedessen werden nur noch Dateien, die in diesen Ordnern liegen, ins Backup aufgenommen und später wiederhergestellt. Dateien, die im Stammverzeichnis des Projektverzeichnisses (oder in einem anderen, oben nicht aufgeführten, Unterordner) liegen, werden nicht ins Backup aufgenommen (und werden nicht wiederhergestellt, selbst wenn das Backup mit einer früheren CoDaBix-Version erstellt wurde, die solche Dateien ins Backup aufgenommen hat).
 - Dialoge, die den Benutzer zur Bestätigung der Wiederherstellungsoperation auffordern, wurden verbessert, um expliziter vor einem möglichen Datenverlust zu warnen und genauer darauf hinweisen, welche Ordner in welchem Verzeichnis beim Ausführen der Wiederherstellung gelöscht werden.

0.23.1 (freigegeben am 30.07.2020)

- **[Linux/Raspberry Pi] .NET Core** wurde auf Version **2.1.20** für Sicherheits- und Stabilitätsverbesserungen upgedatet.
- Das Kommandozeilenargument `--namedInstance:<name>` wurde hinzugefügt, mit dem ein Instanzname festgelegt werden kann.
- Es wurde ein Bug behoben, der den Start der Engine verhindern konnte, wenn die Option „Serve Static Web Files“ in den Projekteinstellungen deaktiviert war.
- Das **OPC UA Client Device Plugin** wurde auf Version **1.0.22** aktualisiert:

- Es wurde ein Bug behoben, der das Lesen von Werten verhinderte, die einen `LocalizedText-Array`-Typ verwenden.

0.23.2 (freigegeben am 09.09.2020)

- **.NET Core** und die **ASP.NET Core**-Komponenten wurden auf Version **2.1.22** für Sicherheits- und Stabilitätsverbesserungen upgedatet.

0.23.3 (freigegeben am 19.10.2020)

- Das **OPC UA Client Device Plugin** wurde auf Version **1.0.23** aktualisiert:
 - Es wurde ein **Sicherheitsproblem** behoben, das dazu führen konnte, dass der Client sich zu einem weniger sicheren Endpoint verbindet, als konfiguriert ist.
- Das **Omron Device Plugin** wurde auf Version **1.0.1** aktualisiert:
 - Unterstützung für Host-Link/C-Mode Commands wurde hinzugefügt.

0.23.4 (freigegeben am 30.10.2020)

- **[Windows]** Der eingebettete Webbrowser in der Codabix-Anwendung wurde auf Chromium v85.0.4183.121 mit einem Fix für die **Sicherheitslücke** CVE-2020-15999 upgedatet.

0.23.5 (freigegeben am 13.11.2020)

- **[Windows]** Der eingebettete Webbrowser in der Codabix-Anwendung wurde auf Chromium v86.0.4240.183 mit Fixes für **Sicherheitslücken** (inklusive CVE-2020-16009) upgedatet.

0.23.7 (freigegeben am 10.02.2021)

- **.NET Core** und die **ASP.NET Core**-Komponenten wurden auf Version **2.1.25** für Sicherheits- und Stabilitätsverbesserungen upgedatet.
- Das **OPC UA Client Device Plugin** wurde aktualisiert:
 - Die Browse-Methodnode jedes Channels hat nun zwei zusätzliche IN-Argument-Nodes, mit denen das Browsing-Verhalten kontrolliert werden kann.
- Das **OPC UA Server Interface Plugin** wurde aktualisiert:
 - Neue Transport-Settings **MaxMessageQueueSize** und **MaxNotificationsPerPublish** wurden hinzugefügt (die auch höhere Standardwerte verwenden), damit der Server eine größere Zahl an Notifications puffern kann.

0.22

0.22.0 (freigegeben am 27.04.2020)

- Das **S7 Device Plugin** wurde auf Version **1.1.0** aktualisiert:
 - Es ist nicht länger notwendig, eine S7-XML-Konfigurationsdatei zu verwenden. Stattdessen können Sie nun Channel und Variablen direkt in der CoDaBix Webkonfiguration erstellen und bearbeiten, genau wie bei jedem anderen Device Plugin.
 - Es ist nun möglich, **STEP7-Projekte** über die CoDaBix Webkonfiguration zu importieren (über das Kontextmenü des Channels).

- Es ist nun möglich, einen Channel in der CoDaBix Webkonfiguration aus einer S7-XML-Konfigurationsdatei zu laden und in diese zu speichern (über das Kontextmenü des Channels).
- Es kann weiterhin eine S7-XML-Konfigurationsdatei im S7DevicePlugin-Ordner des Projektverzeichnisses verwendet werden, um Channel automatisch zu synchronisieren, jedoch mit den folgenden Änderungen:
 - Anstatt der Verwendung einer einzigen Settings-Datei für alle Channel (CoDaBix.S7DevicePlugin.Settings.xml) verwendet nun jeder Channel seine eigene Datei mit dem Dateinamensformat CoDaBix.S7DevicePlugin.<ChannelName>.Settings.xml. Beim Upgraden von einer früheren CoDaBix-Version wird die früher verwendete Datei automatisch in separate Dateien pro Channel aufgeteilt.
 - Einem S7-Channel wird **nur dann** eine S7-XML-Konfigurationsdatei zugeordnet, wenn eine Datei mit dem entsprechenden Namen beim Starten des S7-Plugins (d.h. beim Starten von CoDaBix) oder beim Anlegen eines neuen Channels existiert.
 - Zusätzlich zur Synchronisierung der S7-XML-Konfigurationsdatei nach CoDaBix (was bei einer Änderung der S7-XML-Konfigurationsdatei stattfindet) gibt es nun auch die Synchronisierung von CoDaBix in die S7-XML-Konfigurationsdatei (wenn Variablen in CoDaBix geändert werden).
 - Das ChangeType-Attribut wurde entfernt. Um eine Variable aus CoDaBix zu löschen, genügt es, sie einfach in der S7-XML-Konfigurationsdatei zu entfernen; außer beim Starten des S7-Plugins (d.h. beim Starten von CoDaBix), da dann Variablen, die in der S7-XML-Konfigurationsdatei fehlen, zu dieser hinzugefügt werden.
 - Wenn Änderungen von der S7-XML-Konfigurationsdatei nach CoDaBix synchronisiert werden, müssen das S7-Plugin und seine Channel nicht länger stoppen und neustarten (wodurch die Kommunikation unterbrochen würde); stattdessen werden die Änderungen „live“ übernommen (während das S7-Plugin läuft).
- **[Windows] Der S7 Configurator** wurde geändert:
 - Der S7 Configurator wird nicht länger standardmäßig installiert, kann aber durch das Ändern der Features im CoDaBix Setup installiert werden.
 - Anstatt des Editierens der CoDaBix.S7DevicePlugin.Settings.xml-Datei im aktuell eingestellten CoDaBix-Projektverzeichnis können nun manuell S7-XML-Konfigurationsdateien von beliebigen Orten geöffnet und gespeichert werden.
 - Anstatt mehrere Channel auf einmal zu editieren, kann nun ein einziger Channel pro S7-XML-Konfigurationsdatei editiert werden.
- Das **Allen-Bradley Device Plugin** wurde auf Version **1.0.1** aktualisiert:
 - Unterstützung für String-Variablen wurde hinzugefügt.
- Es wurde ein Bug behoben, der es ermöglichte, einen Datapoint Node unterhalb eines **Channels**-Folders anzulegen, der dann nicht mehr gelöscht werden konnte.
- Es wurde ein Bug behoben, der verhinderte, dass eine XML-Datei bei einem Channel-Node importiert werden konnte, die zuvor von einem anderen Channel-Node exportiert wurde.
- **Script API:**
 - Unterstützung für die Verwendung von PATCH als HTTP-Methode bei der Methode `net.httpClient.sendAsync` wurde hinzugefügt.

0.22.3 (freigegeben am 25.05.2020)

- **[Linux/Raspberry Pi] .NET Core** wurde auf Version **2.1.18** für Sicherheits- und Stabilitätsverbesserungen upgedatet.
- Das **S7 Device Plugin** wurde auf Version **1.1.1** aktualisiert:
 - Es wurde ein Bug behoben, der dazu führen konnte, dass das Importieren einiger STEP7-Projekte fehlschlug, oder der zu falschen Symbolnamen führen konnte.

0.22.4 (freigegeben am 25.06.2020)

- **[Linux/Raspberry Pi] .NET Core** wurde auf Version **2.1.19** für Sicherheits- und Stabilitätsverbesserungen upgedatet.
- Das **S7 Device Plugin** wurde auf Version **1.1.2** aktualisiert:
 - Es wurde ein Bug behoben, der dazu führen konnte, dass die Synchronisation eines Channels in eine zugeordnete S7-XML-Datei nicht angestoßen wurde, wenn zwar die Channel Settings, jedoch nicht die Variablen geändert wurden.

0.22.5 (freigegeben am 29.07.2020)

- **[Linux/Raspberry Pi] .NET Core** wurde auf Version **2.1.20** für Sicherheits- und Stabilitätsverbesserungen upgedatet.
- Es wurde ein Bug behoben, der den Start der Engine verhindern konnte, wenn die Option „Serve Static Web Files“ in den Projekteinstellungen deaktiviert war.

0.21

0.21.0 (freigegeben am 18.12.2019)

- Ein neuer Foldernode `Environment` wurde im System-Pfad hinzugefügt, welcher Nodes enthält, die Informationen über die CoDaBix-Umgebung bereitstellen; einschließlich der CoDaBix-Version, des Projektverzeichnis/-namens sowie der aktuellen Systemzeit.
- Es wurden neue Kommandozeilenargumente zur CoDaBix-Shell-Anwendung hinzugefügt, mit denen das Projektverzeichnis initialisiert, die Projekteinstellungen angezeigt und gesetzt, sowie das Admin-Passwort zurückgesetzt werden kann.
- Das OPC UA Trace Log wurde entfernt, da es im laufenden Betrieb zu viele Logeinträge produzieren könnte.
- **[Linux/Raspberry Pi] .NET Core** wurde auf Version **2.1.14** für Stabilitätsverbesserungen upgedatet.
- Das **Allen-Bradley Device Plugin** (Version 1.0.0) wurde hinzugefügt, das es Ihnen ermöglicht, eine Verbindung zu einer Allen-Bradley-SPS (wie ControlLogix, CompactLogix, Micro800, MicroLogix, PLC-5, SLC 500) herzustellen.
- Das **SQL Exchange Plugin** wurde auf Version **1.1.1** aktualisiert:
 - Unterstützung für den Zugriff auf **SQLite**-Datenbanken wurde hinzugefügt (indem der Hostname-Wert als Dateipfad zur Datenbank verwendet wird).

0.21.2 (freigegeben am 15.01.2020)

- **[Linux/Raspberry Pi] .NET Core** wurde auf Version **2.1.15** für Sicherheits- und Stabilitätsverbesserungen upgedatet.
- Das **OPC UA Server Interface Plugin** wurde auf Version **1.0.19** aktualisiert:
 - Beim Generieren eines Anwendungszertifikats wird nun der korrekte Hostname des OPC UA Servers als Domainname verwendet.
 - Eine neue Diagnose-Logoption `LogNodeManagerTrace` wurde hinzugefügt, mit der die Zeit geloggt werden kann, die der Node Manager mit Operationen wie das

Updates von Nodewerten verbringt.

0.21.5 (freigegeben am 25.05.2020)

- **[Linux/Raspberry Pi] .NET Core** wurde auf Version **2.1.18** für Sicherheits- und Stabilitätsverbesserungen upgedatet.
- Das **S7 Device Plugin** wurde auf Version **1.0.15** aktualisiert:
 - Es wurde ein Bug behoben, der dazu führen konnte, dass das Importieren einiger STEP7-Projekte fehlschlug, oder der zu falschen Symbolnamen führen konnte.

0.20

0.20.0 (freigegeben am 22.07.2019)

- **[Linux]** Das **I²C Device Plugin** wurde aus der Linux-Distribution entfernt, da dieses ausschließlich für die Verwendung auf einem Raspberry Pi vorgesehen ist.
- **[Linux/Raspberry Pi]** Das Datei-Locking-Verhalten wurde verbessert.
- Es wurde ein neuer Werttyp **Password Blob** eingefügt (ähnlich zum „Password“-Typ), der verwendet werden kann, um Blobs abzulegen, die mit dem Password Key des Projekts verschlüsselt werden.
- Um die Performance beim Hochfahren zu verbessern, werden beim Starten die zuletzt geschriebenen Historywerte für jede Node nicht länger eingelesen. Dies bedeutet, wenn die `History Options` eines Nodes auf `Yes; only on Value Change` gestellt sind, dass der erste erfasste historische Wert für diesen Node nach dem Starten immer in die Datenbank geschrieben wird.
- Die maximale Arraygröße für Nodewerte wurde auf 100.000 erhöht.
- Es wurde ein Bug (bestehend seit Version 0.19.0) behoben, der dazu führen konnte, dass ein S7 Device Channel einen falschen Status anzeigte, nachdem dieser neugestartet wurde.
- Es wurde ein Bug (bestehend seit Version 0.18.6) behoben, der dazu führen konnte, dass eine Skalierung/Konvertierung inkorrektweise für eine Node angewendet wurde, bei der kein Node Writer registriert war.
- Es wurde ein Bug behoben, der beim Lesen einer Array-Device-Variablen zu einem Absturz von CoDaBix führen konnte, wenn das Array mehr als die erlaubte maximale Anzahl an Arrayelementen enthielt.
- Das **Dashboard** wurde auf Version **0.0.28** aktualisiert:
 - Es wurde ein Bug behoben, der dazu führen konnte, dass die Seite Nodes nicht richtig lud.
- Das **Omron Device Plugin** (Version 1.0.0) wurde hinzugefügt, das es Ihnen ermöglicht, eine Verbindung zu einer Omron-SPS herzustellen.
- Das **XML Exchange Plugin** (Version 1.0.0) wurde hinzugefügt, das es Ihnen ermöglicht, Daten über eine XML-Datei auszutauschen.
- Das **OPC UA Server Interface Plugin** wurde auf Version **1.0.17** aktualisiert:
 - Es wurde eine neuer Setting-Folder `Server Configuration` hinzugefügt, der es ermöglicht, das minimale und maximale Session-Timeout zu setzen.
 - Es wurde eine neuer Setting-Folder `Diagnostics` hinzugefügt, der es ermöglicht, Diagnose-Logging für verschiedene Events in Bezug auf Sessions, Read/Write, Subscriptions und MonitoredItems zu aktivieren und konfigurieren.
 - Beim Erstellen eines neuen Application Certificates wird dieses nun mit dem Password Key des Projekts verschlüsselt.
- Das **OPC UA Client Device Plugin** wurde auf Version **1.0.18** aktualisiert:
 - Beim Erstellen/Festlegen eines neuen Application Certificates bzw. Login

Certificates wird dieses nun mit dem Password Key des Projekts verschlüsselt.

- Das **REST Interface** wurde aktualisiert:
 - Zusätzliche Node-Properties werden nun unterstützt: `unit`, `precision`, `scalingFactor`, `scalingOffset`, `maxValueAge`.
- **Script API**:
 - Die Methode `net.httpClient.sendAsync()` wurde hinzugefügt, mit der Sie HTTP-Requests an externe Server senden können.
 - Die Methoden `createAsync()`, `deleteAsync()`, `moveAsync()` wurden zum `io.directory`-Namespace sowie die Methoden `getDirectoryName()` und `getFileName()` zum `io.path`-Namespace hinzugefügt.

0.20.1 (freigegeben am 24.07.2019)

- Das **OPC UA Server Interface Plugin** wurde auf Version **1.0.18** aktualisiert:
 - Es wurde ein Bug (bestehend seit CoDaBix 0.20.0) behoben, der dazu führen konnte, dass das Entfernen von Monitored Items nicht korrekt behandelt wurde.
- Das **OPC UA Client Device Plugin** wurde auf Version **1.0.19** aktualisiert:
 - Es wurde ein Bug (bestehend seit CoDaBix 0.19.0) behoben, der dazu führte, dass benutzerdefinierte Datentypen beim Browsen eines OPC-UA-Servers nicht korrekt erkannt wurden.

0.20.2 (freigegeben am 19.08.2019)

- **Script Interface Plugin**:
 - Es wurde ein Bug (bestehend seit CoDaBix 0.20.0) behoben, der dazu führte, dass Scripts mit Strictness Level Medium oder High nicht mehr mit **Go Live** gespeichert werden konnten.

0.20.4 (freigegeben am 16.09.2019)

- **[Linux/Raspberry Pi] .NET Core** wurde auf Version **2.1.13** für Sicherheits- und Stabilitätsverbesserungen upgedatet.
- Das **RFC-1006 Device Plugin** wurde auf Version **1.0.3** aktualisiert:
 - Es wurde ein Bug behoben, der beim Stoppen eines Channels zu einem Crash führen konnte.
- Das **OPC UA Client Device Plugin** wurde auf Version **1.0.20** aktualisiert:
 - Es wurde ein Bug behoben, der dazu führen konnte, dass der Wert einer subscribten Node nicht aktualisiert wurde, wenn die Verbindung zum Server abbrach und der Client keine neue Verbindung aufbauen konnte, oder wenn z.B. die NodeID auf dem Server nicht existiert. In diesen Fällen wird nun ein Bad-Wert in die Node geschrieben, und der Client versucht wiederholt, die Subscription wieder aufzubauen.
 - Es wurde ein Bug (bestehend seit CoDaBix 0.20.1) behoben, der dazu führen konnte, dass Foundation-Datentypen beim Browsen eines OPC-UA-Servers nicht korrekt erkannt wurden.

0.20.5 (freigegeben am 25.09.2019)

- Das **CSV Exchange Plugin** wurde auf Version **1.1.1** aktualisiert:
 - Das Trigger-Handling wurde verbessert, sodass ein synchroner Lesevorgang direkt im Eventhandler des Triggers anstatt nach dem Abschluss des vorherigen synchronen Lesevorgangs ausgeführt wird.

Dadurch wird das Set der gesammelten Werte deterministischer, vor allem wenn der Wert eines Edge-Triggers ebenfalls gesammelt wird; außerdem wird sichergestellt, dass sich Einträge in der Trigger-Warteschlange nicht aufstapeln können, wenn ein Trigger in kürzeren Intervallen auslöst als die Zeit, die synchrone Lesevorgänge zur Ausführung benötigen. Dies gleicht das Verhalten auch an das Database Plugin an.

0.20.6 (freigegeben am 16.10.2019)

- **[Linux/Raspberry Pi]** Es wurde ein Bug (bestehend seit Version 0.20.1) behoben, der dazu führte, dass das OPC UA Client Device Plugin und das OPC UA Server Interface Plugin manchmal nicht funktionierten.
- Das **OPC UA Client Device Plugin** wurde auf Version **1.0.21** aktualisiert:
 - Unterstützung für das Lesen und Schreiben von Werten des Typs `LocalizedText` wurde hinzugefügt, indem ein neues JSON-Format für den Path beim Browsen des Servers verwendet wird.

0.20.7 (freigegeben am 21.10.2019)

- Die Performance wurde verbessert, wenn mehrere Leseaufträge für eine Devicevariable zur gleichen Zeit generiert werden.
- Das **S7 Device Plugin** wurde auf Version **1.0.12** aktualisiert:
 - Es wurde ein Bug behoben, der dazuführen konnte, dass Variablen den falschen Wert bekamen, wenn mehrere Variablen mit der selben Adresse zur gleichen Zeit gelesen wurden.

0.20.8 (freigegeben am 24.10.2019)

- Das **S7 Device Plugin** wurde auf Version **1.0.13** aktualisiert:
 - Eine neue Einstellung `DateTimeInterpretation` wurde hinzugefügt, die verwendet werden kann, um die Interpretierung von `DateTime`-Werten zwischen UTC-Zeit und lokaler Zeit umzuschalten.

0.20.9 (freigegeben am 11.11.2019)

- Das **S7 Device Plugin** wurde auf Version **1.0.14** aktualisiert:
 - Es wurde ein Bug behoben, der dazu führen konnte, dass das Importieren einiger STEP7-Projekte fehlschlug.

0.20.12 (freigegeben am 19.11.2019)

- Das **CSV Exchange Plugin** wurde auf Version **1.1.2** aktualisiert:
 - Es wurde ein Bug behoben, der bei SCP-Verbindungen dazu führen konnte, dass nach einiger Zeit keine Daten mehr gelesen wurden.
 - Es wurde ein Bug behoben, der dazu führen konnte, dass Edge- oder `ValueChanged`-Trigger im Modus `SystemToFile` nicht funktionierten.

0.20.13 (freigegeben am 15.01.2020)

- **[Linux/Raspberry Pi] .NET Core** wurde auf Version **2.1.15** für Sicherheits- und Stabilitätsverbesserungen upgedatet.

0.19

0.19.0 (freigegeben am 29.05.2019)

- Der Produktname wurde vereinfacht: **CoDaBix® Compact** heißt jetzt **CoDaBix**.
- Die **CoDaBix Projektverwaltung** (Project Management) wurde eingeführt, womit nun projektspezifische Einstellungen (wie Backend-Datenbank-Zugangsdaten, Loglevel, HTTP-Port usw.) im **Projektverzeichnis** (Project Directory, früher *Datenverzeichnis* (Data Directory)) in der Datei **project.cbxproj** anstatt in einem benutzer-globalen Verzeichnis gespeichert werden.
Dadurch wird es einfacher, zwischen verschiedenen Projekten zu wechseln, da die Einstellungen für jedes Projekt separat beibehalten werden.
 - Eine neue Einstellung **Project Name** wurde hinzugefügt, mit der Sie einen Namen für Ihr Projekt angeben können. Dieser Name wird dann in der Titelleiste der CoDaBix-Anwendung sowie für den Standard-Dateinamen beim Erstellen eines Backups verwendet.
 - In der CoDaBix File Access Security (FAS) wurden die standardmäßigen Einträge für den Level „Normal“ geändert, sodass nicht länger ein Lese-/Schreibzugriff für das komplette Projektverzeichnis enthalten ist, sondern Lese-/Schreibzugriffe für bestimmte Unterordner wie `plugins`, `log`, `userdata`, `webfiles`, `dashboard`.
 - Die Benutzererfahrung, wenn CoDaBix einen installierten Dienst mit einem unterschiedlichen Projektverzeichnis erkennt, wurde verbessert.
 - Beim Upgraden von einer früheren CoDaBix-Version werden die bisherigen CoDaBix Settings zu Projekteinstellungen im aktuell ausgewählten Projektverzeichnis migriert.
 - **[Windows]** Wenn Sie ein Backup in der CoDaBix-Anwendung wiederherstellen, können Sie nun vor der Wiederherstellung das Projektverzeichnis wechseln und die Einstellungen bearbeiten.
- **[Windows]** Der CoDaBix MSI-Installer wurde verbessert, sodass dieser beim Updaten auf eine neuere CoDaBix-Version das Erstellen eines Backups anbietet.
- **[Linux/Raspberry Pi]** Das CoDaBix Setup wurde verbessert, sodass dieses auch unter Linux-Distributionen funktioniert, auf denen `sudo` nicht vorhanden ist.
- Das Verhalten des **History Timers**, der in Version **0.18** eingeführt wurde, wurde überarbeitet:
 - Wenn die **History Options** eines Nodes aktiviert werden, jedoch kein **History Interval** angegeben ist, werden historische Werte beim Schreiben eines neuen Werts in den Node erfasst (wie in früheren Versionen), anstatt dass diese vom History Timer mit einem Intervall von 500 ms behandelt werden. Wenn hingegen ein **History Interval** angegeben ist, werden historische Werte vom History Timer in diesem Intervall erfasst.
 - Die in der CoDaBix Webkonfiguration in der **History Options**-Combobox angezeigten Einträge „on Interval“ und „on Value Change“ wurden für eine bessere Aussagekraft in „Yes“ und „Yes (only on Value Change)“ umbenannt.
- Wenn ein Wert in eine Device-Variable geschrieben wird und das Device den Wert zurück in die Node schreibt (nachdem dieser erfolgreich ins Device geschrieben wurde), wird nun der korrekte Statuscode und Timestamp verwendet, anstatt diejenigen aus dem übergebenen Wert zu verwenden.
- Das Handling von DateTime-Werten in der CoDaBix Webkonfiguration wurde verbessert, sodass Sie Werte im selben Format wie dasjenige, das zum Anzeigen der Werte verwendet wird, eingeben können.
- Beim Auswählen eines Menüeintrags in der CoDaBix Webkonfiguration können Sie diesen

nun in einem separaten Fenster öffnen. Des Weiteren wurde ein neuer Menüeintrag zum Öffnen der CoDaBix Online-Dokumentation hinzugefügt.

- Das **OPC UA Client Device Plugin** wurde auf Version **1.0.17** aktualisiert:
 - **[Windows]** Sie können nun eine Verbindung zu einem OPC Classic Server über das URL-Format `opc.com://<hostname>(:<port>)/<progId>/<classId>` herstellen.
- Das **OPC UA Server Interface Plugin** wurde auf Version **1.0.16** aktualisiert:
 - Requests von OPC-UA-Clients, die Nodes hinzufügen oder löschen würden (welche von CoDaBix nicht behandelt werden) werden nun korrekt verweigert.
- Das **CSV Exchange Plugin** wurde auf Version **1.1.0** aktualisiert:
 - Es ist nun möglich, den Dateinamen aus einem Nodewert zu generieren, durch die Verwendung der Syntax `${Pfad/zum/Node}` (*Node Query Expression*) im Path-Attribut des `<File>`-Elements. Bei Verwendung des SyncMode `SystemToFile` werden die angegebenen Nodes in den synchronen Lesevorgang der Column-Nodes mit eingeschlossen, und dann beim Schreiben der Datei ausgewertet. Bei Verwendung des SyncMode `FileToSystem` werden die angegebenen Nodes beim Starten des Channels einmalig ausgewertet.
 - Bei der Verwendung des SyncMode `FileToSystem` können Sie einen Filter wie z.B. `XY*.csv` im Path-Attribut des `<File>`-Elements angeben, sodass das Plugin alle Dateien, die dem Filter entsprechen, überwacht und verarbeitet.
 - Bei der Verwendung des SyncMode `FileToSystem` können Sie nun ein `<Binding>`-Element mit `SystemValueType="CreationTimestamp"` angeben, um festzulegen, dass diese Spalte einen Timestamp enthält, der als `CreationTimestamp` für die CoDaBix-Nodewerte dieser Zeile verwendet werden soll.
 - Bei der Verwendung des SyncMode `FileToSystem` steht eine neue SyncAction `MoveFile` zur Verfügung, die im `BeforeSyncAction`- bzw. im `AfterSyncAction`-Attribut zusammen mit dem `BeforeSyncMoveFileTo`- bzw. `AfterSyncMoveFileTo`-Attribut angegeben werden kann. Dies ermöglicht Ihnen, die Datei vor oder nach der Synchronisierung in CoDaBix umzubenennen oder zu verschieben. Der Dateipfad kann auch eine *Node Query Expression* enthalten, welche vor dem Verschieben der Datei ausgewertet wird.
 - Bei der Verwendung des SyncMode `SystemToFile` und eines Triggers vom Typ `Edge` können Sie nun den `ChangeBackValue` in einen anderen Node als den durch den Trigger verwendeten Node schreiben lassen, indem Sie das `ChangeBackNode`-Attribut mit dem Nodepfad angeben.
 - Ein neues Attribut `HasHeader` wurde zum `<File>`-Element hinzugefügt, mit dem angegeben werden kann, ob eine CSV-Datei eine Headerzeile enthält. Dies bedeutet, dass für den SyncMode `FileToSystem` die erste Zeile der CSV-Datei ignoriert wird, und dass für den SyncMode `SystemToFile` eine Headerzeile ausgegeben wird, falls die Datei leer ist. Zusätzlich kann für jedes `<Binding>`-Element das `HeaderName`-Attribut mit dem Wert des Headers angegeben werden, der in die CSV-Datei geschrieben werden soll (dieser kann auch eine *Node Query Expression* enthalten, die in den synchronen Lesevorgang der Column-Nodes eingeschlossen wird).
 - Es ist nun möglich, das Format für `DateTime`- und `TimeSpan`-Werte (für beide SyncModes `FileToSystem` und `SystemToFile`) festzulegen durch die Angabe des `ValueType`-Attributs des `<Binding>`-Elements, welches den Typ (e.g. `DateTime`) enthält, und durch die Angabe des `ValueFormat`-Attributs mit der gewünschten Formatzeichenfolge.
 - Es wurde ein Bug behoben, der dazu führen konnte, dass eine CSV-Datei zweimal

- importiert wurde, wenn sie einmal gespeichert wurde.
- Das **SQL Exchange Plugin** wurde auf Version **1.1.0** aktualisiert:
 - Stored Procedures und Funktionen für Oracle-Datenbanken können nun per Method-Nodes angesprochen werden, die z.B. in einem Script per `codabix.executeNodeCommandsAsync()` aufgerufen werden können.
 - Die Einstellung **DateTime Interpretation** wurde hinzugefügt, mit der Sie die Interpretierung von Datums-/Zeitwerten zwischen UTC-Zeit und lokaler Zeit pro Datenbank umschalten können.
 - Beim Browsen eines MSSQL- oder MySQL-Servers wird nun der `time`-Datentyp zum `CoDaBix-TimeSpan`-Werttyp gemappt.
 - Das **H1 Device Plugin** wurde auf Version **1.0.1** aktualisiert:
 - Es wurde ein Bug behoben, der das Plugin daran hindern konnte zu starten, wenn mehrere Netzwerkadapters mit dem gleichen Namen verfügbar waren.
 - Das **Extension XML Plugin** wurde auf Version **1.0.1** aktualisiert:
 - Sie können nun auch Nodewerte exportieren und importieren.
 - **Script API:**
 - Die Methode `runtime.date.format()` wurde hinzugefügt, die zum Formatieren eines Date-Objektes mit einer .NET DateTime-Formatzeichenfolge verwendet werden kann (dies kann eine [standardmäßige](#) or [benutzerdefinierte](#) Formatzeichenfolge sein).
 - Die Methode `codabix.executeNodeCommands()` wurde durch die Methode `codabix.executeNodeCommandsAsync()` ersetzt, welche ein Promise-Objekt zurückgibt, das aufgelöst wird, wenn die Ausführung des Node-Commands abgeschlossen ist.

0.19.1 (freigegeben am 22.07.2019)

- **[Windows]** Die **ASP.NET Core**-Komponenten wurden auf Version **2.1.12** für Sicherheitsverbesserungen (CVE-2019-1075) upgedatet.
- **[Windows]** Es wurde ein Bug behoben, der dazu führen konnte, dass der Start des CoDaBix-Dienstes fehlschlug, wenn das ausgewählte Projektverzeichnis mit einem Backslash (\) endete.
- **[Linux/Raspberry Pi]** **.NET Core** wurde auf Version **2.1.12** für Stabilitätsverbesserungen upgedatet.
- Es wurde ein Bug behoben, der verhinderte, dass der private Schlüssel von SSL-Zertifikaten, die für Kestrel verwendet werden, mit dem Password Key des Projekts verschlüsselt wird.

0.18

0.18.6 (freigegeben am 08.02.2019)

- **[Windows]** Es stehen nun separate .MSI-Installer für **x64** (64-Bit) und **x86** (32-Bit) zur Verfügung.
- **[Windows]** Die Performance des eingebetteten Webbrowsers wurde durch das Wechseln auf einen Chromium-basierten Browser verbessert.
- **[Windows]** CoDaBix unterstützt nun offiziell den Betrieb auf einer **Windows Server Core**-Installation mit installiertem [Server Core App Compatibility FOD](#).
- **[Windows]** Die **ASP.NET Core**-Komponenten wurden auf Version **2.1.7** für Sicherheitsverbesserungen upgedatet.
- **[Raspberry Pi]** **Wichtige Änderung:** CoDaBix benötigt nun Raspbian 9 (Stretch) oder höher.

- **[Raspberry Pi] .NET Core** wurde auf Version **2.1.7** für Stabilitäts- und Sicherheitsverbesserungen upgedatet.
- Unterstützung für **Microsoft SQL Server** 2012 oder höher für die Verwendung als Backend-Datenbank wurde hinzugefügt.
- Das **CoDaBix Dashboard** ist nun in CoDaBix Compact eingebettet und wird vom Webserver über den Pfad /dashboard/ ausgeliefert. Es kann außerdem in der Webkonfiguration angezeigt werden.
- Ein **History-Timer** wurde implementiert, welcher den derzeitigen Wert eines Nodes im angegebenen History-Intervall erfasst (für Nodes, bei denen die History Options aktiviert sind), und dadurch das frühere Verhalten ablöst, bei dem ein historischer Wert immer dann erfasst wurde, wenn ein Wert in die Node geschrieben wurde. Wenn die History Options auf on Value Change eingestellt sind, wird ein erfasster historischer Wert nur dann in die Datenbank geschrieben, wenn dieser sich vom zuletzt geschriebenen historischen Wert dieses Nodes unterscheidet. Das neue Verhalten macht das Erfassen historischer Werte daher deterministischer.
- Eine neue Node-Eigenschaft **History Resolution** wurde hinzugefügt, die, falls angegeben, numerische historische Werte beim Erfassen auf die angegebene Auflösung rundet.
- Historywerte werden in der Webkonfiguration nun nach Einfügereihenfolge (statt nach dem Creation Timestamp) sortiert, um eine bessere Übersicht zu erhalten, und zusätzlich wird nun der Receive Timestamp angezeigt.
- Ein neuer Menüpunkt **Password Security** wurde zur Webkonfiguration hinzugefügt, mit dem Sie Passwörter unter Verwendung des **Password Keys** der Backend-Datenbank verschlüsseln können. Dies ermöglicht Ihnen, verschlüsselte Passwörter z.B. in Plugin-XML-Konfigurationsdateien oder in Scriptcode, wo das Passwort mit `codabix.security.decryptPassword()` entschlüsselt werden kann, abzulegen.
- **Der Password Key** (welcher zum Verschlüsseln von Passwörtern verwendet wird) kann nun in den CoDaBix® Settings zurückgesetzt (neu generiert) werden.
- Wenn der Log Level auf **Debug** gesetzt ist, wird nun das **OPC UA Trace Log** in die Logdatei `OpcUaTraceLog.txt` geschrieben, das Ihnen dabei helfen kann, OPC-UA-Verbindungsprobleme zu diagnostizieren.
- Die Performance beim Einfügen von historischen Werten wurde verbessert.
- Die maximale Größe eines Scripts wurde auf 1.000.000 Zeichen erhöht.
- Die maximale Größe eines String- bzw. Blob-Werts wurde auf 1.000.000 Zeichen bzw. Bytes erhöht.
- Das Handling beim Anwenden einer Skalierung oder von Value Convertern auf einen Node wurde verbessert.
- Die Beschränkung auf die durch die Lizenz vorgegebene maximale Channelanzahl bzw. Clientanzahl für Plugins wurde implementiert.
- Das Lizenz-Handling in Bezug auf den Maschinencode wurde verbessert.
- Es wurde ein Bug (bestehend seit Version **0.17.0**) behoben, der bei der Verwendung eines Scripts, das einen `codabix.NodeReader` für einen Node registriert, zu einem Hängenbleiben von CoDaBix führen konnte.
- Es wurde ein Bug behoben, der das Löschen eines Nodes innerhalb eines Variable-Ordners verhinderte, wenn der Node oder einer seiner Kindnodes einen virtuellen Link enthielt.
- **[Windows] Das H1 Device Plugin** (Version 1.0.0) wurde hinzugefügt, das es Ihnen ermöglicht, eine Verbindung zu SPS-Geräten über das SINEC-H1-Protocoll herzustellen. Beachten Sie: Für dieses Plugin muss WinPcap 4.1.3 oder höher installiert sein.
- Das **Socket Device Plugin** (Version 1.0.0) wurde hinzugefügt, das es Ihnen ermöglicht, eine Verbindung zu einem TCP-Server herzustellen.

- Das **Modbus Device Plugin** (Version 1.0.0) wurde hinzugefügt, das eine Verbindung zu Modbus TCP-Servern ermöglicht.
- Das **Extension XML Plugin** (Version 1.0.0) wurde hinzugefügt, mit dem Sie in der Webkonfiguration über das Kontextmenü Nodestrukturen als XML-Dateien exportieren und importieren können.
- Das **OPC UA Client Device Plugin** wurde auf Version **1.0.15** aktualisiert:
 - Channels, bei denen SubscriptionType auf OpcSubscription festgelegt ist, verwenden nun korrekt eine OPC-basierte Subscription anstatt Polling.
 - Anstatt den kompletten Server zu browsen, können Sie nun auch einzelne Folder-Nodes browsen.
 - Ein neues Setting MaxMonitoredItemsPerSubscription wurde hinzugefügt, das es Ihnen ermöglicht, Monitored Items über mehrere Subscriptions aufzuteilen (z.B. bei einem Siemens SPS OPC-UA-Server, der nur 1000 Elemente pro Subscription unterstützt). Standardmäßig ist der Wert 0 gesetzt, wodurch alle Monitored Items eine einzige Subscription verwenden.
 - Die Performance beim Hinzufügen und Entfernen von CoDaBix-Node-Subscriptions wurde verbessert.
 - Wenn mehr als 1000 Nodes auf einmal vom Server gelesen werden sollen, werden diese nun in Teile zu je bis zu 1000 Nodes aufgeteilt, um sicherzustellen, dass diese korrekt gelesen werden können, wenn der Server die maximale Anzahl an Requestelementen limitiert.
 - Beim Browsen eines Servers werden nun Variable-Nodes mit einem Enumeration-DataType in CoDaBix als Int32-Nodes erstellt.
 - Beim Browsen eines Servers werden nun Variable-Nodes, die als einzige Child-Node „HA Configuration“ enthalten, als Datenpunktnodes statt als Foldernodes erstellt.
- **[Windows]** Das **Melsec QJ Device Plugin** wurde auf Version **1.0.1** aktualisiert:
 - Es wurde ein Bug behoben, der dazu führen konnte, dass mehr Bytes als notwendig gelesen wurden, wenn boolesche Werte gelesen wurden.
- Das **S7 Device Plugin** wurde auf Version **1.0.10** aktualisiert:
 - Variablen vom Typ S5 Time werden nun korrekt gelesen, wenn sie den Wert 0 haben.
 - Es wurde ein Bug behoben, der beim Importieren zu STEP7-Projekten zum Fehlen oder zu falscher Ausrichtung einiger Variablen führen konnte.
- Das **SQL Exchange Plugin** wurde auf Version **1.0.5** aktualisiert:
 - Es wurde ein Bug behoben, der dazu führen konnte, dass ausstehende synchrone Lesevorgänge nach dem Stoppen des Channels einen Wert mit Status Good zurücklieferten, auch wenn der Lesevorgang tatsächlich fehlgeschlagen ist.
 - Die Unterstützung für Microsoft SQL Server wurde verbessert, sodass auch Werte von Datentypen ohne Vorzeichen geschrieben werden können.
 - Die Unterstützung für Oracle wurde verbessert.
- Das **Database Plugin** wurde auf Version **1.0.11** aktualisiert:
 - Sie können nun ein **verschlüsseltes Passwort** anstatt eines Klartext-Passworts im password-Attribut angeben.
 - Die Unterstützung für Microsoft SQL Server wurde verbessert, sodass auch Werte von Datentypen ohne Vorzeichen geschrieben werden können.
 - Der Start des Plugins schlägt nicht länger fehl, wenn ein Node, der für einen edge- oder valueChange-Trigger oder für eine AfterSyncAction angegeben wurde, nicht gefunden werden konnte; stattdessen wird nun eine Warnung geloggt.
- Das **CSV Exchange Plugin** wurde auf Version **1.0.4** aktualisiert:

- Sie können nun ein **verschlüsseltes Passwort** anstatt eines Klartext-Passworts im Password-Attribut angeben.
- Das **REST Interface** wurde aktualisiert:
 - Wenn historische Werte abgerufen werden und dabei alle der from-, to- und count-Parameter angegeben werden, und dabei to kleiner ist als from, werden nun die ältesten Werte (limitiert durch count) anstatt der neusten Werte zurückgegeben.
 - Das browse-Kommando zeigt nun Kind-Nodes an, auf die der aktuelle Benutzer nur indirekt Zugriff hat (wenn er keinen Zugriff auf den Node selbst hat, sondern auf einen seiner Nachfahren).
- **Script-API:**
 - **Inkompatible Änderung:** Die value-Property eines `codabix.NodeValue`, das einen `TimeSpan` enthält, gibt nun die Millisekunden als `number`-Wert anstatt eines formatierten `string`-Werts zurück.
 - **Inkompatible Änderung:** Wenn historische Werte abgerufen werden und dabei alle der `startTime`-, `endTime`- und `maxCount`-Parameter angegeben werden, und dabei `endTime` kleiner ist als `startTime`, werden nun die ältesten Werte (limitiert durch `count`) anstatt der neusten Werte zurückgegeben.
 - **Inkompatible Änderung:** Wenn das `Promise`-Objekt, das durch eine Implementierung der Methode `codabix.NodeReader.readValueAsync()` zurückgegeben wird, mit einem Array aus `codabix.NodeValue` aufgelöst wird, werden diese Werte für das Ergebnis des synchronen Lesevorgangs ignoriert, und stattdessen werden die Werte der zu lesenden Nodes (zu dem Zeitpunkt, an dem das `Promise` aufgelöst wird), verwendet. Dies hat den Grund, dass ansonsten die zurückgegebenen Werte kein Post-Processing (Skalierung, Konvertierung) erhalten würden und deshalb inkorrekt sein könnten.
 - Callbacks, die mit `codabix.Node.registerValueConverter()` oder `codabix.nodeconfiguration.registerNodeContextMenuAction()` registriert werden, können nun eine Ausnahme werfen (oder das zurückgegebene `Promise`-Objekt `rejecten`), ohne dass dies zum Abbruch des Scripts wegen einer unbehandelten Ausnahme führt.
 - Ein optionaler Parameter `throwIfNotFound` wurde zu den Methoden `codabix.findNode()` und `codabix.Node.findNode()` hinzugefügt, mit dem festgelegt werden kann, dass die Methoden eine Ausnahme werfen sollen, falls der Node nicht gefunden werden konnte, anstatt `null` zurückzugeben.
 - Die Eigenschaft `historyResolution` wurde zu den Typen `codabix.Node` und `codabix.NodeStructure` hinzugefügt. Zusätzlich wurden die Eigenschaften `historyOptions` und `historyInterval` zu `codabix.Node` hinzugefügt.
 - Die Eigenschaft `isValueChanged` wurde zum `EventListener`-Argument für das `ValueChanged`-Event des Nodes hinzugefügt, welche angibt, ob sich der neue Wert tatsächlich vom alten Wert unterscheidet.
 - Die statische Methode `codabix.NodeValue.equals()` wurde hinzugefügt, mit der festgestellt werden kann, ob zwei Nodewerte als gleich betrachtet werden können, indem ihr Wert und ihr Status geprüft wird.
 - Die Eigenschaft `receiveTimestamp` wurde zu den von `codabix.readNodeHistoryValuesAsync()` zurückgegebenen Nodewert-Instanzen hinzugefügt, mit der die Zeit abgerufen werden kann, wann dieser historische Wert in CoDaBix erfasst wurde.
 - Der Enum-Wert `IntervalElapsed` von `codabix.NodeHistoryOptions` wurde in `Subscription` umbenannt. Existierende Scripts, die den vorherigen Namen verwenden, funktionieren nach dem Update auf diese Version weiterhin.

- **[Raspberry Pi] Das UniPi Device Plugin** wurde auf Version **1.0.1** aktualisiert:
 - **Inkompatible Änderung:** [Hier finden Sie die Migrationsanleitung.](#)

0.18.7 (freigegeben am 11.02.2019)

- Es wurde ein Bug behoben, der bei einem Upgrade von einer früheren CoDaBix-Version dazu führen konnte, dass Channel-Settings verloren gingen.

0.18.8 (freigegeben am 14.02.2019)

- **[Raspberry Pi] .NET Core** wurde auf Version **2.1.8** für Sicherheits- und Stabilitätsverbesserungen upgedatet.

0.18.9 (freigegeben am 04.03.2019)

- Es wurde ein Bug behoben, der dazu führen konnte, dass unnötige historische Werte für Status-Nodes von nicht geladenen Plugins erzeugt wurden.
- Das **S7 Device Plugin** wurde auf Version **1.0.11** aktualisiert:
 - Es wurde ein Bug (bestehend seit CoDaBix® **0.18.7**) behoben, der dazu führen konnte, dass Änderungen (z.B. Umbenennen von Channels oder Variablen oder Ändern der Variablenadresse) aus der XML-Konfigurationsdatei nicht übernommen wurden, nachdem diese editiert wurde.
 - **[Raspberry Pi]** Es wurde ein Bug behoben, der das Lesen von Gleitkommazahl-Variablen (Single/Double) verhindern konnte.

0.18.10 (freigegeben am 11.03.2019)

- Das Subscription-Handling wurde verbessert, sodass bei einem Device-Channel mit aktiven Subscriptions beim Stoppen ein „Bad“-Status in die subscribten Variablen geschrieben wird, wodurch das Verhalten nun konsistent mit dem synchronen Lesen eines Wertes ist, bei dem ebenfalls ein solcher „Bad“-Status in die Variable geschrieben wird.
- Beim Duplizieren eines Nodes wird nun ebenfalls dessen Wert dupliziert.
- Das **OPC UA Client Device Plugin** wurde auf Version **1.0.16** aktualisiert:
 - Es wurde ein Bug (bestehend seit CoDaBix® **0.18.9**) behoben, der unter speziellen Umständen einen Absturz von CoDaBix verursachen konnte.
- Das **OPC UA Server Interface Plugin** wurde auf Version **1.0.13** aktualisiert:
 - Es wurde ein Bug behoben, der beim Stoppen oder Neustarten des Channels dazu führen konnte, dass Subscriptions von OPC-UA-Clients nicht aus CoDaBix entfernt wurden.

0.18.11 (freigegeben am 18.03.2019)

- **CoDaBix® Compact für Linux (x64)** ist nun verfügbar für Debian 9 oder höher, Ubuntu 16.04 oder höher, Fedora 29 oder höher, OpenSUSE Leap 15.0 oder höher.

0.18.13 (freigegeben am 15.04.2019)

- **[Linux/Raspberry Pi] .NET Core** wurde auf Version **2.1.10** für Stabilitätsverbesserungen upgedatet.
- Es wurde ein Bug (bestehend seit CoDaBix® **0.18.6** (Raspberry Pi), **0.18.10** (Windows) und **0.18.11** (Linux)) behoben, der beim Upgraden einer Backend-Datenbank, die zuletzt

mit CoDaBix® **0.14.0** oder früher verwendet wurde, zu einem Fehler führte.

- Das **OPC UA Server Interface Plugin** wurde auf Version **1.0.15** aktualisiert:
 - Write-Requests, die einen **IndexRange** angeben (welcher in CoDaBix nicht unterstützt wird), werden nun korrekt mit dem Statuscode **BadWriteNotSupported** in Übereinstimmung mit der OPC-UA-Spezifikation abgewiesen.
 - Es wurde ein Bug (bestehend seit CoDaBix® **0.18.10**) behoben, der dazu führen konnte, dass der CoDaBix-Prozess nach dem Herunterfahren der CoDaBix® Engine nicht beendet wurde, falls versucht wurde, einen Channel mit einem bereits belegten Port zu starten.

0.18.14 (freigegeben am 25.04.2019)

- Es wurde ein Bug (bestehend seit CoDaBix® **0.18.10**) behoben, der dazu führen konnte, dass ein ausstehender synchroner Lesevorgang ein falsches Ergebnis zurücklieferte, wenn der entsprechende Channel nach dem Start des Lesevorgangs, jedoch bevor der Lesevorgang fertig war, gestoppt wurde.
- Es wurde ein kleinerer Memory Leak behoben, der bei wiederholtem Erstellen und Löschen von Device-Variablen auftreten konnte.

0.18.15 (freigegeben am 29.05.2019)

- **[Linux/Raspberry Pi] .NET Core** wurde auf Version **2.1.11** für Sicherheits- und Stabilitätsverbesserungen upgedatet.

0.17

0.17.0 (freigegeben am 27.07.2018)

- **CoDaBix® Compact für Raspberry Pi** ist nun verfügbar für Raspbian OS 8.7 oder 9, welches auf einem Raspberry Pi 2 oder 3 ausgeführt wird.
- **[Windows] Wichtige Änderung:** CoDaBix benötigt nun das **.NET Framework 4.7.2** oder höher. Weitere Informationen finden Sie unter **Installation von CoDaBix®**.
- **[Windows]** Kestrel kann nun als alternative Webserver-Implementierung statt der Windows-basierten HTTP.sys verwendet werden. Unter Windows 7 SP1 und Windows Server 2008 R2 SP1 wird Kestrel automatisch verwendet, um sicherzustellen, dass WebSocket-Verbindungen auch auf diesen Betriebssystemen funktionieren.
- **[Raspberry Pi]** Das **I²C Device Plugin** (Version 1.0.0) wurde hinzugefügt, mit dem es möglich ist, auf Geräte über die I²C-Schnittstelle zuzugreifen.
- **[Raspberry Pi]** Das **UniPi Device Plugin** (Version 1.0.0) wurde hinzugefügt, mit dem es möglich ist, auf UniPi-Neuron-Devices zuzugreifen.
- **[Raspberry Pi] Service-Management** wurde zur CoDaBix Shell hinzugefügt, sodass Sie nun CoDaBix Compact als Dienst (wie unter Windows) installieren können, der beim Booten der Maschine automatisch gestartet wird.
- Die Verwendung von **MySQL 8.0** als Backend-Datenbank wird nun unterstützt.
- Die Performance beim Ausführen einer großen Anzahl an Node-Änderungen wurde verbessert (z.B. beim Erstellen vieler Nodes). Wenn SQLite als Backend-Datenbank verwendet wird, schützt dies auch gegen den Fehler „attempt to write a readonly database“, der manchmal bei hohem Datenaufkommen auftreten kann.
- Das Handling beim Schreiben von Werten in Device-Variablen wurde verbessert, sodass der Aufrufer nun feststellen kann, ob der Schreibvorgang erfolgreich war. Wenn Sie beispielsweise in der Webkonfiguration einen Wert in eine Variable schreiben und das

Device nicht verfügbar ist, wird entsprechend ein Fehler angezeigt.

Zusätzlich werden nun auch Werte, die in eine Variable geschrieben werden, in der Node gespeichert, wenn der Schreibvorgang erfolgreich war.

- Nodes mit einem anderen Datentypen als „Null“ können nun auch NULL-Werte bekommen, selbst wenn der Status nicht Bad ist. Dies ermöglicht nun beispielsweise, Null-Werte in Tabellenspalten für das SQL Exchange Plugin korrekt darzustellen.
- Beim Erstellen eines Backups können Sie nun auswählen, dass Logdateien enthalten sein sollen.
- Bei den CoDaBix Settings wurde die neue Einstellung Update DB Mode hinzugefügt, mit der die Anzahl der Schreibvorgänge in die Backend-Datenbank eingeschränkt werden kann. Dies kann für eingebettete System wie beispielsweise Raspberry Pi (auf dem diese Option standardmäßig aktiviert ist) nützlich sein.
- Es wurde ein Problem bei der Verwendung von Node-Subscriptions behoben, das dazu führen konnte, dass ein Node-Wert nicht upgedatet wurde, obwohl er im Gerät geändert wurde.
- Es wurde ein Problem behoben, das beim Erstellen mehrerer Node-Subscriptions für die gleichen Nodes dazu führte, dass Subscription-Parameter wie Intervall nicht korrekt angewendet wurden. Um das Problem zu beheben, musste die Script-API für `codabix.NodeReader` angepasst werden (siehe unten).
- Es wurde ein Bug beim Parsen und Anzeigen von DateTime-Werten behoben, welche zuvor in UTC-Zeit anstatt in lokaler Zeit angezeigt wurden.
- In der CoDaBix® File Access Security (FAS) wurden die standardmäßigen Einträge für den Level „Normal“ geändert, sodass nicht länger ein Nur-Lese-Zugriff für das Installationsverzeichnis enthalten ist (welcher normalerweise nicht benötigt wird).
- Das **SQL Exchange Plugin** wurde auf Version **1.0.3** aktualisiert:
 - NULL-Werte, die erfolgreich aus der Datenbank gelesen wurden, werden nun mit einem Good-Status dargestellt.
- Das **OPC UA Server Interface Plugin** wurde auf Version **1.0.10** aktualisiert:
 - Die Startzeit und der Speicherverbrauch wurden reduziert.
 - Die Verarbeitung von Node-Werten wurde verbessert.
 - Wenn das Schreiben eines CoDaBix-Nodewerts fehlschlägt (z.B. weil das Gerät nicht verfügbar ist), erhält der OPC UA Client nun einen Bad-Status, der den Fehler anzeigt.
 - Der Name eines CoDaBix-Nodes wird nun als `BrowseName` für den entsprechenden OPC-UA-Node verwendet, und der `DisplayName` eines CoDaBix-Nodes wird nun als `DisplayName` für den entsprechenden OPC-UA-Node verwendet.
 - Es wurde ein Bug behoben, der dazu führte, dass existierende Subscriptions von OPC UA Clients nicht mehr funktionierten, wenn bei einer der subscribten Nodes die History Options geändert wurden.
- Das **OPC UA Client Device Plugin** wurde auf Version **1.0.12** aktualisiert:
 - Beim Browsen eines Siemens OPC-UA-Servers, welcher Child-Nodes für jedes Arrayelement erstellt, werden diese Arrayelement-Nodes nicht länger in CoDaBix erstellt, da der Parent-Node bereits den Array-Value enthält.
 - Die maximale Anzahl der gebrowsten Nodes ist nun konfigurierbar.
 - Der `BrowseName` eines OPC-UA-Nodes wird nun als `Name` für den entsprechenden CoDaBix-Node verwendet, und der `DisplayName` eines OPC-UA-Nodes wird nun als `DisplayName` für den entsprechenden CoDaBix-Node verwendet.
- Das **RFC-1006 Device Plugin** wurde auf Version **1.0.2** aktualisiert:
 - Sie können nun den TCP-Port angeben, der für die Verbindung verwendet werden soll.
- **Script-API:**

- **Inkompatible Änderung:** Die Parameter der Methoden `codabix.NodeReader.subscriptionCreated()` und `codabix.NodeReader.subscriptionModified()` wurden geändert, um die Parameter `interval` und `forceUpdate` für jeden `codabix.Node` bereitzustellen, anstatt einzelne Parameter für alle Nodes. Diese Änderung war für die Behebung des oben beschriebenen Problems notwendig, bei dem manchmal Subscription-Parameter nicht korrekt angewendet wurden, wenn mehrere Subscriptions für die gleichen Nodes erstellt wurden.
- Die Methode `codabix.NodeReader.readValuesAsync()` wurde geändert, sodass diese ein Array aus `codabix.NodeValue` zurückgeben kann, womit der `NodeReader` nun direkt die gerade gelesenen Werte angeben kann. Wenn `null` (oder `undefined`) zurückgegeben wird, verwendet CoDaBix die aktuellen Nodewerte als Ergebnis des synchronen Lesevorgangs.
- Die Methode `codabix.readNodeHistoryValues()` wurde durch die Methode `codabix.readNodeHistoryValuesAsync()` ersetzt, damit CoDaBix während des Lesens von historischen Werten (was einige Zeit dauern kann) nicht blockiert ist.
- Die Methoden `codabix.writeNodeValueAsync()` und `codabix.writeNodeValuesAsync()` geben nun einen `codabix.NodeValueStatus` (bzw. ein Array davon) zurück, der angibt, ob die Werte erfolgreich geschrieben werden konnten (wenn die Nodes Device-Variablen sind).
- Die Eigenschaft `codabix.Node.virtualLinkDestination` wurde hinzugefügt, die es Ihnen ermöglicht, das Ziel eines Virtual Links einer Node abzurufen.
- Der Namespace `codabix.nodeconfiguration` wurde hinzugefügt mit Methoden, mit denen es möglich ist, eine Kontextmenü-Action zu registrieren, welche einen Dateidownload als Resultat generieren kann.
- Die Methode `timer.scheduleCallback()` wurde durch die Methode `codabix.scheduleCallback()` ersetzt, da diese tatsächlich eine CoDaBix-spezifische Funktion und keine reguläre Timerfunktion ist.
- Die Methode `timer.yield()` wurde hinzugefügt, die es Ihnen ermöglicht, die Ausführung asynchron zu unterbrechen, indem ein Promise-Objekt zurückgegeben wird, welches „fulfilled“ wird, nachdem die derzeitige Scriptausführung beendet wird (und der aktuelle `NodeLock` freigegeben wird).
- Scripts können nun mit `runtime.onShutdown` einen Shutdown-Handler registrieren, der aufgerufen wird, wenn das Script auf normale Weise gestoppt wird (z.B. weil die Engine heruntergefahren wird oder der Benutzer das Script gestoppt hat).

0.17.1 (freigegeben am 10.08.2018)

- **[Raspberry Pi]** Das Deployment wird nun als selbstextrahierendes Setup ausgeliefert.
- **[Raspberry Pi]** Ein Fehler im Setup wurde behoben, durch den der Installationsordner falsche Zugriffsberechtigungen erhalten hatte.
- **[Raspberry Pi]** Ein Fehler im **OPC UA Server Interface Plugin** wurde behoben, der dazu führte, dass ein Channel, mit dem in OPC UA Client verbunden ist, nicht neugestartet werden konnte.
- **OPC UA Client Device Plugin** Ein Channel kann nun im Simulationsmodus betrieben werden. Dadurch kann der Channel ohne einen vorhandenen Server betrieben werden, um eine Simulation durchzuführen.

0.17.2 (freigegeben am 13.08.2018)

- **OPC UA Client Device Plugin** Bugfix im Simulationsmodus: Wenn ein Node keinen Wert besitzt, wird nun ein Default-Wert verwendet.

0.17.5 (freigegeben am 03.09.2018)

- Das **OPC UA Client Device Plugin** wurde auf Version **1.0.13** aktualisiert:
 - Beim Browsen eines OPC UA-Servers, der das `Description`-Attribut für Nodes nicht ausgibt (z.B. bei einem S7-1500 OPC UA Server), können Sie nun die Description-Werte über eine exportierte NodeSet-XML-Datei bereitstellen, indem Sie dieses im Pfad `<CodabixDataDir>\OpcUaClientDevicePlugin\<ChannelName>\NodeSet.xml` ablegen.
- Das **OPC UA Server Interface Plugin** wurde auf Version **1.0.11** aktualisiert:
 - Es können nun zusätzliche Servereinstellungen (z.B. Transport-Settings) konfiguriert werden.
- **Script-API:**
 - `Node/Value-Tuple`-Objekte, die an die `codabix.writeNodeValuesAsync()`-Methode übergeben werden, können nun zusätzlich eine `source`-Property enthalten mit einem Wert, der später im `ValueChanged`-Event einer `codabix.Node` zur Verfügung gestellt wird; beispielsweise um zu erkennen, wenn ein `NodeWriter` einen Wert zurück in die Node schreibt, nachdem er erfolgreich in das Device geschrieben wurde.

0.17.7 (freigegeben am 26.09.2018)

- **[Raspberry Pi]** Wenn CoDaBix als Dienst läuft, nimmt der Dienst nach dem Binden der Web-Ports die Identität des Benutzers an, der den Dienst ursprünglich installiert hat, um Probleme mit Dateiberechtigungen zu vermeiden.
- Die Verwendung von **MariaDB** 10.3 oder höher als Backend-Datenbank wird nun offiziell unterstützt.
- Die Performance beim Einfügen von großen Datenmengen in eine MySQL- oder MariaDB-Backend-Datenbank (z.B. beim Wiederherstellen eines Backups oder beim Einfügen von historischen Werten) wurde spürbar verbessert.
- Wenn das Scannen eines Verzeichnisses, das durch eine `Directory Node` spezifiziert wird, fehlschlägt, werden die Kindknoten nicht gelöscht (da der Fehler als temporärer Fehler behandelt wird), und eine Warnung wird ins Runtime Log geschrieben.
- Das **OPC UA Server Interface Plugin** wurde auf Version **1.0.12** aktualisiert:
 - Es wurde ein Bug behoben, der dazu führen konnte, dass CoDaBix nicht mehr reagiert, wenn eine große Anzahl an Requests von OPC-UA-Clients gleichzeitig eintreffen.
- Das **SQL Exchange Plugin** wurde auf Version **1.0.4** aktualisiert:
 - Es wurde ein Bug behoben, der das Browsen einer Tabelle direkt nach deren Erstellung verhinderte.
- Das **REST Interface** wurde aktualisiert:
 - Bei `get`- und `browse`-Requests wird nun die Einheit der Node über die `unit`-Property ausgegeben, und die History-Options werden nun korrekt über die `hi`-Property ausgegeben.

0.17.8 (freigegeben am 23.10.2018)

- Es wurde ein Bug behoben, der unter bestimmten Umständen dazu führen konnte, dass Requests von OPC-UA-Clients (über das OPC UA Server Interface) fehlschlagen und Fehlermeldungen ins Runtime-Log geloggt werden.

0.16

0.16.0 (freigegeben am 13.04.2018)

- Benutzergruppen (User Groups) können nun als Nur-Lese-Zugriff (*Read-only*) festgelegt werden, sodass Benutzer aus dieser Gruppe nur Lesezugriff auf die dieser Gruppe zugehörigen Nodes haben, aber keine Werte schreiben oder Nodes verändern können. Diese Einstellung wird im REST Interface sowie im OPC UA Server Interface geprüft.
- Beim Ausliefern von statischen Dateien im `webfiles`-Ordner liefert der eingebettete Webserver nun auch GZIP-komprimierte Dateien in ihrer komprimierten Form aus, die auf `.gz` enden, wenn der Dateiname ohne `.gz` angefragt wird, wodurch der Netzwerk-Traffic reduziert werden kann.
- Das **Melsec QJ Device Plugin** (Version 1.0.0) wurde hinzugefügt, das es Ihnen ermöglicht, Verbindungen zu Mitsubishi-SPSen der Melsec-Q- und Melsec-A-Serie herzustellen.
- Das **AKLAN Device Plugin** (Version 1.0.0) wurde hinzugefügt, das es Ihnen ermöglicht, Daten an AKLAN-Dienste zu senden. Beachten Sie, dass dieses Plugin zusätzliche Komponenten benötigt, welche manuell installiert werden müssen.
- Das **Demo Data (Continuous)**-Plugin wird beim Ausführen des CoDaBix®-Compact-Setups nicht länger standardmäßig installiert.
- Das **OPC UA Server Interface Plugin** wurde auf Version **1.0.9** aktualisiert:
 - Die Zugriffszeit (z.B. wenn ein Client den Server browsst) wurde verbessert, wenn sehr viele Nodes erstellt wurden.
 - Beim Schreiben von Nodewerten wird nun geprüft, ob der User nur Lesezugriff hat.
 - Eine neue Einstellung „Hostname Mode“ wurde zu den Channel-Einstellungen hinzugefügt, mit der man auswählen kann, ob als Hostname des Endpoints „localhost“, der aktuelle Computername oder ein benutzerdefinierter Name verwendet werden soll.
 - Der Typ `CodabixBlob` wird nun dem OPC UA Data Type `ByteString` zugeordnet.
 - Es wurde ein Bug (bestehend seit Version **1.0.8**) behoben, der dazu führte, dass die `AutoAcceptUntrustedCertificates`-Einstellung ignoriert wurde.
 - Es wurde ein Bug behoben, der dazu führte, dass der Server Skalartypen fälschlicherweise als Arraytypen im `ValueRank`-Attribut angab.
- Das **OPC UA Client Device Plugin** wurde auf Version **1.0.10** aktualisiert:
 - Der OPC UA Data Type `ByteString` wird nun dem Typ `CodabixBlob` zugeordnet.
- Das **S7 Device Plugin** wurde auf Version **1.0.9** aktualisiert:
 - Die Option `IsSimulation` wurde zu den Channel-Settings hinzugefügt, die es ermöglicht, einen S7-Channel im Simulationsmodus zu betreiben, wodurch er nicht tatsächlich mit einer physischen SPS kommuniziert.
 - Die Performance wurde verbessert, wenn viele synchrone Lesevorgänge für die selbe Variable ausgeführt werden.
 - Es wurde ein Bug behoben, der zu fehlerhafter Dekodierung von Umlauten beim Importieren von STEP7-Projekten führte.
 - Es wurde ein Bug behoben, der zu fehlenden Blöcken, Typen, Werten und Objekten beim Importieren von STEP7-Projekten führen konnte.
- Das **CSV Exchange Plugin** wurde auf Version **1.0.4** aktualisiert:

- Ein Intervalltrigger wurde hinzugefügt, der im SyncMode SystemToFile verwendet werden kann.
- Es wurde ein Bug behoben, der dazu führte, dass das ColumnIndex-Attribut beim Synchronisieren von CSV-Dateien zu CoDaBix ignoriert wurde.

- **Script-API:**

- Die Methode `close()` in `io.Closeable` wurde durch die asynchrone Methode `closeAsync()` ersetzt, da das Schließen einer Resource möglicherweise eine blockierende Operation ist. Existierende Scripts, die die `close()`-Methode aufrufen, funktionieren nach dem Update auf diese Version weiterhin (solange sie die Methode nicht aufrufen, während eine asynchrone Operation noch läuft), und loggen eine Warnung.
- Das globale Objekt `Guid` wurde hinzugefügt, welches es ermöglicht, einen neuen GUID-String zu generieren.
- Die Methode `io.util.encodeXml()` wurde nach `xml.encode()` verschoben. Existierende Scripts, die die vorherige Methode verwenden, funktionieren nach dem Update auf diese Version weiterhin.
- Das Script-Timeout wurde von 1500 ms auf 5000 ms erhöht.

0.16.2 (freigegeben am 14.05.2018)

- Es wurde ein Bug beim Schreiben von Werten in Device-Variablen behoben, der dazu führen konnte, dass der zu schreibende Wert ignoriert wurde, wenn der Schreibaufwurf rekursiv aus einem im ValueChanged-Event des Devices zur Ausführung geplanten Handler aufgerufen wurde, z.B. in einem Script, das `timer.schedule()` verwendet.
- Es wurde ein Bug im Diff Script Editor behoben, der das Speichern des Scripts verhinderte.
- Das Handling in Bezug auf die Zeitmessung in Device Worker Threads wurde verbessert, um sicherzustellen, dass Variablen, für die eine Subscription erstellt wurde, rechtzeitig gelesen werden, auch wenn sehr viele synchrone Lesevorgänge bearbeitet werden.
- Pluginlizenzen können nun auch dann verwaltet werden, wenn CoDaBix® Compact als Dienst läuft oder die CoDaBix® Engine nicht läuft.
- Das **OPC UA Client Device Plugin** wurde auf Version **1.0.11** aktualisiert:
 - Beim Browsen eines OPC-UA-Servers wird nun das **Description**-Attribut eines OPC-UA-Nodes gelesen und im erstellten CoDaBix-Node gespeichert.
- Das **RFC-1006 Device Plugin** wurde auf Version **1.0.1** aktualisiert:
 - Es wurde ein Bug behoben, der dazu führte, dass der Typ des ReceiveBuffer- oder TransmitBuffer-Nodes beim Starten auf Blob geändert wird, wenn der Benutzer diesen zuvor auf String geändert hat.
- **Script API:**
 - Die Methoden `codabix.subscribeNodes()` und `codabix.unsubscribeNodes()` wurden hinzugefügt, mit denen Sie explizit eine Subscription für eine oder mehrere Nodes erstellen und dabei das Subscription-Intervall angeben können.
 - Die Methoden `codabix.Node.registerReader()` und `codabix.Node.unregisterReader()` wurden hinzugefügt, mit denen Sie einen **NodeReader** registrieren können, der aufgerufen wird, wenn ein synchroner Lesevorgang für die angegebenen Nodes angefordert oder eine Subscription erstellt wird.

0.16.3 (freigegeben am 24.05.2018)

- Es wurde ein Bug (bestehend seit Version **0.16.2**) behoben, der manchmal dazu führen

konnte, dass der erste Schreibvorgang in eine Device-Variable ignoriert wurde.

- Das **SQL Exchange Plugin** wurde auf Version **1.0.1** aktualisiert:
 - Es wurde Unterstützung für ODBC-Datenbankverbindungen hinzugefügt, wobei der Database Name des Channels als ODBC Data Source Name (DSN) verwendet wird.

0.16.4 (freigegeben am 20.06.2018)

- Es wurde ein Bug behoben, der dazu führte, dass HTTPS-Bindings bei Angabe eines Hostnamens nicht funktionierten.
- Das **SQL Exchange Plugin** wurde auf Version **1.0.2** aktualisiert:
 - Sie können nun 0 als Port für den Microsoft SQL Server angeben, wodurch das Plugin versucht, eine Verbindung zur standardmäßigen SQL-Server-Instanz herzustellen.
- Das **Database Plugin** wurde auf Version **1.0.10** aktualisiert:
 - Sie können nun 0 als Port für den Microsoft SQL Server angeben, wodurch das Plugin versucht, eine Verbindung zur standardmäßigen SQL-Server-Instanz herzustellen.

0.15

0.15.0 (freigegeben am 18.01.2018)

- **Inkompatible Änderung:** CoDaBix benötigt nun das **.NET Framework 4.6.1** oder höher. Weitere Informationen finden Sie unter **Installation von CoDaBix®**.
- Der **eingebettete Webserver** kann nun statische Dateien aus dem Ordner `webfiles` des Datenverzeichnisses ausliefern. Dies ermöglicht Ihnen in Kombination mit dem neuen Script-HTTP-Handler-Feature (siehe unten), Webapps zu entwickeln, die durch den CoDaBix®-Webserver ausgeliefert werden (z.B. zur Visualisierung). Zusätzlich können Sie nun in den CoDaBix® Settings eine benutzerdefinierte HTTP-Redirect-URL für den Root-Pfad angeben, sodass Nutzer nur beispielsweise `http://<hostname>/` eingeben müssen, um die benutzerdefinierte Webseite anzuzeigen.
- Das **Datenverzeichnis** enthält nun einen neuen Ordner `userdata`, in dem Sie benutzerdefinierte Dateien speichern können. Bei diesem Ordner ist garantiert, dass er in zukünftigen CoDaBix®-Versionen nicht anderweitig verwendet wird.
- Die Sicherheit von **verschlüsselten Passwörtern**, die in Datenpunktnodes vom Typ „Passwort“ abgelegt oder über die Script-API `codabix.security.encryptPassword()` erstellt werden, wurde verbessert.
- CoDaBix® File Access Security (FAS) verhindert nun den Schreibzugriff auf die SQLite-Backend-Datenbankdateien (außer bei der Einstellung „Full Access“), da diese als Systemdateien betrachtet werden.
- Die Eigenschaften `HistoryOptions` und `HistoryInterval` eines verlinkten Nodes werden nun vom Linkziel abgerufen, da sie in enger Beziehung zu den historischen Werten des Datenpunktnodes stehen.
- Die **Log-Rotation** wurde verbessert, sodass die CoDaBix® Runtime-Logdatei, die Database Plugin-Logdatei sowie Logdateien des Entity Models nicht länger jeden Tag rotiert werden, sondern sobald sie eine Größe von mehr als 5 MiB erreicht haben, in welchem Fall sie außerdem automatisch in eine ZIP-Datei komprimiert werden, um Speicherplatz zu sparen. Außerdem werden nun standardmäßig Logdateien, die älter als 30 Tage sind, automatisch gelöscht.
- CoDaBix® Compact kann nun auf Windows-Systemen ausgeführt werden, die die

- Richtlinie „Systemkryptographie: FIPS-konformen Algorithmus für Verschlüsselung, Hashing und Signatur verwenden“ aktiviert haben.
- Das Lizenz-Handling wurde in Bezug auf den Maschinencode verbessert.
 - Das **CSV Exchange Plugin** wurde auf Version **1.0.2** aktualisiert:
 - **Inkompatible Änderung:** Der Ordnername des Plugins im Datenverzeichnis wurde von `B. CSV-Server` in `CsvExchangePlugin` umbenannt, um der Dokumentation zu entsprechen. Wenn Sie CoDaBix® Compact von einer früheren Version updaten, versucht CoDaBix® automatisch, den Ordner zumzubenennen; Sie sollten jedoch prüfen, ob Sie den früheren Pfad in anderen Anwendungen benutzen.
 - **Inkompatible Änderung:** Bei Verwendung des SyncModes `SystemToFile` werden CSV-Zeilen nicht länger implizit geschrieben, wenn sich der Wert eines der Nodes ändert. Stattdessen können Sie nun Trigger (ähnlich wie beim Database Plugin) definieren, um festzulegen, wann Werte in die CSV-Datei geschrieben werden sollen. Außerdem führt das CSV Exchange Plugin nun einen synchronen Lesevorgang (wie das Database Plugin) durch, wenn ein Trigger auslöst.
 - Bei Verwendung des Modus `SystemToFile` können Sie nun angeben, dass ein Systemwert statt eines Nodewerts für ein `<Binding>` verwendet werden soll, z.B. um die aktuelle Uhrzeit in die CSV-Datei zu schreiben.
 - **Script-API:**
 - Sie können nun **HTTP-Handler** durch den Aufruf der Methode `net.registerHttpRequest()` registrieren. Dies ermöglicht Ihnen, asynchrone Funktionen festzulegen, die aufgerufen werden, wenn ein Client einen HTTP-Request nach CoDaBix® ausführt. Damit können Sie beispielsweise dynamisch HTML-Seiten, ähnlich wie bei PHP oder ASP.NET, generieren. Dieses Feature ermöglicht Ihnen außerdem, serverseitige **WebSocket-Verbindungen** über das `net.RawWebSocket`-Interface zu handeln, wenn CoDaBix® Compact unter Windows 8 / Windows Server 2012 oder höher ausgeführt wird.
 - Es wurden Überladungen zur `timer.scheduleCallback()`-Methode hinzugefügt, die einen Callback mit Parametern akzeptieren, und darüber hinaus weitere Argumente akzeptieren, die dem Callback übergeben werden. Dies ermöglicht beispielsweise für einen Eventlistener den Callback abzuspeichern und durch das Durchreichen der Event-Parameter wiederzuverwenden, anstatt bei jedem Eventlistener-Aufruf ein Closure zu erstellen, was relativ teuer ist.
 - Zu den Methoden `copyFileAsync()`, `writeAllTextAsync()` und `openFileWriterAsync()` des `io.file`-Namespaces wurde ein Parameter hinzugefügt, mit dem Sie angeben können, dass eine Ausnahme geworfen werden soll, falls die Datei bereits vorhanden ist, anstatt diese zu überschreiben.
 - Das Script-Timeout wurde von 1000 ms auf 1500 ms erhöht.
 - **Inkompatible Änderung:** Die Eigenschaft `fileLength` der Klasse `io.FileReader` wurde durch die asynchrone Methode `getFileLengthAsync` ersetzt, da das Abrufen der aktuellen Dateilänge eine I/O-Operation erfordert, die möglicherweise blockierend ist.

0.14

0.14.0 (freigegeben am 13.07.2017)

- In den CoDaBix® File Access Security (FAS)-Einstellungen können nun für jeden Pfad **Zugangsdaten** angegeben werden. Somit kann nun beispielsweise in eine Script auf

- Dateien zugegriffen werden, auf die der aktuelle Benutzer keinen Zugriff haben würde (intern wird dafür Impersonation verwendet). Zusätzlich kann im Falle eines UNC-Pfads dieser als Netzwerkresource hinzugefügt werden. Dies ist nützlich, wenn CoDaBix® als Dienst ausgeführt wird, da dieser keinen Zugriff auf die verbundenen Windows-Netzlaufwerke (mit entsprechenden Zugangsdaten) des aktuellen Benutzers hat.
- Bei Device Plugins, Interface Plugins sowie Exchange Plugins werden nun die wichtigsten Einstellungen direkt angezeigt, wenn ein Kanal in der Webkonfiguration angelegt oder editiert wird.
Weiterhin wird nun ein **Control Panel** beim Öffnen eines Kanals angezeigt, das eine Live-Übersicht des Kanals darstellt.
 - Die Eigenschaften *Scaling Factor* und *Scaling Offset* von Datenpunktnodes werden nun verwendet, um Werte zu skalieren, die von einem Gerät gelesen (oder in dieses geschrieben) werden.
Da diese Eigenschaften bisher nicht genutzt wurden (man jedoch beliebige Werte eingeben konnte), werden diese Eigenschaften beim Updaten auf diese Version zurückgesetzt, damit vorherige Werte nicht zu Problemen führen.
 - Beim Erstellen eines **Backups** können Sie nun angeben, dass **Passwörter nicht exportiert** werden sollen (aus Sicherheitsgründen). Dies gilt für das MySQL-Backend-Datenbankpasswort und für Passwörter, die in Nodes vom Typ *Password* abgelegt werden (z.B. bei OPC UA Client Device und SQL Exchange), sowie ebenfalls für über die Script-API `codabix.security.encryptPassword()` verschlüsselte Passwörter, die z.B. in Scripts abgelegt werden.
 - Das CoDaBix® Runtime-Protokoll wird nun im `log`-Ordner statt im `logfiles`-Ordner abgelegt.
 - Wenn der Start eines Plugins fehlschlägt, wird nach 30 Sekunden versucht es neuzustarten, damit temporäre Fehlerzustände nicht verhindern, dass das Plugin nach dem Beheben des Fehlers wieder arbeiten kann.
 - Historische Werte können nun im CoDaBix® Settings-Dialog gelöscht werden.
 - Das **REST Interface** wurde aktualisiert:
 - Authentifizierung per Benutzername und Passwort wurde implementiert.
 - Historische Werte können nun für eine bestimmte Zeitspanne abgerufen werden.
 - Das **CSV Exchange Plugin** wurde auf Version **1.0.1** aktualisiert:
 - Unterstützung für Dateiübertragung über einen **SSH-Server** (SCP oder SFTP) wurde hinzugefügt.
 - Das `CustomerName`-Attribut muss nicht länger in der XML-Konfigurationsdatei angegeben werden.
 - Das **S7 Device Plugin** wurde auf Version **1.0.7** aktualisiert:
 - Es wurde ein Bug behoben, der das Lesen von Strings mit einer Maximallänge von mehr als 221 Zeichen verhinderte.
 - Das **OPC UA Server Interface Plugin** wurde auf Version **1.0.6** aktualisiert:
 - Die Policy-Einstellungen wurden refaktoriert, sodass mehrere Policies für den selben Endpunkt (Channel) aktiviert werden können.
 - Das automatisch erstellte Anwendungsinstanz-Zertifikat verwendet jetzt den Computernamen anstatt „localhost“ als Domainnamen.
 - Es wurde ein Bug behoben, der dazu führte, dass bei jedem Start von CoDaBix® ein neues Anwendungsinstanz-Zertifikat generiert wurde.
 - Das **OPC UA Client Device Plugin** wurde auf Version **1.0.7** aktualisiert:
 - Es wurde eine zusätzliche Einstellung hinzugefügt, die angibt, ob eine normale OPC UA Subscription oder Polling für eine CoDaBix® Subscription auf OPC UA Client Variablen angewendet werden soll.
 - Es wurde ein Bug behoben, der dazu führte, dass bei jedem Start von CoDaBix®

ein neues Anwendungsinstanz-Zertifikat generiert wurde (und der verhinderte, dass das Zertifikat als Blob-Wert im entsprechenden CoDaBix® Node abgelegt wurde).

- Das **SQL Exchange Plugin** (Version 1.0.0) wurde hinzugefügt, das es Ihnen ermöglicht, Daten bidirektional mit einer MySQL-, MSSQL- oder Oracle-Datenbank auszutauschen.
- **Script-API:**
 - Das Event `NodeChangeCompleted` wurde zum `codabix`-Namensraum hinzugefügt, welches ausgelöst wird, wenn eine Serie von mehrfachen `ValueChanged`-, `PropertyChanged`- oder `ChildrenChanged`-Ereignissen von Nodes abgeschlossen ist.
 - Die Methode `codabix.Node.registerValueConverter()` wurde hinzugefügt, mit der eine Rückmeldung registriert werden kann, die aufgerufen wird, wenn ein Wert von einem Gerät gelesen (oder in ein Gerät geschrieben) wird, sodass der Wert nachbearbeitet werden kann.
 - Ein neuer Parameter wurde zur `logger.log()`-Methode hinzugefügt, der ein Log-Level akzeptiert (der Standardwert ist `Info`). Zusätzlich wurden die Methoden `logDebug()`, `logInfo()`, `logWarning()` und `logError()` zum `logger`-Namensraum hinzugefügt.
 - Der `codabix.security`-Namensraum wurde hinzugefügt, der Methoden zum Verschlüsseln und Entschlüsseln von Passwörtern enthält, die in einem Passwortnode gespeichert werden sollen.

0.14.1 (freigegeben am 27.07.2017)

- Das Log-Handling für die Webkonfiguration wurde verbessert: Fehlereinträge im Runtime-Protokoll werden nicht länger direkt angezeigt; stattdessen zeigt ein Button die Anzahl neuer Logeinträge an. Beim Klick auf den Button öffnet sich ein (Live-)Dialog, der die letzten Logeinträge anzeigt.
- Die maximale Länge der Pfadereigenschaft des Nodes wurde von 200 auf 2000 Zeichen erhöht, um z.B. längere Dateipfade zu unterstützen.

0.14.2 (freigegeben am 06.09.2017)

- Beim Erstellen eines **Backups** können Sie nun angeben, dass historische Werte nicht eingeschlossen werden sollen, wenn Sie diese nicht benötigen, um die Größe des Backups zu reduzieren.
- Das **OPC UA Client Device Plugin** wurde auf Version **1.0.8** aktualisiert:
 - Es wurde ein Speicherleck behoben, das auftreten konnte, wenn ein Server nicht erreichbar ist, während der Client versucht, sich zu verbinden.
 - Das Verhalten beim Browsen wurde verbessert, sodass für einen gebrowsten Data Variable Node eine genauere Typinformation abgerufen werden kann, anstatt einfach den „Null“-Datentyp zu verwenden.
 - Unterstützung für den `Basic256Sha256`-Security Policy Algorithm wurde hinzugefügt.
- Das **OPC UA Server Interface Plugin** wurde auf Version **1.0.7** aktualisiert:
 - Es wurde ein Speicherleck behoben, das auftreten konnte, wenn Clients sich oft verbinden und die Verbindung wieder trennen.
 - Unterstützung für den `Basic256Sha256`-Security Policy Algorithm wurde hinzugefügt.
- **Script API:**
 - Die Deklaration von `codabix.NodeValueConverter` wurde verbessert, um das Zurückgeben von `undefined` zusätzlich zu `null` zu erlauben.

0.14.3 (freigegeben am 28.09.2017)

- Es wurde ein Bug behoben, der dazu führen konnte, dass das CSV Exchange Plugin hängen blieb, wenn ein Dateiaustausch über SCP verwendet wurde.
- Kleinere Bugfixes und Stabilitätsverbesserungen.

0.14.4 (freigegeben am 09.10.2017)

- Es wurde ein Bug behoben, der dazu führen konnte, dass das CSV Exchange Plugin bei Dateizugriff über SMB (Windows-Freigaben) nach einem Netzwerkproblem keine Dateiänderungen mehr erkannte.

0.14.12 (freigegeben am 24.11.2017)

- Die **Log-Rotation** wurde verbessert, sodass die CoDaBix® Runtime-Logdatei und die Database Plugin-Logdatei nicht länger jeden Tag rotiert werden, sondern sobald sie eine Größe von mehr als 5 MiB erreicht haben, in welchem Fall sie außerdem automatisch in eine ZIP-Datei komprimiert werden, um Speicherplatz zu sparen. Außerdem werden nun standardmäßig Logdateien, die älter als 30 Tage sind, automatisch gelöscht.
- Das **Database Plugin** wurde auf Version **1.0.9** aktualisiert:
 - Das `instantChangeBackValue`-Attribut wurde zum Trigger-Element hinzugefügt (für `type="edge"`), mit dem Sie einen Wert angeben können, der sofort nach dem Auslösen des Triggers in die Node geschrieben wird; im Gegensatz zum `changeBackValue`-Wert, der erst dann geschrieben wird, wenn das Recordset gesammelt wurde (d.h. der synchrone Lesevorgang abgeschlossen ist).
 - Die Performance wurde durch das Schreiben von Zeilen in einer einzelnen Transaktion, wenn viele Zeilen gepuffert sind, verbessert.

0.14.13 (freigegeben am 20.12.2017)

- CoDaBix® Compact kann nun auf Windows-Systemen ausgeführt werden, die die Richtlinie „Systemkryptographie: FIPS-konformen Algorithmus für Verschlüsselung, Hashing und Signatur verwenden“ aktiviert haben.
- Die verbesserte Log-Rotation (inklusive Erstellen einer ZIP-Datei) wird nun auch auf Logdateien des Entity Models angewendet.
- Das **OPC UA Client Device Plugin** wurde auf Version **1.0.9** aktualisiert:
 - Die Performance beim Browsen eines OPC-UA-Servers wurde verbessert.
 - Es wurde ein Bug behoben, der dazu führte, dass unter einem bereits existierenden Folder-Node hinzugefügte Variablen nicht funktionierten, bis CoDaBix® neugestartet wurde.
 - Der festgelegte `OperationTimeout`-Wert wird nun korrekt auf Verbindungsversuche angewendet.
 - Nodelds können nun in einem Format wie `ns=2;s=ABC` angegeben werden, das auch von anderen OPC UA Tools verwendet wird.
- Das **OPC UA Server Interface Plugin** wurde auf Version **1.0.8** aktualisiert:
 - Das `ValueRank`-Attribut von Datenpunktnodes wird nun für Arraytypen korrekt spezifiziert.

0.12

0.12.0 (freigegeben am 09.05.2017)

- Im CoDaBix® Settings-Dialog kann nun das Protokoll-Level festgelegt werden, der bestimmt, wie detailliert Protokolldateien geschrieben werden.
- Log-Rotation wurde implementiert, sodass nun auch die Report-Logdateien nach einem Tag rotiert werden.
- Wenn SQLite als Backend-Datenbank verwendet wird, werden historische Werte nun in eine separate Datenbank (codabixhistorydb.db) abgelegt, um die Performance zu verbessern.
- Aus Sicherheitsgründen wird bei neuen CoDaBix® Installationen, bei denen CoDaBix® als Dienst installiert wird, nicht länger automatisch ein Remote-HTTP-Binding für den lokalen HTTP-Port hinzugefügt.
Wenn Sie den Zugriff auf CoDaBix® von anderen Geräten im Netzwerk erlauben möchten, müssen Sie die Option „Use the Local Port as remote HTTP Binding“ in den „Service HTTP(S) Bindings“-Optionen aktivieren, oder manuell HTTP-/HTTPS-Bindings hinzufügen.
- Es wurde ein Speicherleck behoben, das beim Ausführen als Dienst auftrat, welches nach einiger Zeit zu einem Crash führen konnte.
- Es wurde ein Bug behoben, der das Wiederherstellen von Backups verhinderte, wenn diese einen Datenpunktnode vom Typ DateTime enthielten.
- Es wurde ein Bug behoben, der das Wiederherstellen von Backups auf Windows-Server-Systemen verhinderte, die in der **Server-Core**-Variante installiert wurden (d.h. ohne Desktopdarstellung).
- Das **S7 Device Plugin** wurde auf Version **1.0.6** aktualisiert:
 - Es wurde ein Bug behoben, der bei Verwendung von Char-Variablen dazu führte, dass zu viele Bytes gelesen oder geschrieben wurden.
- Das **Database Plugin** wurde auf Version **1.0.7** aktualisiert:
 - Es wurde ein Speicherleck behoben, das auftreten konnte, wenn Trigger schneller auslösen, als Datensätze in die Datenbank geschrieben werden können. Dies konnte zu einem Absturz führen.
- **Script-API:**
 - Der Script Editor zeigt nun innerhalb einer async-Funktion einen Fehler bei Methodenaufrufen an, die ein Promise-Objekt zurückgeben, wenn das zurückgegebene Objekt nicht mit await erwartet oder anderweitig verwendet wird. Wenn das „Vergessen“ des Promise-Objekts beabsichtigt ist, können Sie den void-Operator benutzen, um den Fehler auszublenden, z. B. `void codabix.writeNodeValueAsync()`.

0.12.1 (freigegeben am 11.05.2017)

- Die Startzeit beim Ausführen von CoDaBix® Compact als Dienst wurde verbessert.
- Es wurde ein Bug behoben, der manchmal verhinderte, dass ein Node über die Webkonfiguration dupliziert werden konnte.
- Das **Database Plugin** wurde auf Version **1.0.8** aktualisiert:
 - Ein neuer Triggertyp `valueChange` wurde hinzugefügt, der auslöst, wenn sich der Wert eines Nodes ändert.
 - Das `property`-Attribut wurde zum `<Node>`-Element hinzugefügt, mit dem Sie die Eigenschaft des Nodes angeben können, die in die Datenbank geschrieben werden soll. Anstatt des Nodewerts können Sie nun auch den Zeitstempel des Nodewerts, die Einheit des Nodes usw. schreiben.
 - Wenn ein angegebener Node nicht gefunden wird, wird nun eine bessere Fehlermeldung geloggt.
- Das **OPC UA Client Device Plugin** wurde auf Version **1.0.5** aktualisiert:

- Wenn ein OPC UA Server gebrowst wird, werden OPC Nodes mit den Datentypen Integer und UInteger nun als CoDaBix® Nodes mit den Typen Int64 und UInt64 angelegt anstatt des Typs Null, sodass deren Werte gelesen werden können. Beachten Sie aber, dass Integer und UInteger abstrakte Datentypen in der OPC UA sind, weswegen ein Schreibvorgang fehlschlagen kann und Sie in diesem Fall einen anderen, kleineren Datentyp wie Int32 für den Node auswählen müssen.

0.12.2 (freigegeben am 16.05.2017)

- Wenn ein einzelnes Plugin nicht gestartet werden kann, wird eine Fehlermeldung ins Runtime-Protokoll geschrieben, und anschließend setzt CoDaBix® nun den Startvorgang fort.
- Es wurde eine Verknüpfung in der CoDaBix® Compact Programmgruppe hinzugefügt, mit der Sie die historischen Werte in der aktuellen Backend-Datenbank löschen können.
- Das **OPC UA Client Device Plugin** wurde auf Version **1.0.6** aktualisiert:
 - Standardmäßig sind nur 10 gleichzeitige Channels (Verbindungen) erlaubt. Die Anzahl kann über eine Lizenz erhöht werden.
- **Script-API:**
 - Es wurde ein Speicherleck behoben, das auftreten konnte, wenn Scripts häufig neugestartet werden. Um das Speicherleck zu beheben, musste die `codabix.Node.children`-Eigenschaft so angepasst werden, dass diese kein Live-Array mehr zurückgibt, d.h. es wird bei jedem Zugriff auf die Eigenschaft ein neues Array zurückgegeben.

0.11

0.11.0 (freigegeben am 27.02.2017)

- Die Webkonfiguration wurde überarbeitet:
 - Es wurde ein Kontextmenü implementiert.
 - Foldernodes werden in der Baumansicht erst dann geladen, wenn sie aufgeklappt werden.
 - Variablennodes werden nun ebenfalls in der Baumansicht angezeigt, anstatt nur in der Tabellenansicht.
 - Wenn die historischen Werte eines Nodes mit numerischen Werten angezeigt werden, wird ein Liniendiagramm mit den Historywerten eingeblendet.
- Es wurde ein Bug behoben, der das Duplizieren von Nodes verhinderte.
- Es wurde ein Bug behoben, der das Erstellen von neuen Benutzern verhinderte.
- Das **S7 Device Plugin** wurde auf Version **1.0.4** aktualisiert:
 - Die Indirektion über den „Variables“-Foldernode beim Erstellen einer Variable vom Typ „Object“ wurde entfernt.
 - Es wurde ein Bug behoben, der die Verwendung des **S5-String**-Datentyps verhinderte.
- Das **OPC UA Client Device Plugin** wurde auf Version **1.0.3** aktualisiert:
 - Die Indirektion über den „Variables“-Foldernode beim Erstellen eines Unterordners wurde entfernt.
- Das **Database Plugin** wurde auf Version **1.0.5** aktualisiert:
 - Das Trigger-Handling wurde so geändert, dass ein synchroner Lesevorgang direkt im Eventhandler des Triggers, statt kurze Zeit später, ausgeführt wird. Dadurch werden die gesammelten DataSet-Werte deterministischer, vor allem wenn der Wert eines Edge-Triggers ebenfalls gesammelt wird.

0.11.1 (freigegeben am 12.04.2017)

- Eine **Backup-And-Restore**-Funktion (Sichern und Wiederherstellen) wurde implementiert, mit der Sie die aktuelle **Backend-Datenbank**, das **Datenverzeichnis** sowie die **CoDaBix® Einstellungen** in einer einzigen komprimierten Datei sichern und später wiederherstellen können. Dies funktioniert sogar dann, wenn Sie beim Wiederherstellen einen anderen Backend-Datenbanktyp (MySQL/SQLite) verwenden.
- Im CoDaBix® Settings Dialog können Sie nun die Einstellungen speichern und laden.
- Es können nun Anweisungen von Methodennodes in der Webkonfiguration ausgeführt werden.
- Es wurde ein zusätzliches Editor Strictness Level „Medium“ für Scripts hinzugefügt, bei dem der Script Editor keine Fehler bei unbenutzten lokalen Variablen sowie unbenutzten Parametern anzeigt.
- Wenn CoDaBix® Compact als Dienst installiert wird, wird der Starttyp nun auf „Automatisch (Verzögerter Start)“ gesetzt, um sicherzustellen, dass der CoDaBix® Dienst gestartet werden kann, auch wenn die Festplatte des Computers beschäftigt oder die MySQL-Datenbank noch nicht verfügbar ist.
- Es wurde ein Bug behoben, der verhinderte, dass FAS (File Access Security) „Deny“-Einträge korrekt verarbeitet.
- Die Performance beim Lesen von durch einen Zeitstempel gefilterten historischen Werten von der eingebetteten (SQLite) Backend-Datenbank wurde verbessert.
- Das **OPC UA Client Device Plugin** wurde auf Version **1.0.4** aktualisiert:
 - Die Implementierung wurde umgeschrieben, um einen dedizierten Thread pro Channel zu benutzen. Dies vermeidet Verzögerungen in CoDaBix und verbessert die Stabilität des Plugins.
 - Eine Browse-Funktionalität wurde implementiert. Durch das Ausführen des **Browse**-Kommandos wird der OPC UA Server gescannt und neue Variablenodes werden automatisch im Kanal erstellt.
 - Es wurde ein Bug behoben, der in seltenen Fällen zum Hängenbleiben beim Starten führen konnte.
 - Es wurde ein Bug behoben, der beim Handling von Subscriptions auftrat.
- Das **OPC UA Server Interface Plugin** wurde auf Version **1.0.5** aktualisiert:
 - Unterstützung für Methodennodes wurde hinzugefügt, die nun über einen OPC UA Client aufgerufen werden können.
- Das **S7 Device Plugin** wurde auf Version **1.0.5** aktualisiert:
 - Eine neue Einstellung „SolveSiemensPduBug“ wurde hinzugefügt, die ein Problem bei manchen S7-Geräten behebt, die eine fehlerhafte Antwort senden, was zu Fehlern wie „specified data area does not exist“ führen kann.
- Das **Database Plugin** wurde auf Version **1.0.6** aktualisiert:
 - In der XML-Konfigurationsdatei wurde das Attribut `commandTimeout` zum `<DbConnection>`-Element hinzugefügt.
- Kleinere Bugfixes und Stabilitätsverbesserungen.
- **Script-API:**
 - Die `codabix.Node.getNodePath()`-Methode wurde implementiert, mit der der absolute Nodepfad abgerufen werden kann.
 - Die `codabix.readNodeHistoryValues()`-Methode wurde implementiert, mit der historische Werte eines Nodes abgerufen werden können.

0.10

0.10.0 RC1 (freigegeben am 21.12.2016)

- Das CoDaBix® Typsystem wurde überarbeitet:
 - Für Nodewerte und den Pfad des Nodes können Enum-Typen definiert und Validation-Expressions angegeben werden.
 - Die Typen **DateTime** und **TimeSpan** wurden hinzugefügt.
 - Der Typ „**Byte-Array**“ wurde an den Typ „**Blob**“ angeglichen:
 - Die maximale Größe eines Byte-Arrays wurde von 1000 Byte auf 500000 Byte erhöht.
 - Byte-Arrays werden effizienter serialisiert.
- Das CoDaBix® Entity Modell wurde überarbeitet:
 - Devices und Channels können nun unter Verwendung ihres Action-Nodes gestoppt und (neu-)gestartet werden.
 - Beim Erstellen eines neuen Channelnodes wird dessen Teilbaum automatisch erstellt und angezeigt.
- Es ist nun möglich, diverse HTTP- und HTTPS-Bindungen für den eingebetteten Webserver festzulegen, wenn CoDaBix® als Dienst ausgeführt wird. Dies ermöglicht Ihnen, eine Verbindung zu CoDaBix® von fremden Maschinen über eine **verschlüsselte TLS-Verbindung** herzustellen.
- Es wurde eine pluginspezifische Lizenzierung implementiert. Für jedes Plugin kann dessen Lizenz verwaltet werden.
- Es wurden Benutzerzugriffslevel pro Node implementiert, sodass ein Plugin festlegen kann, welche Nodes der Benutzer verändern darf.
- Das **RFC-1006 Device Plugin** (Version 1.0.0) wurde hinzugefügt, mit dem Sie sich zu Clients oder Servern über das RFC-1006-Protocol verbinden können.
- Das **S7 Device Plugin** wurde auf Version **1.0.2** aktualisiert:
 - Unterstützung für die Date, DateTime, Time, TimeOfDay und S5Time-Datentypen wurde hinzugefügt.
 - Wenn das Lesen einer Variable fehlschlägt, wird nun ein detaillierterer Statustext (statt nur „Good“ oder „Bad“) protokolliert und in der Webkonfiguration angezeigt.
- Das **OPC UA Server Interface Plugin** wurde auf Version **1.0.4** aktualisiert:
 - Der Standardport für neue Installationen wurde auf 4840 geändert.
- **Inkompatible Änderungen:** Das **Database Exchange Plugin** wurde in **Database Plugin** (Version 1.0.3) umbenannt:
 - Der Ordner des Plugins im Datenverzeichnis (<CodabixDataDir>) wurde von "DataBaseExchangePlugin" in "DatabasePlugin" umbenannt.
 - Die Konfigurationsdatei des Plugins wurde von "CoDaBix.DataBaseExchangePlugin.Settings.xml" in "CoDaBix.DatabasePlugin.Settings.xml" umbenannt.
- Scripts starten nun ebenfalls automatisch neu, wenn eine unbehandelte Ausnahme in der Initialisierungsphase aufgetreten ist, und nicht nur dann, wenn diese in der Callbackphase aufgetreten ist.
- **Inkompatible Änderungen der Script-API:**
 - Die Methode `timer.setImmediate()` wurde durch die Methode `timer.scheduleCallback()` ersetzt, welche effizienter ist.
 - Die Methode `timer.sleepAsync()` wurde umbenannt in `timer.delayAsync()`.
 - Die `NodeValue`-Klasse unterstützt nun `Date`-Instanzen, um mit dem CoDaBix®-`DateTime`-Typ zu arbeiten.
 - Die `type`-Eigenschaft der `codabix.Node`-Klasse enthält keine Werttypen wie `Int32` mehr. Stattdessen sind diese nun über die `valueType.typeCode`-Eigenschaft verfügbar. Die Eigenschaft `valueType.isArray` gibt dabei an, ob es sich um einen Array-Typ handelt.

- Der Typ `codabix.NodeValueStatus` (wie er z.B. in `codabix.NodeValue.status` benutzt wird) wurde in ein Interface mit einer `statusCode`- und einer `statusText`-Eigenschaft geändert.

0.10.1 (freigegeben am 27.01.2017)

- **Inkompatible Änderung: File Access Security (FAS)** wurde implementiert, um den Zugriff auf Dateien in CoDaBix® zu beschränken. Das Feature ist in den **CoDaBix®-Einstellungen** konfigurierbar.

Dieses Feature verhindert, dass ein Benutzer, der das CoDaBix®-Adminpasswort sowie lokalen oder Netzwerkzugriff auf die Webkonfiguration hat, auf beliebige Dateien zugreifen kann (vor allem wenn CoDaBix® als Dienst läuft). Standardmäßig ist das **Datenverzeichnis** für Lese- und Schreibzugriff sowie das **Installationsverzeichnis** für Lesezugriff zugelassen.

Diese Einstellung wird angewendet auf:

- Filenodes (auf die per HTTP-Access-URL oder über einen OPC UA Client zugegriffen wird) sowie Directorynodes
 - Scripts, die die `io.file`- und `io.directory`-Namensräume benutzen
 - Pfade zu CSV-Dateien, die in der XML-Konfigurationsdatei des CSV Exchange Plugins angegeben werden
- Geschwindigkeitsverbesserungen und Bugfixes.
 - Das **S7 Device Plugin** wurde auf Version **1.0.3** aktualisiert:
 - Konfigurator: Funktion zum Kopieren von Kanälen und Variablen
 - Kanäle: Unterstützung für Beschreibungen wurde hinzugefügt
 - Variablen: Unterstützung für Beschreibungen wurde hinzugefügt
 - Der Typ „Method“ für Nodes wurde hinzugefügt, über den Plugins ausführbare Befehle bei einem solchen Node registrieren können.
 - Das **Database Plugin** wurde auf Version **1.0.4** aktualisiert:
 - In der XML-Konfigurationsdatei wurden die Attribute `connectTimeout` und `maxPoolSize` zum `<DbConnection>`-Element hinzugefügt.
 - **Script-API**:
 - Die `io.file`-, `io.directory`- und `io.path`-Namensräume wurden implementiert, um Scripts den **Zugriff auf Dateien** zu ermöglichen, z. B. zum Lesen und Schreiben von Textdateien.

0.10.2 (freigegeben am 07.02.2017)

- Die Zuverlässigkeit von CoDaBix® Compact wurde verbessert.
- Es wurde ein Bug behoben, der das Lesen von Arraywerten und historischen Werten verhinderte.
- Es wurde ein Bug behoben, der zum Reset der „HistoryOptions“-Eigenschaft eines Nodes nach einem Neustart von CoDaBix® Compact führte.
- Es wurde ein Bug behoben, der verhinderte, dass das Script-Timeout auf Scriptcode in `catch`- oder `finally`-Abschnitten angewendet wird.
- Es wurde ein Bug behoben, der das Erstellen von Variablen für das OPC UA Client Device Plugin verhinderte.

0.10.4 (freigegeben am 09.02.2017)

- Das **OPC UA Client Device Plugin** wurde auf Version **1.0.2** aktualisiert:
 - Es wurde ein Bug behoben, der das Lesen und Schreiben von Variablenodes verhinderte, die in einer Foldernode erstellt wurden.

- Die Konvertierung von Arrays aus gewrappten OPC UA Werten wurde verbessert, sodass diese in CoDaBix® Nodes mit einem Arraytyp geschrieben werden können.
- Wenn mehrere Nodes gleichzeitig gelesen werden und einer davon nicht gelesen werden kann, werden die anderen Nodes nun dennoch gelesen.

0.9

0.9.0

- Die interne Datenbankstruktur wurde geändert, sodass nun jeder Node eine GUID zur Identifizierung bekommt.
- Plugins können sich nun automatisch neustarten, wenn ihre Einstellungsdatei verändert wurde.
- Es können nun Nodewerte direkt über die Webkonfiguration geschrieben werden.
- Das **Database Exchange Plugin** wurde hinzugefügt, mit dem man Nodewerte in externe MySQL- bzw. MSSQL-Datenbanken schreiben kann.
- S7-Variablen haben keine GUID mehr als Namen, sondern ihren korrekten Variablennamen.
- Das **S7 Device Plugin** kann nun Arrays (z. B. Int32-Array) lesen und schreiben.
- Das **S7 Device Plugin** liest nun nicht mehr automatisch Werte. Stattdessen wartet es darauf, dass eine externe Subscription (z. B. über einen OPC UA Client, der über das OPC UA Server Plugin verbunden ist) erstellt wird, oder bis ein synchroner Lesevorgang angefordert wird.

0.9.2

- Das **CoDaBix.AutoSubscribe**-Plugin wurde hinzugefügt, welches automatisch Subscriptions für Nodes erstellt, bei denen „Has History Values“ auf „Yes“ gesetzt ist. Damit können nun S7-Variablen automatisch regelmäßig gelesen werden, ohne dass extra eine Subscription (z. B. über einen OPC UA Client) erstellt werden muss.
- Im CoDaBix® Compact Setup kann man nun genau auswählen, welche Plugins installiert werden sollen.
- Die Webkonfiguration zeigt nun den Status für Istwerte eines Nodes an.
- Ein Bug wurde behoben, der das Verwenden des Int16-Array-Datentyps verhinderte.

0.9.3

- Ein Bug wurde behoben, der das Schreiben von UInt64-Werten verhinderte.
- Ein Bug wurde behoben, der die Verwendung von MySQL als Backend-Datenbank verhinderte.

0.9.4 (freigegeben am 07.09.2016)

- Die Webkonfiguration wurde komplett überarbeitet für bessere Optik und Bedienung. Beispielsweise ist es nun möglich, in der Webkonfiguration einen synchronen Read auf S7-Variablen durchzuführen.
- Das **S7 Device Plugin** überprüft nun automatisch regelmäßig die Verbindung zur S7, damit der Kanalstatus immer den aktuellen Verbindungsstatus zur SPS anzeigen kann.
- Das **S7 Device Plugin** kann nun SPS-Objekte lesen.
- Der Verzeichnisname des **S7 Device Plugins** im Datenverzeichnis (<CodabixDataDir>) wurde von "S7DeviceServer" zu "S7DevicePlugin" geändert, und die Konfigurationsdatei wurde von

- "CoDaBix.S7DeviceServer.Settings.xml" zu "CoDaBix.S7DevicePlugin.Settings.xml" geändert.
- Der Verzeichnisname des **Database Exchange Plugins** im Datenverzeichnis (<CodabixDataDir>) wurde von "DataBase" zu "DataBaseExchangePlugin" geändert, und die Konfigurationsdatei wurde von "CoDaBix.DataBase.Settings.xml" zu "CoDaBix.DataBaseExchangePlugin.Settings.xml" geändert.
- Das **DataBase Exchange Plugin** unterstützt nun Flankentrigger.

0.9.5 (freigegeben am 08.09.2016)

- Einige kleinere Bugs in der Webkonfiguration wurden behoben.
- Das **DataBase Exchange Plugin** wurde auf Version **1.0.1** aktualisiert:
 - Es wurden „AfterSyncActions“ hinzugefügt, die es erlauben, einen Wert in einen Node zu schreiben, nachdem die Werte des Datensatzes gesammelt wurden.

0.9.6 (freigegeben am 13.09.2016)

- Das **S7 Device Plugin** wurde auf Version **1.0.1** aktualisiert:
 - Der **S5 String** Datentyp wurde hinzugefügt, welcher kein 2-Byte-Längenpräfix verwendet.

0.9.7 (freigegeben am 14.09.2016)

- Es wurde ein Bug behoben, der das Speichern von Script Plugins verhinderte.

0.9.8 (freigegeben am 20.09.2016)

- Die Typen „Directory“ und „File“ für Nodes wurden hinzugefügt, die eine physische Datei auf dem Hostcomputer repräsentieren. Dies ermöglicht Ihnen, Dateien über einen OPC UA Client zu lesen (sobald das Feature im OPC UA Server Interface Plugin implementiert wird) oder per HTTP Access URL herunterzuladen.
- Der Plugin Manager erstellt nun automatisch eine Sicherung der Pluginkonfiguration, wenn ein Plugin erfolgreich startet. Falls der Start eines Plugins fehlschlägt, wird stattdessen die zuvor gesicherte Konfiguration verwendet, falls vorhanden.
- Das **Database Exchange Plugin** wurde auf Version **1.0.2** aktualisiert:
 - Es wurde ein Bug behoben, der zum doppelten Schreiben von Datensätzen führte, wenn derselbe Trigger in verschiedenen Datensätzen verwendet wurde.

0.9.10 (freigegeben am 05.10.2016)

- Die Script-Plugin-API wurde überarbeitet, um eine durchdachtere und stabilere API für zukünftige Veröffentlichungen bereitzustellen.
- Das **OPC UA Client Device Plugin** wurde hinzugefügt, mit dem Sie sich zu anderen OPC UA Servern verbinden und deren Variablen in CoDaBix® abbilden können.
- Das **OPC UA Server Interface Plugin** wurde auf Version **1.0.2** aktualisiert:
 - Der Zugriff auf historische Werte von CoDaBix® Nodes über HDA (Historical Data Access) wird nun unterstützt.
 - Das Übertragen von **Dateien**, die durch CoDaBix® Filenodes dargestellt werden, wird nun unterstützt.
- Die Oberfläche der Webkonfiguration wurde überarbeitet.

0.9.11 (freigegeben am 17.10.2016)

- Das **OPC UA Server Interface Plugin** wurde auf Version **1.0.3** aktualisiert:
 - Das Konfigurationsschema wurde verbessert, um dem der anderen Plugins zu entsprechen. Die Konfiguration befindet sich nun unter dem Systemnode „Interfaces“.
 - Man kann nun das Anwendungsinstanz-Zertifikat konfigurieren, sowie den „SignAndEncrypt“-Modus für eine verschlüsselte Übertragung aktivieren.
 - Es wurden einige Bugs bezogen auf die Benutzerauthentifizierung sowie das Aktualisieren von Nodes behoben.
- Das **OPC UA Client Device Plugin** wurde auf Version **1.0.1** aktualisiert:
 - Wenn mehrere Nodes zusammen auf einmal gelesen werden und ein Node davon nicht gelesen werden kann, bekommt dieser nun einen Wert mit dem Status „Bad“, wohingegen die anderen Nodes korrekt gelesen werden.
- Systemweite Fehler werden nun sofort in der Webkonfiguration angezeigt (zusätzlich zur CodabixDebugLog.txt), sodass Sie nicht mehr regelmäßig die Protokolldatei manuell öffnen und schließen müssen.

0.9.12 (freigegeben am 18.10.2016)

- Es wurde ein Problem in der Webkonfiguration behoben, das auftrat, wenn das System für die Verwendung eines Proxyservers konfiguriert wurde.

0.9.13 (freigegeben am 21.10.2016)

- Es wurde ein Bug behoben, der den Start des OPC UA Server Interface Plugins und des OPC UA Client Device Plugins sowie das Installieren von CoDaBix® Compact als Dienst verhinderte, wenn der CoDaBix® Compact Process zuvor unsauber beendet wurde.

0.9.15 (freigegeben am 03.11.2016)

- Die Zuverlässigkeit von CoDaBix® Compact wurde verbessert.
- Dateien, die durch Filenodes repräsentiert werden, können nun direkt in der Webkonfiguration über den „Write a new value“-Button heruntergeladen werden.
- Die browse-Funktion des REST/JSON Interface Plugins unterstützt nun den ttl-Parameter, um einen synchronen Lesevorgang auszuführen.

From:

<https://www.codabix.com/> - **CoDaBix®**

Permanent link:

<https://www.codabix.com/de/version-history>

Last update: **2021/10/14 17:32**