



BOSCH

Invented for life

Visual Rules

Release Informationen

Version 8.0.1

Europe:

Bosch Software Innovations GmbH
Ullsteinstrasse 128
12109 Berlin
Germany
Tel. +49 30 726112-0
info-de@bosch-si.com
www.bosch-si.de

Asia:

Bosch Software Innovations
c/o Robert Bosch (SEA) Pte Ltd
11 Bishan Street 21
Singapore 573943
Tel. +65 6571 2220
info-sg@bosch-si.com
www.bosch-si.com

Release Informationen: Version 8.0.1

Visual Rules Release Informationen, Version 8.0.1
Copyright © 2019 Bosch Software Innovations GmbH

© Bosch Software Innovations GmbH 2019. Alle Rechte vorbehalten, auch bzgl. jeder Verfügung, Verwertung, Reproduktion, Bearbeitung, Weitergabe sowie für den Fall von Schutzrechtsanmeldungen.

Inhaltsverzeichnis

Version 8.0.1	1
Modeler	1
Visual Rules Server	1
Builder	2
Version 8.0	3
Modeler	3
Visual Rules Server	3
Builder	5
Verschiedenes	6
Version 7.2.1	7
Modeler	7
Visual Rules Server	7
Builder	8
Verschiedenes	8
Version 7.2	9
Modeler	9
Visual Rules Server	9
Builder	11
Verschiedenes	11
Version 7.1	12
Modeler	12
Team Platform	12
Execution Platform	13
Builder	15
Verschiedenes	15
Version 7.0	16
Modeler	16
Team Platform	16
Execution Platform	16
Builder	18
Verschiedenes	18

Version 8.0.1

Legende

- ☆ Neue Funktionalität
- ↗ Verbesserung
- ↔ Änderung
- 🔧 Behobener Fehler
- 📦 Migration

Modeler

- 🔧 Beim Ausführen eines Regeltests im Modeler trat aufgrund eines Klassenpfad-Problems ein Fehler (*NoSuchMethodError*) auf.
- 🔧 Beim Arbeiten mit Regelprojekten, die von Verzweigungen (Branches) heruntergeladen wurden, konnten Fehler auftreten, die eine Synchronisation lokaler Ressourcen mit dem Stand auf dem Visual Rules Server verhinderten. Beispielsweise wurden lokale Änderungen in der Ansicht *Team Server Aktivitäten* nicht mehr angezeigt.
- 🔧 Bei der Authentifizierung am Visual Rules Server wurde eine Fehlermeldung ausgegeben, wenn der Server eine Antwort ohne Inhalt lieferte.
- 🔧 Eine geänderte Lese- oder Schreibberechtigung für ein Regelprojekt wurde im Regeleditor nach dem Schließen und erneuten Öffnen einer Regel nicht korrekt angezeigt. Nun wird zudem eine Meldung angezeigt, wenn eine Berechtigungsänderung eine Aktion nicht mehr zulässt.
- 🔧 Wurde beim Veröffentlichen von Rule Services versucht, eine Bibliothek in eine Regel-Bibliothek zu konvertieren (oder umgekehrt), dann kam es zur Ausgabe einer Fehlermeldung, die nicht hilfreich war.
- 📦 6.4.x/7.x nach 8.0.1: Migration der Regelmodelle notwendig, welche allerdings automatisch beim Öffnen des zugehörigen Regelprojekts im Modeler angeboten und durchgeführt wird.
- 📦 8.0 nach 8.0.1: Es ist keine Migration Ihrer existierenden Regelmodelle notwendig. Nach der Installation des Modeler 8.0.1 können Sie Ihre bestehenden Arbeitsbereiche wiederverwenden.

Visual Rules Server

- ↔ Die Version einiger Third Party Bibliotheken wurde aktualisiert, um potenzielle Sicherheitsverletzungen zu vermeiden.
- 🔧 In der Web UI des Visual Rules Servers wurden bei der Verwendung einer Oracle-Datenbank keine Benutzer/Gruppen angezeigt, sodass beispielsweise die Zuteilung von Berechtigungen für ein Source-Repository nicht möglich war.
- 🔧 Beim Aktualisieren eines bereits bestehenden Rule Service trat bei der Verwendung einer Oracle-Datenbank ein Fehler (*UniqueConstraintViolation*) auf.
- 🔧 Ein Administrator hatte keine Möglichkeit, nicht mehr benötigte Gruppen im Visual Rules Server über die Web UI zu löschen. Die Web UI wurde nun um die Anzeige von Gruppen erweitert, worin ein Administrator auch Gruppen löschen kann.

- 🔑 Die Anzeige aller beim Visual Rules Server angemeldeten Benutzer sollte in der Web UI nur einem *Visual Rules Administrator* möglich sein.
- 🔑 In einem Cluster der Build-Komponente konnten Nebenläufigkeitsprobleme auftreten, die zu einer Ausnahme und dem Abbruch des Build-Prozesses führen konnten.
- 🔑 Der Load-Balancer (Gateway) verteilte eingehende Anfragen nicht gleichmäßig auf die Clusterknoten.
- 🔑 Beim Herunterladen der Log-Datei für einen kontinuierlichen Build-Prozess trat manchmal ein Fehler auf.
- 🔑 Benutzer, denen die Rolle *Visual Rules Regelausführer* zugeordnet war, konnten den Import eines Rule Service als Visual Rules Archiv (VRA) auslösen, was zu einem Fehler in der Web UI führte.
- 🔑 Die Gesamtanzahl an Team-Aktivitäten wurde nicht korrekt angezeigt.
- 🔑 Gelöschte Release-Konfigurationen wurden weiterhin angezeigt.
- 🔑 Das Anlegen einer Verzweigung (Branch), deren Name Punkte enthält, war nicht möglich.
- 🔑 6.4.x - 7.1 nach 8.0.1: Der Execution Server und der Team Server benötigen eine Migration der Datenbank und des Identity Managements. Die Migrationsschritte sind im Visual Rules Server Benutzerhandbuch beschrieben.
- 🔑 7.2.x nach 8.0.1: Der Visual Rules Server benötigt eine Migration der Datenbank und des Identity Managements. Die Migrationsschritte sind im Visual Rules Server Benutzerhandbuch beschrieben.
- 🔑 8.0 nach 8.0.1: Beim Starten des Visual Rules Server Gateways wird in der Benutzer-Datenbanktable der Spaltentyp für die Mandanten-ID automatisch modifiziert.

Builder

- 🔑 6.4.x/7.x nach 8.0.1: Manuelle Migration ist notwendig, sofern Sie Ant-Tasks oder das "visualrules-teammaven-plugin" genutzt haben. Die alternative Verwendung von Git-Kommandos ist im Visual Rules Builder Handbuch beschrieben.
- 🔑 8.0 nach 8.0.1: Es ist keine Migration notwendig.

Version 8.0

Modeler

- ↗ OpenJDK 8 wird nun unterstützt.
- ☞ Oracle Java Runtime Environment (JRE) ist nicht mehr in der Distribution des Visual Rules Modelers enthalten. Man kann mit einer bereits installierten Oracle JRE arbeiten oder auch OpenJDK verwenden.
- ☞ Beim Veröffentlichen von Regeln wird die Authentifizierungsmethode "Form-based" angeboten, die "IM3" ersetzt. Diese Methode muss für die Veröffentlichung auf den Visual Rules Server ausgewählt werden. Die Spezifikation eines Mandanten als Rule Service Besitzer ist nicht mehr notwendig.
- ☞ Beim Ressourcotyp "HTTP Resource" wird die Authentifizierungsmethode "Identity Management Context-ID" durch "Client Token Weiterleitung" ersetzt. Als Client Token ist weiterhin die Context-ID des Bosch IoT Permissions Service zur Authentifizierung nutzbar, neuerdings aber auch ein OAuth2 Access-Token.
- 🔗 6.4.x/7.x nach 8.0: Migration der Regelmodelle notwendig, welche allerdings automatisch beim Öffnen des zugehörigen Regelprojekts im Modeler angeboten und durchgeführt wird.

Visual Rules Server

- ☆ Die Modernisierung des Visual Rules Backends ist abgeschlossen und die Komponenten *Team*, *Build* und *Execution* sind als Microservices in einem Visual Rules Server vereint.
- ☆ Die *Registry* steht als neue Serverkomponente zur Verfügung, ermöglicht die zentrale Registrierung aller Komponenten-Instanzen und vereinfacht die Kommunikation zwischen den Komponenten. Sie ist Grundlage für das Clustering von Komponenten und damit für die Umsetzung von Lastverteilung (Load Balancing), hoher Verfügbarkeit und dynamischer Skalierbarkeit des Visual Rules Servers.
- ☆ Das Gateway fungiert als Load Balancer für ein Cluster der Team-, Build- oder Execution-Komponente. Ein zusätzlicher Load Balancer wie Nginx ist für diese Cluster nicht mehr notwendig.
- ☆ Durch die Einführung der Registry vereinfacht sich das Einrichten und Konfigurieren eines Clusters der Team-, Build- oder Execution-Komponente. Beim Hinzufügen oder Entfernen eines Cluster-Knotens ist kein Neustart oder eine Neukonfiguration des Visual Rules Servers notwendig. Weitere Details, einschließlich zu beachtender Sicherheitsaspekte, finden Sie im Visual Rules Server Benutzerhandbuch.
- ☆ Der Visual Rules Server ermöglicht die Verwendung eines Authentifizierungs- und Autorisierungssystems, das auf *OAuth2* basiert und *OpenID Connect* unterstützt. *Keycloak* ist ein Open Source Identity und Access Management System, das diese Standards unterstützt und zur Evaluierung in der Visual Rules Distribution enthalten ist. *Keycloak* ermöglicht auch die Umsetzung der Mandantentrennung.
- ☆ Beim Ausführen einer Regel als SOAP/RESTful Web Service kann zur Authentifizierung auch ein OAuth2 Access-Token übermittelt werden.
- ☆ Ein Regelprojekt kann als Zip-Archiv vom Visual Rules Server heruntergeladen werden.
- ☆ In der UI werden existierende Verzweigungen eines Source-Repositories angezeigt. Zudem kann man eine neue Verzweigung erstellen, welche von einer existierenden Verzweigung oder Kennzeichnung ausgeht.
- ☆ Die Regeln eines Regelprojektes werden nun auch auf der Detailseite eines Regelprojektes angezeigt und können im Web Viewer angesehen werden.

- ☆ Informationen über Benutzer, die am Visual Rules Server angemeldet sind, werden nicht ständig beim darunterliegenden Authentifizierungs- und Autorisierungssystem angefragt, sondern bei der Benutzeranmeldung gespeichert. Die UI bietet berechtigten Personen die Möglichkeit, diese zwischengespeicherten Benutzerinformationen anzusehen und jederzeit wieder zu löschen.
- ☆ Der Visual Rules Server enthält das Maintenance Tool *vsadmin*, das insbesondere zur Migration der im Identity Management verwalteten Benutzer, Gruppen und Rollen eingesetzt wird. Die Umstellung auf ein OAuth2 basiertes Authentifizierungssystem (Keycloak) wird dadurch erleichtert.
- ↗ Die UI des Visual Rules Servers zeigt nun eine Breadcrumb an, welche es erleichtert, die aktuelle Position zu erkennen und zu übergeordneten Seiten zu navigieren.
- ↗ Beim kontinuierlichen Bauen eines Regelprojekts oder für ein Release wird der Git-Commit-Hashwert ausgegeben, um den exakten Stand des gebauten Regelprojekts spezifizieren zu können.
- ↗ Der Filterabschnitt in der UI wurde überarbeitet mit dem Ziel, die Bedienbarkeit zu verbessern und Darstellungsprobleme zu lösen. Die Anzahl aktiver Filterkriterien für eine Tabelle/Liste in der UI ist nun auch bei ausgeblendetem Filterabschnitt ersichtlich.
- ↗ Verschiedene Layout-Probleme der UI (z.B. abgeschnittene Inhalte, mangelnde Dialoggröße) wurden bearbeitet, um die Benutzerfreundlichkeit zu erhöhen.
- ↗ Die Request-ID in der Ausführungsübersicht einer Regel wird nun in der UI ohne Zeilenumbrüche dargestellt, was eine kompaktere Ansicht zur Folge hat. Generell wurde die Anzeige langer Texte in der UI überarbeitet und vereinheitlicht.
- ↗ Ein Rule Service kann nun auch von seiner Detailseite aus gelöscht werden.
- ↗ Dialoge zur Bestätigung von Löschvorgängen wurden in der UI vereinheitlicht.
- ↗ Das Editieren von Eigenschaften oder Schlüssel/Wert-Paaren wurde in der UI vereinheitlicht und ist dadurch benutzerfreundlicher.
- ↗ Fehlermeldungen des Visual Rules Servers wurden überarbeitet, sodass keine irrelevanten Informationen enthalten sind.
- ↗ Die Maintenance Tools *epadmin* und *tpadmin* wurden bezüglich der Migration von Nicht-Standard-Mandanten im Identity Management verbessert, sodass zuvor notwendige manuelle Schritte nun wegfallen.
- ↗ OpenJDK 8 wird nun unterstützt.
- ↻ Das Identity Management und der UI-Integrator sind nicht mehr Bestandteil der Visual Rules Distribution. Zur Evaluierung wird nun Keycloak als Authentifizierungssystem eingesetzt.
- ↻ Um Probleme durch das fehlende Setzen des SQL-Dialektes (entsprechend der verwendeten Datenbank-Version) zu vermeiden, gilt das zugehörige Konfigurations-Property nun nicht mehr als optional.
- ↻ Die Version einiger Third Party Bibliotheken wurde aktualisiert, um potenzielle Sicherheitsverletzungen zu vermeiden.
- ↻ Voraussetzungen für den sicheren Betrieb des Visual Rules Servers werden im Installationskapitel des Visual Rules Server Benutzerhandbuchs beschrieben.
- 🔑 Das Ausführen einer Regel als SOAP Web Service konnte zu einer ungültigen Antwort führen, in der die XML Namespace URI nicht korrekt erstellt wurde.
- 🔑 In der Build-Komponente trat bei der Verwendung einer Microsoft SQL Server Datenbank eine Ausnahme auf.
- 🔑 Wurde nach der Durchführung eines Release-Builds die Verzweigung gewechselt und erneut ein Release-Buildprozess gestartet, dann trat eine Ausnahme auf.
- 🔑 Beim Löschen einer größeren Anzahl von Ausführungen trat ein Fehler auf.

- 🔑 Bestimmte Tabellenspalten waren in der UI mit dem Sortiersymbol ausgestattet, obwohl die zugehörigen Eigenschaften nicht als Sortierkriterien möglich waren.
- 🔑 Benutzer mit bestimmten Rollen (*Visual Rules Modellierer*, *Visual Rules Regelverwalter*) hatten Zugriff auf zu wenig bzw. zu viel Funktionalität in der UI. Die rollenabhängige Anzeige von UI-Komponenten wurde überarbeitet.
- 🔑 In der UI des Visual Rules Servers kam es vereinzelt zu Darstellungsproblemen, wenn die Bildschirmgröße verkleinert wurde.
- 🔑 Das Editieren des Statistik-Levels für einen Rule Service war möglich, obwohl der Bearbeitungsmodus deaktiviert sein sollte.
- 🔑 Der Begriff "Bereitstellung" wurde in der UI nicht immer korrekt verwendet und wurde z.T. durch "Erstellung/Bearbeitung" ersetzt.
- 🔑 Bei der Vergabe von Benutzer- oder Gruppen-Berechtigungen wurden mögliche Namen erst nach dem Eintippen mehrerer Buchstaben vorgeschlagen.
- 🔑 Bei der Verwendung der Build API fand insbesondere beim Erstellen/Bearbeiten von Build-Konfigurationen keine ausreichende Validierung der Parameter statt. Bei der Erstellung eines Source- oder Maven-Repositories in der UI fehlten ebenfalls Validierungen, was beispielsweise Probleme beim Bauen von Regelprojekten nach sich ziehen konnte.
- 🔑 Der Dialog zum Editieren der Build-Konfiguration zeigte nach wiederholtem Öffnen noch veraltete Einstellungen bzw. Meldungen an. Der Dialog zum Erstellen eines Source- oder Maven-Repositories konnte aufgrund einer unnötigen Validierungsmeldung nicht sofort geschlossen werden.
- 🔑 Ein auftretender Fehler konnte in der UI zur Anzeige falscher oder keiner Werte führen, was Fehlinterpretationen ermöglichte. Zudem wurden Fehlermeldungen auch nach dem Verlassen einer Seite in der UI weiterhin angezeigt.
- 🔑 Beim Löschen eines Source-Repositories konnte eine Timeout auftreten.
- 🔑 Schreibberechtigungen von Gruppen auf einem Source-Repository wurden nicht berücksichtigt, sodass das Erstellen oder Editieren eines Regelprojekts dort nicht möglich war.
- 🔑 Die WSDL-Datei zum Aufruf einer Regel als SOAP Web-Service enthielt eine fehlerhafte URL im Attribut *location* des Elements *soap:address*.
- 🔑 6.4.x - 7.1 nach 8.0: Der Execution Server und der Team Server benötigen eine Migration der Datenbank und des Identity Managements. Die Migrationsschritte sind im Visual Rules Server Benutzerhandbuch beschrieben.
- 🔑 7.2.x nach 8.0: Der Visual Rules Server benötigt eine Migration der Datenbank und des Identity Managements. Die Migrationsschritte sind im Visual Rules Server Benutzerhandbuch beschrieben.

Builder

- ↗ OpenJDK 8 wird nun unterstützt.
- ↻ Ant-Tasks werden nicht länger unterstützt.
- 🔑 6.4.x/7.x nach 8.0: Manuelle Migration ist notwendig, sofern Sie Ant-Tasks oder das "visualrules-team-maven-plugin" genutzt haben. Die alternative Verwendung von Git-Kommandos ist im Visual Rules Builder Handbuch beschrieben.

Verschiedenes

- 🔗 Der Einsatz von Visual Rules 8.0 erfordert eine neue Lizenz, die man über support@bosch-si.com anfordern kann.

Version 7.2.1

Modeler

- ↗ Bei der Verwendung des *Call HTTP Servicetyps* kann die Konvertierung der Service-Antwort auf Ausgabeparameter (Unmarshalling), die vom Content-Type abhängig ist, übersprungen werden. Durch das Setzen der neuen (Un)marshalling-Option *vr.bind.unmarshaller.skip* ist es möglich, den rohen Antworttext des Service einem Ausgabeparameter vom Typ *Zeichenkette* zuzuweisen.
- ↗ Im Falle von Verbindungsunterbrechungen zwischen dem Visual Rules Modeler und dem Visual Rules Server wurde die Fehlerbehandlung verbessert, sodass nun aussagekräftigere Fehlermeldungen und Hinweise ausgegeben werden.
- 🔑 Die Regelsuche lieferte in manchen Fällen keine korrekten Ergebnisse.
- 🔑 Wenn man ein Regelmodell, das sich in einem auf dem Visual Rules Server veröffentlichten Regelprojekt befindet, umbenannte und die Änderung wieder zum Server sandte, führte dies zu einer Verdoppelung des Regelmodells im Source-Repository des Visual Rules Servers.
- 🔑 Beim Hinzufügen einer Abhängigkeit zu einem Regelprojekt, das zur gemeinsamen Bearbeitung auf dem Visual Rules Server veröffentlicht ist, trat ein Fehler auf.
- 🔑 Beim Veröffentlichen eines Regelprojekts als Rule Service auf den Visual Rules Server konnte während des Bauens eine Ausnahme auftreten, wenn das Regelprojekt eine Abhängigkeit zu einem mit Java 8 (oder höher) gebauten Artefakt besaß.
- 🔑 Sofern die Verbindung zum Visual Rules Server Proxy-Einstellungen erfordert, konnte beim Veröffentlichen eines Regelprojekts zur gemeinsamen Nutzung auf dem Visual Rules Server ein Fehler auftreten, da der Visual Rules Modeler fälschlicherweise auch Proxy-Einstellungen des Systems auswertete. Es sollten nur die Einstellungen in den *Benutzervorgaben* relevant sein.
- 🔑 6.4.x/7.0/7.1 nach 7.2.1: Migration der Regelmodelle ist notwendig, welche allerdings automatisch beim Öffnen eines Regelprojekts im Modeler angeboten und durchgeführt wird.
- 🔑 7.2 nach 7.2.1: Es ist keine Migration Ihrer existierenden Regelmodelle notwendig. Nach der Installation des Modeler 7.2.1 können Sie Ihre bestehenden Arbeitsbereiche wiederverwenden.

Visual Rules Server

- ↗ Das Bauen von Regelprojekten mit beliebigen Abhängigkeiten (Snapshots, Releases, Third Party) und die Veröffentlichung auf Maven-Repositories wurden überarbeitet. In diesem Zusammenhang wurde die Unterscheidung verschiedener Typen eines Maven-Repositories eingeführt (*SNAPSHOT*, *RELEASE*, *MIXED*).
- ↗ Bibliotheken, die von einem kontinuierlichen Build-Prozess resultieren, können nun auch auf einem Maven-Repository veröffentlicht werden.
- ↗ Gelöschte Einstiegsmodelle, die allerdings noch in Release- bzw. Build-Konfigurationen aufgelistet werden, werden in der Web UI als "nicht vorhanden" gekennzeichnet.
- ↗ Für die Entwicklungsversion nach einem Release-Build schlägt der Visual Rules Server nun einen Namen mit der Endung "-*SNAPSHOT*" vor, um späteren Build-Problemen vorzubeugen.
- ↗ Metriken zur Regelausführung sind nun auch für Regeln, die als SOAP Web Services ausgeführt werden, verfügbar.

- 🔑 Die Ausführung eines kontinuierlichen Build-Prozesses für ein Regelprojekt mit Abhängigkeit zu einem anderen Regelprojekt scheiterte, da die Bibliothek des erforderlichen Projekts nicht auf ein Maven-Repository veröffentlicht werden konnte.
- 🔑 In der auf Docker basierenden Evaluierungsumgebung von Visual Rules war es nicht möglich, im Modeler ein Source-Repository vom Visual Rules Server herunterzuladen.
- 🔑 Eingaben im Dialog "Neue Release-Konfiguration" wurden nicht oder fehlerhaft validiert. Zudem wurden Werte nicht immer korrekt zurückgesetzt.
- 🔑 Beim erneuten Öffnen des Dialoges "Build-Konfiguration editieren" kam es zur fehlerhaften Anzeige des Einstiegsmodells, wenn zwischenzeitlich die Verzweigung (Branch) des Regelprojekts gewechselt wurde.
- 🔑 Auf bestimmte Probleme, die beim Arbeiten mit der Teamkomponente auftraten, wurde weder in der UI noch in der Log-Datei hingewiesen.
- 🔑 6.4.x/7.0/7.1 nach 7.2.1: Der Execution Server und der Team Server benötigen eine Migration der Datenbank und des Identity Managements. Die Migrationsschritte sind im Visual Rules Server Benutzerhandbuch beschrieben.
- 🔑 7.2 nach 7.2.1: Beim Starten des Visual Rules Servers wird automatisch die Datenbanktabelle für Maven-Repositories bzgl. der Typ-Information erweitert und, sofern aus dem Namen ableitbar, ein Typ je Repository gesetzt. Bitte überprüfen Sie in der Web UI des Visual Rules Servers die konfigurierten Maven-Repositories und korrigieren Sie, falls erforderlich, den Typ.

Builder

- 🔑 6.4.x/7.0/7.1 nach 7.2.1: Manuelle Migration ist notwendig, sofern Sie das "visualrules-team-maven-plugin" genutzt haben. Die alternative Verwendung von Git-Kommandos ist im Visual Rules Builder Handbuch beschrieben.
- 🔑 7.2 nach 7.2.1: Es ist keine Migration notwendig.

Verschiedenes

- 🔑 Die Version des in der Visual Rules Distribution ausgelieferten Identity Managements (IM3) wurde auf 3.8.1 aktualisiert.

Version 7.2

Modeler

- ☆ Es ist möglich, die Abbildung eines Attributs einer Struktur auf ein JSON-Attribut zu definieren, sodass man JSON-Attribute verarbeiten kann, die nicht den Namenskonventionen für Attribute in Java entsprechen.
- ⇨ Sie können wie gewohnt mit den Ansichten "Team Server Explorer" und "Team Server Aktivitäten" arbeiten, um Regelprojekte in einem zentralen Source-Repository bereitzustellen, auch wenn jetzt die neue Team-Komponente des Visual Rules Servers die Versionskontrolle und die Bearbeitung durch mehrere Team-Mitglieder verwaltet. Allerdings muss ein Repository zunächst in den lokalen Arbeitsbereich heruntergeladen werden, damit enthaltene Projekte bearbeitet werden können. Die Ansicht "Team Server Explorer" wurde um diese Funktionalität erweitert.
- ⇨ Java Runtime Environment (JRE) wurde auf 8 Update 181 aktualisiert.
- ⇨ Die Version einiger Third Party Bibliotheken wurde aktualisiert, um potenzielle Sicherheitsverletzungen zu vermeiden.
- 🔑 Beim Importieren von XML-Typen (XSD-Import) traten Fehler auf, wenn die Log-Datei von Velocity nicht erstellt werden konnte. Zudem führten Umlaute in XSD-Dateien zu Encoding-Problemen.
- 🔑 Nach der Ausführung einer Testsuite wurde keine Warnung angezeigt, obwohl enthaltene Tests Warnungen aufwiesen.
- 🔑 Eine missverständliche Fehlermeldung wurde zurückgegeben, wenn Sie versucht haben, ein nicht vorhandenes Projekt über den Import-Assistenten vom Team Server zu importieren.
- 🔑 6.4.x/7.x nach 7.2: Migration der Regelmodelle ist notwendig, welche allerdings automatisch beim Öffnen eines Regelprojekts im Modeler angeboten und durchgeführt wird.

Visual Rules Server

- ☆ Der Visual Rules Server mit den Komponenten "Gateway", "Execution", "Team" und "Build" löst den Visual Rules Execution Server und den Visual Rules Team Server ab. Die verschiedenen Server-Funktionalitäten stehen nun über eine einzige Web UI zur Verfügung und werden im gemeinsamen Visual Rules Server Benutzerhandbuch beschrieben.
- ☆ Die neue Team-Komponente verwaltet die Versionskontrolle und Teamzusammenarbeit und stellt ein zentrales Source-Repository zur Verfügung, auf das über ein Git-Protokoll zugegriffen werden kann. Sie ist eine Zwölf-Faktoren-App, die Cloud-fähig und skalierbar ist.
- ☆ Die neue Build-Komponente ermöglicht das Bauen eines Regelprojekts zur Entwicklungszeit als auch das Bauen mehrerer Regelprojekte für ein Release. Die entstehenden Regelbibliotheken mit den ausführbaren Rule Services können anschließend in einer Ausführungsumgebung bereitgestellt oder auch in einem Maven-Repository installiert werden. Sie ist eine Zwölf-Faktoren-App, die Cloud-fähig und skalierbar ist.
- ☆ Defizite der Execution Server Web UI wurden analysiert und eine neue Informationsarchitektur für den Visual Rules Server entwickelt. Auf dieser Basis entstand ein neues Layout und Design der Visual Rules Server Web UI.
- ↗ Die Verwaltung von Regelprojekten und das Starten von Build-Prozessen ist in der Web UI stärker integriert, was den Entwicklungszyklus vereinfacht.

- ↗ Es besteht keine Notwendigkeit, eine Konfiguration für einen kontinuierlichen Build manuell anzulegen. Nachdem Änderungen an einem gemeinsamen Regelprojekt gemacht wurden, wird eine Build-Konfiguration, sofern noch keine existiert, angelegt und ein Build-Prozess angestoßen.
- ↗ Die Web UI bietet eine Übersicht aller gesperrten Projekt-Ressourcen und die Möglichkeit, mehrere Ressourcen auf einmal zu entsperren.
- ↗ Wenn Sie ein Attribut einer Struktur im XML/JSON-Format angeben und dieses Attribut in der modellierten Zielstruktur nicht vorhanden ist, wird die Verarbeitung nicht abgebrochen, sondern dieses Attribut wird übersprungen. Dies macht die XML/JSON-Verarbeitung robuster.
- ↗ Im Web Viewer des Visual Rules Servers kann man verschachtelte Regeln öffnen, auch wenn sie in anderen Regelbibliotheken definiert sind.
- ↗ Die Homepage der Web UI wurde überarbeitet. Für jede Server-Komponente werden Lizenzinformationen angezeigt und man kann sehen, ob eine Komponente konfiguriert und verfügbar ist. Die Metriken bzgl. Server Uptime und Speicherverbrauch werden nicht mehr angezeigt.
- ↗ Die Verwendbarkeit der Tabellenspaltensortierung wurde verbessert, indem nun klar angezeigt wird, welche Spalten sortierbar sind und welche Sortierrichtung aktuell aktiv ist.
- ↗ Ein gemeinsames, zentrales Audit-Log für alle Serverkomponenten steht zur Verfügung, dessen Einträge auch gefiltert werden können.
- ↗ Zur Vereinfachung der Visual Rules Server-Konfiguration wurden die Konfigurations-Properties der einzelnen Backend-Komponenten zusammengeführt, wobei sich die Namen bestimmter Properties geändert haben. Zudem ist nun nur noch eine Konfigurationsdatei für den Visual Rules Server nötig, welche von allen Server-Komponenten benutzt wird.
- ↗ Ein Konfigurations-Property ermöglicht es Ihnen, das Überschreiben einer Release-Version eines Rule Service zu verhindern.
- ↗ Es wurde die Dokumentation verbessert, wie Sie den Log-Level für Aktionen vom Typ "Log Eintrag schreiben" konfigurieren können.
- ↔ Der Applikationsserver WildFly 11 wird unterstützt. Der Support für den Applikationsserver JBoss 7.1 wurde beendet.
- ↔ Der Support für die Datenbank MySQL wurde auf Version 5.7 aktualisiert und für Version 5.5 beendet.
- ↔ Wenn Sie beim Einsatz eines SOAP oder RESTful Web Service zur Regelausführung als Version *LATEST* (oder keine Version) angeben, dann wird die neueste *Release*-Version des Regelmodells verwendet. Eine Version mit der Endung `-SNAPSHOT` wird nur dann verwendet, wenn es keine Release-Version gibt.
- ↔ Der Visual Rules Server bietet keine *integrierte* Betriebsart mehr an, d.h. die Team- Build- und Execution Web-UIs können nicht in den UI-Integrator eingebettet werden. Jedoch kann die Web UI des Visual Rules Servers als Anwendung im UI-Integrator konfiguriert werden.
- ↔ Der Visual Rules Server verwendet dasselbe Rollenkonzept wie der Execution Server. Eine Rolle wurde hinzugefügt (*Visual Rules Modellierer*) und die anderen Rollen wurden umbenannt.
- ↔ Der Visual Rules Server ermöglicht die Zuordnung von Berechtigungen auf Source-Repository-Ebene als auch auf Regelprojektebene (aber nicht für spezifische Ressourcen innerhalb eines Regelprojekts oder für Build-Prozesse). Die Zugriffsrechte können Benutzern und Gruppen gewährt werden (aber nicht Mandanten). Die explizite Berechtigung "*Keine Zugriffsrechte*" ist nicht mehr verfügbar.
- ↔ Für die globale Build-Konfiguration stehen Konfigurations-Properties zur Verfügung, um optional eine andere Java- und Maven-Installation anzugeben, die während des Build-Prozesses verwendet werden sollen. Zusätzlich ermöglicht die Web UI die Spezifikation von Maven-Repositories, in denen die gebaueten Regelbibliotheken bereitgestellt werden können.

- ↪ Der Begriff *Rule Service* wurde für die Ausführung von Regeln auf dem Visual Rules Server wieder eingeführt, wohingegen der Begriff *Regelmodell* eher während des Designs von Regeln Verwendung findet. In einige Web UI Seiten kommt diese Änderung zum Tragen.
- ↪ Die Version einiger Third Party Bibliotheken wurde aktualisiert, um potenzielle Sicherheitsverletzungen zu vermeiden.
- 🔑 Statistiken wurden beim Löschen von Regelausführungen nicht gelöscht.
- 🔑 Die Formatierung von Zahlen wurde in der Web UI nicht internationalisiert.
- 🔑 Aktionen einer aufgerufenen Regel wurden im Element "Ablaufregel aufrufen" nicht angezeigt, wenn die aufrufende Regel in der Web UI geöffnet wurde.
- 🔑 Es wurde ein falscher HTTP Status-Code zurückgegeben, wenn eine Regel mit "application/xml" als *Accept* HTTP-Header ausgeführt wurde und ein Fehler auftrat.
- 🔑 Das Bauen und Testen eines Regelprojekts mit einer Regel, die einen Service vom Typ *Call HTTP Service* und *JSON* verwendet, führte zu einem Fehler (*java.lang.NoClassDefFoundError*).
- 🔑 6.4.x/7.x nach 7.2: Der Execution Server und der Team Server benötigen eine Migration der Datenbank und des Identity Management. Die Migrationsschritte sind im Visual Rules Server Benutzerhandbuch beschrieben.

Builder

- ↪ Wegen einer Änderung der zugrundeliegenden Technologie wurde das "*visualrules-team-maven-plugin*" entfernt. Die Verwendung von Git wird zum Check-Out/In und zum Erstellen einer Kennzeichnung (Tag) empfohlen.
- ↪ Die Ant-Tasks sind veraltet und werden nicht mehr Teil des Visual Rules Builder 8.0 sein.
- ↪ Die Version einiger Third Party Bibliotheken wurde aktualisiert, um potenzielle Sicherheitsverletzungen zu vermeiden.
- 🔑 6.4.x/7.x nach 7.2: Manuelle Migration ist notwendig, sofern Sie das "*visualrules-team-maven-plugin*" genutzt haben. Die alternative Verwendung von Git-Kommandos ist im Visual Rules Builder Handbuch beschrieben.

Verschiedenes

- ↪ VR Runtime unterstützt JRE 1.4 nicht mehr. Die Mindestversion der JRE für die Regelausführung ist 1.7. Zusätzlich wurde die Abhängigkeit von VR Runtime zu *org.apache.xmlbeans* entfernt und die Abhängigkeit zu *commons-lang* wurde durch die Abhängigkeit zu *commons-lang3* ersetzt.
- ↪ Die Visual Rules Distribution beinhaltet keine Verzeichnisse für den *Execution Server* oder den *Team Server* mehr. Aber nun enthält sie ein Verzeichnis für jede Visual Rules Server-Komponente (*Gateway*, *Execution*, *Team* und *Build*).
- 🔑 Die Bibliothek *vr-execution-core-starter-7.1.0.jar* fehlte in der Execution Core Distribution (*mvn-repo*).

Version 7.1

Modeler

- ☆ Modeler unterstützt Linux als Betriebssystem (siehe Systemanforderungen).
- ☆ Ein Fahrzeug-Konfigurator Beispiel veranschaulicht die Integration von Regeln in eine Geschäftsanwendung.
- ↗ Verbesserte Modeler Distribution (inkl. Kurzanleitung)
- ↔ Java Runtime Environment (JRE) wurde auf 8 Update 151 aktualisiert.
- ↔ Die Version einiger Third Party Bibliotheken wurde aktualisiert, um potenzielle Sicherheitsverletzungen zu vermeiden.
- 🔧 Die JSON-Erweiterung konnte nicht als Projektabhängigkeit hinzugefügt werden.
- 🔧 Bei der Verwendung der DBQuery-Aktion trat eine Ausnahme auf, wenn die WHERE-Bedingung des SQL SELECT-Befehls zwei oder mehr aufeinanderfolgende Bindestriche in Anführungszeichen enthielt.
- 🔧 Die Namen von Ressourcen wurden nicht validiert.
- 🔧 Debug-Meldungen wurden für einen Service vom Typ *Call HTTP Service* nicht angezeigt, auch wenn ein Regeltest mit dem Flag "-l debug" ausgeführt wurde.
- 🔗 6.4/7.0 nach 7.1: Es ist keine Migration Ihrer existierenden Regelmodelle notwendig. Nach der Installation des Modeler 7.1 können Sie Ihre bestehenden Arbeitsbereiche wiederverwenden.

Team Platform

- ↗ Die Datenbank MySQL 5.7 wird nun unterstützt.
- ↔ Die Version des in der Visual Rules Distribution ausgelieferten Identity Management (IM3) wurde auf 3.8.0 aktualisiert.
- ↔ Die Version des in der Visual Rules Distribution ausgelieferten UI Integrator wurde auf 1.4.1 aktualisiert.
- ↔ Maven 3.3 und 3.5 werden nun unterstützt. Der Support für Version 3.1 wurde beendet.
- ↔ Die Version einiger Third Party Bibliotheken wurde aktualisiert, um potenzielle Sicherheitsverletzungen zu vermeiden.
- 🔧 Regelprojekte wurden bereitgestellt, auch wenn während eines Builds Testfehler auftraten.
- 🔧 Die Anzeige von Debug-Meldungen in der Konsolenausgabe eines Buildprozesses wurde unterdrückt, auch wenn das Debug-Flag "-X" in den Maven-Argumenten der Buildprozess-Konfiguration gesetzt wurde.
- 🔗 6.4/7.0 nach 7.1: Es ist keine Migration notwendig und Sie können die gleiche Konfiguration verwenden. Sie müssen nur die war-Dateien des Team Server 7.1 bereitstellen (Wenn Sie die war-Dateien des vorherigen Team Server angepasst haben, dann sollten Sie die neuen war-Dateien ebenfalls anpassen).

Execution Platform

- ☆ Web Viewer für bereitgestellte Ablaufregeln und Entscheidungstabellen ist nun verfügbar. Die Kontextpalette einer Regel kann ebenso eingeblendet werden. Sie können verschachtelte Regeln aus derselben Bibliothek öffnen.
- ☆ Das Kommandozeilen-Maintenance Tool (epadmin) steht für die automatische Migration von Version 6.4.x auf 7.1 zur Verfügung.
- ☆ Automatisches Überschreiben von Ressourcen beim Bereitstellen von Regelbibliotheken: Wenn eine bereits vorhandene Regelbibliothek in derselben Version bereitgestellt wird, werden alle Überschreibungen von Ressourcen beibehalten.
- ☆ Rezepte für die Bereitstellung des Execution Server auf Amazon Web Services (AWS) und für die Nutzung des AWS EC2 Container Services (ECS) werden zur Verfügung gestellt.
- ☆ Ein Fahrzeug-Konfigurator Beispiel veranschaulicht die Integration von Regeln in eine Geschäftsanwendung.
- ↗ Es ist möglich, mehrere Regelmodelle auf einmal zu löschen.
- ↗ Der Execution Server verwendet einen verteilten Cache für Cluster-Umgebungen.
- ↗ Nichtauflösbare Abhängigkeiten einer Regelbibliothek werden jetzt als Warnung angezeigt.
- ↗ Die Datenbank MySQL 5.7 wird nun unterstützt.
- ↗ Der Execution Core bietet die Möglichkeit, nach Regeln zu suchen, die eine Service-Schnittstelle implementieren.
- ↗ Das Arbeiten mit *Überschreibungen von Ressourcen* wurde in der Execution Server UI verbessert. Zusätzliche Filter wurden hinzugefügt (z.B. um nur diejenigen Ressourcen aufzulisten, die von einer speziellen Server-Ressource überschrieben oder von einem speziellen (aufrufenden) Regelmodell verwendet werden).
- ↗ Die *Audit Log*-Ansicht bietet einen neuen Filter an, um nur diejenigen Audit-Ereignisse, die mit einer speziellen Regelbibliothek verknüpft sind, aufzuführen.
- ↗ Informationen zur Version des Execution Server werden in der UI angezeigt.
- ↗ Die Geschwindigkeit der Regelausführung wurde verbessert.
- ↗ Das Zwischenspeichern (Cachen) von Authentifizierungsinformationen kann konfiguriert werden und Sie können die Gültigkeitsdauer von Benutzeranmeldedaten beeinflussen.
- ↗ Fehlermeldungen für den Import eines fehlerhaften Visual Rules Archivs (VRA) wurden verbessert.
- ↗ Die Startseite der Execution Server UI wurde verbessert: besseres Layout der Tabelle *Top Regelausführungen* und die Anzeige der Rolle des Benutzers, der aktuell eingeloggt ist. Lizenzinformationen (inkl. Ablaufwarnung) werden nun auch angezeigt.
- ⇄ Die Version des in der Visual Rules Distribution ausgelieferten Identity Management (IM3) wurde auf 3.8.0 aktualisiert.
- ⇄ Die Version des in der Visual Rules Distribution ausgelieferten UI Integrator wurde auf 1.4.1 aktualisiert.
- ⇄ Die Version einiger Third Party Bibliotheken wurde aktualisiert, um potenzielle Sicherheitsverletzungen zu vermeiden.
- ⇄ Das Format der JSON/XML-Eingabe für eine Regelausführung unterscheidet sich nicht, wenn die Regel einen oder mehrere Eingabeparameter hat.

- ↻ UTC wird nun einheitlich als Zeitformat in der Execution Platform genutzt, was auch für die REST API gilt.
- 🔑 In einer Cluster-Umgebung wurden bestimmte Metriken auf der Startseite der Execution Server UI nicht korrekt angezeigt.
- 🔑 In einer Cluster-Umgebung konnten Ausnahmen auftreten, wenn verschiedene Cluster-Knoten dieselbe Eigenschaft einer Server-Ressource aktualisierten.
- 🔑 In der Ansicht *Ausführungen* war es möglich, einen ungültigen Zeitraum als Filter für den Zeitpunkt der Ausführung anzugeben.
- 🔑 Der Export eines Visual Rules Archivs (VRA) mit fehlenden Abhängigkeiten war nicht möglich und verursachte eine nicht aussagekräftige Fehlermeldung.
- 🔑 Einige Probleme, die WSDL bzw. WADL-Dateien betrafen, wurden behoben: Regelpfade enthielten keine absoluten URLs und die Kodierung war nicht korrekt.
- 🔑 Der WSDL/WADL-Import funktionierte nicht für Regeln, die in einem Regelpaket enthalten waren.
- 🔑 Die "neueste" Version eines Rule Service entsprach nach der Bereitstellung auf dem Execution Server nicht der aktuellen Version.
- 🔑 Der Start des Execution Servers schlug fehl, wenn die IM3-Authentifizierung ein Kennwort mit Sonderzeichen enthielt.
- 🔑 Der REST-Endpunkt für Metriken war nicht nur Benutzern mit der Rolle *Operator* zugänglich.
- 🔑 Das Anzeigen und Herunterladen von WSDL-Dateien war mit JBoss WildFly nicht möglich.
- 🔑 Der REST-Endpunkt für Bibliotheken und der REST-Endpunkt für Regelbibliotheken verwendeten unterschiedliche IDs für ihre Instanzen.
- 🔑 Der Rule Execution Client unterdrückte keine Fehler bei ungültigen Eingabedaten.
- 🔑 Die Aktivierung von Regelstatistiken im Modeler (Standardfall) wurde im Execution Server nicht berücksichtigt.
- 🔑 Die Execution Server UI zeigte in manchen Fällen Umlaute nicht korrekt an (Kodierungsproblem).
- 🔑 Das Zurücksetzen eines Filters verursachte eine leere Tabelle in der UI und ein erneutes Laden der Seite war erforderlich.
- 🔑 Beim Sortieren einer Tabelle in der Execution Server UI wurde die Seitengröße auf die ursprüngliche Größe zurückgesetzt.
- 🔑 Beim Veröffentlichen einer neuen Regelbibliothek wurde der Bereitstellungszeitpunkt nicht aktualisiert.
- 🔑 Es gab doppelte Einträge für den Filter des Audit-Protokolls.
- 🔑 REST-Endpunkte des Execution Server lieferten in Fehlerfällen Stacktraces mit.
- 🔑 Ein XXE-Angriff war beim Import eines Visual Rules Archive (VRA) möglich.
- 🔑 6.4 nach 7.1: Der Execution Server benötigt eine Migration der Datenbank und des Identity Management. Die Migrationsschritte sind im Execution Server Benutzerhandbuch beschrieben.
- 🔑 7.0 nach 7.1: Es ist keine Migration notwendig, aber Sie müssen die alte war-Datei durch die war-Datei des Execution Server 7.1 ersetzen und auch die war-Datei des VR Gateway bereitstellen.

Builder

- ↔ Maven 3.3 und 3.5 werden nun unterstützt. Der Support für Version 3.1 wurde beendet.
- ↔ Die Version einiger Third Party Bibliotheken wurde aktualisiert, um potenzielle Sicherheitsverletzungen zu vermeiden.
- 🔑 Regelprojekte wurden bereitgestellt, auch wenn während eines Builds Testfehler auftraten.
- 🔑 Das Bauen eines Regelprojekts erzeugte Warnungen für fehlende POM-Dateien.
- 🔑 6.4/7.0 nach 7.1: Es ist keine Migration notwendig.

Verschiedenes

- ☆ Eine neue Anwendung (Gateway) wurde eingeführt, die Benutzeroberflächen für Web Viewer und Administration bereitstellt.
- ☆ Eine auf Docker basierende Evaluierungsumgebung für Visual Rules liefert ein Rezept für Docker-Operationen.
- ↔ Visual Rules wird als einzelnes Archiv, das alle Komponenten enthält, ausgeliefert. Die Visual Rules Distribution enthält auch das ehemalige Evaluation Bundle.

Version 7.0

Modeler

- ☞ Der Modeler wird mit Eclipse der Version 4.6 (Neon) installiert.
- ☞ Java Runtime Environment (JRE) wurde auf 8 Update 131 aktualisiert.
- 🔗 Es ist keine Migration Ihrer existierenden Regelmodelle notwendig. Nach der Installation des Modeler 7.0 können Sie Ihre bestehenden Arbeitsbereiche wiederverwenden.

Team Platform

- ↗ Die Datenbank Microsoft SQL Server 2016 wird nun unterstützt.
- ↗ Der Support von Applikationsservern wurde auf Tomcat 8, JBoss 7 und WildFly 10 aktualisiert. Beachten Sie, dass WildFly 10 vom Identity Management nicht unterstützt wird.
- ↗ Der Web Browser Microsoft Edge wird nun unterstützt.
- ☞ Der Support für den WebSphere Applikationsserver wurde beendet.
- ☞ WebModeler und WebViewer sind nicht mehr länger Bestandteil der Visual Rules Distribution.
- 🔗 Es ist keine Migration notwendig und Sie können die gleiche Konfiguration verwenden. Sie müssen nur die war Dateien des Team Server 7.0 bereitstellen (Wenn Sie die war Dateien des Team Server 6.4.x angepasst haben, dann sollten Sie die neuen war Dateien ebenfalls anpassen).

Execution Platform

- ↗ Die Execution Platform wurde mit einer modernen Architektur auf der Basis aktueller Technologien (Spring 4, Hibernate, Angular 2) neu implementiert, während sie überwiegend kompatibel blieb zu vorherigen Versionen (z.B. Regelausführungs-API der VR-Runtime, Bereitstellung von Regelbibliotheken, Web Service Beschreibungen). Sie kann mit einem Visual Rules Modeler, Team Server und Builder der Version 6.4.x kombiniert werden.
- ↗ Der Execution Server ist eine Zwölf-Faktoren-App, die Cloud-fähig und skalierbar ist. Rezepte für Docker und die Bereitstellung in Cloud Foundry und der Bosch IoT Cloud sind Bestandteil der Dokumentation.
- ↗ Der Execution Server wurde eine eigenständige Spring-Boot-Applikation, die als ausführbare WAR-Datei mit eingebettetem Tomcat verfügbar ist. Alternativ kann er wie bisher als Webarchiv auf einem Applikationsserver bereitgestellt werden.
- ↗ Die Initialisierung der Datenbank und des Identity Management laufen vollständig automatisiert ab, was das epadmin Maintenance Tool und den Installationsassistenten überflüssig machte.
- ↗ Die Daten werden für die Mandantenfähigkeit nicht mehr länger durch verschiedene Datenbankschemas getrennt, sondern mittels eines Diskriminators. Datenbankadministrator(DBA)-Rechte sind nicht mehr erforderlich, um eine Datenbank für eine mandantenfähige Umgebung aufzusetzen.
- ↗ Größere Flexibilität beim Aufsetzen eines Authentifizierungssystems, da mehrere Optionen unterstützt werden: keine Authentifizierung, Identity Management (On-Premises) oder Bosch IoT Permissions Service (Cloud).

- ↗ Größere Flexibilität beim Aufsetzen des Systems bzgl. Mandantenfähigkeit und Integration der Web UI, da es mehrere Möglichkeiten gibt: nur ein Mandant oder mehrere Mandanten, eigenständig oder integriert in den UI Integrator, der eine nahtlose Web UI für alle Komponenten bietet.
- ↗ Moderne, ansprechende Benutzeroberfläche mit verbesserter Bedienbarkeit.
- ☆ Ein interaktiver REST-Client zur Ausführung einer Regel ermöglicht das Testen von Regeln innerhalb der Web UI des Execution Server.
- ☆ Das Audit-Protokoll ist nun auch in der UI sichtbar, wo Sie die Möglichkeit haben, Konfigurationsänderungen und Zugriffsversuche zu verfolgen.
- ☆ Der Execution Server stellt nun Metriken bereit, die mittels JMX, Prometheus, Graphite und Grafana zur Überwachung des Betriebs eingesetzt werden können.
- ☆ Eine Startseite der Web UI zeigt eine Auslastungsübersicht, wie die Verteilung von Regelausführungen über einen Zeitraum hinweg, an.
- ↗ Die Geschwindigkeit der Regelausführung wurde erhöht, unabhängig von der Ausführungsart (SOAP oder REST) und den Authentifizierungsmethoden.
- ↗ Das Überschreiben von Ressourcen zur Laufzeit ist nun komfortabler: Ressourcen, die auf dem Execution Server definiert sind, werden global verwaltet und sind wiederverwendbar.
- ↗ Der Execution Server liefert eine optimierte REST API für sämtliche Verwaltungsaufgaben, die in der Web UI ausführbar sind, mit. Alle Verwaltungsoperationen können durch die REST API automatisiert werden.
- ↗ Die Execution Core Java API wurde neu strukturiert und vereinfacht. Sie stellt eine Fluent Java API zur Konfiguration, der Verwaltung von Bibliotheken, Ressourcen, Regeln und deren Ausführungen dar.
- ↗ Die Dokumentation der REST API des Execution Server ist zur Swagger OpenAPI Spezifikation 2.0 konform, welche heruntergeladen und in Postman importiert werden kann.
- ↗ Die Navigation innerhalb des Online-Benutzerhandbuchs des Execution Server und dessen Lesbarkeit wurden verbessert.
- ↗ Die Datenbank Microsoft SQL Server 2016 wird nun unterstützt.
- ↗ Der Support von Applikationsservern wurde auf Tomcat 8, JBoss 7 und WildFly 10 aktualisiert. Beachten Sie, dass WildFly 10 vom Identity Management nicht unterstützt wird.
- ↗ Der Web Browser Microsoft Edge wird nun unterstützt.
- ⇨ Feingranulare Zugriffsrechte (ACLs) werden nicht mehr länger angeboten, da sie die Regelausführungszeit negativ beeinflussen. Der Zugriff kann durch die Zuordnung von Rollen bzw. Mandanten kontrolliert werden.
- ⇨ Die Datenmigration von früheren Versionen des Execution Server ist auf Regelbibliotheken beschränkt. Zugriffsrechte (ACLs), Konfigurationen eines Regelmodells (Metadaten, Überschreibungen von Ressourcen, Aktivierung, etc.) und Daten des Audit-Protokolls können nicht migriert werden.
- ⇨ Der Execution Core bietet keinen Modus an, um Mandantenfähigkeit auf Basis des Identity Managements zu erhalten.
- ⇨ Der Support für den WebSphere Applikationsserver wurde beendet.
- ⇨ Das Element *effectiveDate* wird für die Weiterleitung einer generischen SOAP-Anfrage zu einem Regel-Web Service auf der Basis seiner *validTo/validFrom* Eigenschaften nicht länger unterstützt.
- ⇨ Das *Atom Syndication Format* steht nicht mehr als Rückgabeformat des Execution Server für einen RESTful Web Service zur Verfügung.

- ↳ Der Execution Server benötigt eine Migration der Datenbank und des Identity Management. Die Migrationsschritte sind im Execution Server Benutzerhandbuch beschrieben.
- ↳ Die API des Execution Core 7.0 hat sich geändert. Bestehende Implementierungen, die den Execution Core nutzen, müssen entsprechend angepasst werden. Die neue Java API ist im Execution Core Entwicklerhandbuch beschrieben.

Builder

- ↳ Es ist keine Migration notwendig.

Verschiedenes

- ↗ Die Auslieferungspakete des Produkts wurden optimiert, und ein Dokumentationspaket wurde hinzugefügt. Die Release-Informationen aller Visual Rules Komponenten wurden in einem Dokument zusammengefasst, und deren Lesbarkeit wurde verbessert.
- ↻ Die Batch Platform ist nicht mehr länger Bestandteil der Visual Rules Distribution.
- ↳ Der Einsatz von Visual Rules 7.0 erfordert eine neue Lizenz, die man über support@bosch-si.com anfordern kann.