



**BOSCH**

Invented for life

## Visual Rules

## Feature-Liste

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

## **Feature-Liste: Version 8.0.1**

Visual Rules Feature-Liste, 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

<b>Visual Rules Modeler</b> .....	<b>1</b>
Grafischer Regeleditor .....	1
Regelmanagement .....	1
Elemente der Ablaufregel .....	2
Elemente der Entscheidungstabelle .....	2
Aktionen & Services .....	2
Funktionen .....	2
Daten .....	2
Regelausführung .....	3
Testen der Regeln .....	3
Debugging .....	3
Java Code Generator .....	3
APIs und Erweiterungen .....	3
Onlinehilfe und Dokumentation .....	3
Import .....	3
Zustandsabläufe .....	4
<b>Visual Rules Server: Team-Komponente</b> .....	<b>5</b>
Source-Repository .....	5
APIs und Erweiterungen .....	5
Wartung .....	5
<b>Visual Rules Server: Build-Komponente</b> .....	<b>6</b>
Build-Verwaltung .....	6
APIs und Erweiterungen .....	6
<b>Visual Rules Execution Platform</b> .....	<b>7</b>
Visual Rules Server: Execution-Komponente .....	7
APIs und Erweiterungen .....	3
Execution Core .....	7
<b>Visual Rules Databaselntegrator</b> .....	<b>8</b>
<b>Visual Rules Builder</b> .....	<b>9</b>
<b>Technische Limitierungen</b> .....	<b>10</b>
<b>Datenschutz</b> .....	<b>12</b>
Allgemein .....	12
Visual Rules Modeler .....	12

Visual Rules Server (Team und Build) .....	13
Execution Platform .....	14
Benutzerverwaltung .....	15
IoT Permissions .....	15
<b>Über Bosch Software Innovations .....</b>	<b>16</b>

## Visual Rules Modeler

### Grafischer Regeleditor

- Grafischer Editor zur Bearbeitung von Ablaufregeln und Entscheidungstabellen mit Drag&Drop
- Regelkontext Palette ermöglicht Schnellzugriff auf regelspezifische Daten und Tests
- Hover Palette zum schnellen Einfügen von Regелеlementen
- Editieren der Regelausdrücke und Beschreibungen direkt in der grafischen Darstellung
- Sofortige Validierung bei Eingabe und Syntaxhervorhebung
- Schnellzugriff auf verfügbare Daten, Operatoren und Funktionen durch Code Assistent
- Automatische Validierung und Konsistenzprüfungen der Regelmodelle
- Regel-Konzeptionsmodus (nur Beschreibung und/oder Notizen von Regeln bearbeiten)
- Statistikanzeige (Anzahl Aufrufe, prozentuale Verteilungen, Zeiten) im Regeleditor
- Dynamisches Layout und Zoomfunktion
- Unbegrenzt Zurücknehmen/Wiederholen von Änderungen
- Hyperlinks und Tooltips bei Ausdrücken, Daten, Funktionen etc.
- Automatisches Anlegen von Datenelementen, Typen, Aktionen usw. bei Erstverwendung
- Automatisches Anpassen der Regeln nach einer Umbenennung von Elementen
- Anfügen von Notizen an Regelelemente
- Kommentare und Verknüpfungen zu externen Dokumenten
- Verkleinerte Ansicht zur schnellen Navigation in großen Regeln
- Ein- und Ausklappen von Regelzweigen zur Erhöhung der Übersichtlichkeit
- Mehrseitige Ausdrucke der Ablaufregeln und Entscheidungstabellen
- Exportieren der Regeln als JPG, PNG oder BMP
- Exportieren von Regelartefakten als OSGi-kompatible JAR Dateien
- Assistent für die Aufteilung großer Ablaufregeln in kleinere Logikteile
- Assistent zur Konvertierung von Entscheidungstabellen in Ablaufregeln und zurück
- Erstellung von Ablaufregeln auf Basis von Java Schnittstellen

### Regelmanagement

- Hierarchische Strukturierung von Regeln und Regelmodellen durch Regelpakete
- Gleichzeitiges Bearbeiten von mehreren Regelmodellen
- Import von Regelpaketen in anderen Regelpaketen zur Wiederverwendung von Regeln, Datentypen, Funktionen, Aktionstypen und Services
- Regel Browser zum schnellen Auffinden und Öffnen von Regeln
- Kontextbezogene Suche nach Namen, Beschreibungen, Notizen oder Ausdrücken
- Interaktive Darstellung der Abhängigkeiten von Regeln und Regelpaketen
- Suche nach Verwendungen einer Regel, eines Datenelements, einer Aktion oder anderer Elemente
- Grafischer Vergleich und Abgleich zwischen Regeln und Regel Paketen (inklusive Versionsgeschichte)
- Notizen für alle Elemente eines Regelmodells möglich
- Kommentare und Verlinkungen zu externen Dokumenten für alle Regelelemente
- Export einer Regelmodellokumentation in HTML, als PDF oder als MS Word-Dokument
- Verwaltung der Regelmodelle in einem Versionierungssystem oder im Visual Rules Server

- Beliebige in Eclipse integrierte Versionierungssysteme einsetzbar
- Vollständige Maven Unterstützung, um dynamisch Abhängigkeiten zwischen Regelprojekten und anderen Artefakten aufzulösen
- Versionierungsschema für die Verwaltung von mehreren Versionen desselben Regelprojekts mit jeweils unterschiedlichen Abhängigkeiten
- Darstellung von Verwendungen von Regeln und Regelementen anhand einer Matrix

## Elemente der Ablaufregel

- Entscheidung mit zwei Ausgängen (wenn – dann – sonst)
- Entscheidung mit mehr als zwei Ausgängen
- Aktionsauslösung zur Definition der Ergebnisse der Entscheidungen
- Zuweisung für beliebige Berechnungen
- Ablaufregel- oder Entscheidungstabellenaufruf zur Wiederverwendung anderer Regeln
- Serviceaufruf zum Aufruf von externen Services
- Wiederholung und Wiederholungsabbruch zum mehrmaligen Durchlaufen von Regelteilen
- Rückkehr zum Verlassen oder Beenden von Regeln
- Ausnahmeauslösung und Ausnahmebehandlung zur Behandlung von Fehlern

## Elemente der Entscheidungstabelle

- Verschachtelung von Entscheidungen sowohl am linken als auch am oberen Rand der Entscheidungstabelle möglich
- Zuweisung zur Definition der Ergebnisse (konstant oder dynamisch)
- Aktionsauslösung zum Ausführen bestimmter Aktionen
- Ausnahmeauslösung für unerlaubte Kombinationen von Bedingungen
- Ausführung einzelner oder mehrerer zutreffender Bedingungen zur Abbildung von Scorecards

## Aktionen & Services

- Ausgabe in eine Datei oder auf den Bildschirm
- Ausgabe in eine CSV Datei (kommaseparierte Werte)
- Ausgabe in ein Log
- E-Mail versenden
- Benutzerdefinierte Aktionen mit beliebiger Funktionalität oder zur Anbindung anderer Systeme
- HTTP Services aufrufen (REST, XML, JSON)
- Inubit Prozesse via REST Connector auslösen

## Funktionen

- Arithmetische Funktionen
- String-Funktionen
- Datum- und Zeitfunktionen
- Listenfunktionen (select, slice, sort, exists, filter, avg, sum, min, max etc.)
- Benutzerdefinierte Funktionen

## Daten

- Definition der Ein-/Ausgabe Daten und Aktionen für jede Regel, Regelpaket oder Regelmodell
- Definition interner und konstanter Daten für jede Regel, Regelpaket oder Regelmodell
- Flexible Datensicht auf Ebene des Regelmodells, Regelpakets oder der Regel
- Einfache Datentypen (Float, Integer, Boolean, String, Date, Time, Timestamp)
- Benutzerdefinierte Datentypen
- Strukturierte Datentypen
- Aufzählungen
- Typ Aliase
- Unterstützung für Vererbungen
- Automatischer Import und Synchronisation von Java Objektmodellen

- Unterstützung für Sammlungen (Collection, List, Set)
- Unterstützung für Maps
- Unterstützung für Java Enumerationen und Generics
- Definition von Metadaten auf verschiedenen Regelmodellkomponenten

## Regelausführung

- Sofortige Regelausführung innerhalb der Modellierungsumgebung
- Ausführungsstatistik- und protokolle (auf Request- oder Session-Ebene)
- Grafische Darstellung der Ausführungsstatistik inklusive Ausführungspfade und Ausführungszeiten (Profiling)
- Regelausführung per API
- Unterstützung von Java 8 Compact Profile zur Regelausführung auf limitierter Hardware

## Testen der Regeln

- Definition von Testfällen im integrierten Testeditor
- Editor für Testdaten und erwarteten Ergebnissen
- Konfiguration und Ausführung von Tests
- Grafische Darstellung der Testergebnisse und Abweichungen von erwarteten Ergebnissen
- Testsuite-Editor zur Definition von Testsuites und zur Ausführung aller darin enthaltenen Tests
- Simulation von Serviceaufrufen während Testausführung

## Debugging

- Grafisches Debugging der Regelausführung (lokal oder remote)
- Unterbrechungspunkte in Ablaufregeln und Entscheidungstabellen
- Anzeige der aktuellen Daten während der Regelausführung

## Java Code Generator

- Umwandlung von Regeln in Java Quellcode durch den Java Code Generator
- Java Code ist leichtgewichtig, zustandslos und multithreading-fähig
- Hoch-performer Regelcode, ideal zur Massen-datenverarbeitung
- Direkte Integration des Regelcodes in Anwendungen (kein Server notwendig)
- Inkrementelle Codegenerierung erzeugt Code nur für tatsächlich veränderte Regeln

## APIs und Erweiterungen

- API zur Integration und für den Aufruf des generierten Regelcodes
- API zur Integration kundenspezifischer Funktionen
- API zur Integration kundenspezifischer Aktionen
- API zur Integration kundenspezifischer Benutzeroberflächen für Aktionen in den Modeler
- API zur Integration kundenspezifischer Services
- API für Statistiken und Profiling (Request und Session Ebene)
- API zum Import und Abgleich externer Datenmodelle

## Onlinehilfe und Dokumentation

- Benutzeroberfläche, Onlinehilfe und Dokumentation in Deutsch und Englisch
- Tutorials und Beispielprojekte
- Assistent für Supportanfragen und Bug-Reports

## Import

- Import von Excel-Tabellen als Entscheidungstabellen
- CSV Datei Import für statische Daten
- Import von Testdaten für Regeltests
- Import/Export von XML Schema

- Import von Simple Rules XML als Ablaufregeln

## **Zustandsabläufe**

- Grafischer Editor zur Modellierung von Zustandsabläufen mit Drag&Drop
- Regeln beim Betreten und Verlassen von Zuständen
- Automatische Validierung und Konsistenzprüfungen der Zustandsabläufe
- Java Code Generator zur Überführung der modellierten Zustandsabläufe in Java Quellcode
- Ausführungs-API zur Ausführung von Zustandsabläufen



## Visual Rules Server: Team-Komponente

### Source-Repository

- Zentrales, mandantenfähiges Source-Repository, auf das über ein Git-Protokoll zugegriffen werden kann
- Zugriff auf Detailinformation der verwalteten Projekt-Ressourcen über das Web UI
- Strikte Trennung von Mandanten-Daten
- Konfliktfreie Modellierung mit mehreren Benutzern
- Veröffentlichung und Synchronisierung von Änderungen von unterschiedlichen Benutzern
- Benutzer- und teambasierte Berechtigungen auf Source-Repositories und Regelprojekten
- Vollständige Versionsgeschichte und Änderungsprotokolle
- Unterstützung von Trunk, Branch und Tag Versionen
- Visualisierung, Vergleich und Abgleich beliebiger Versionsstände ganzer Regelmodelle
- Beliebige andere Dateiformate (z.B. Word-Dokumente) können ebenso gespeichert und versioniert werden
- Konfigurierbares Audit-Logging, um in der Web UI Konfigurationsänderungen und Zugriffsversuche verfolgen zu können
- Bereitstellung von Metriken zur Überwachung des Betriebs mit JMX, Prometheus, Graphite und Grafana
- Verfügbar als eigenständige Spring-Boot-Applikation (mit eingebettetem Tomcat) oder als Webarchiv zur Bereitstellungen auf einem Applikationsserver
- Zwölf-Faktoren-App, die Cloud-fähig und skalierbar ist
- Docker-Support für die Entwicklung
- Optionale Integration mit einem Authentifizierungs-/Autorisierungssystem (inkl. LDAP-Anbindung): auf OAuth 2.0/OpenID Connect basierendes System wie Keycloak oder Bosch IoT Permissions

### APIs und Erweiterungen

- REST API zur Verwaltung von Source-Repositories, Regelprojekten, Team-Aktivitäten (Locks) und Berechtigungen als RESTful Web Services



Änderungen an den APIs sind vorbehalten und können der aktuellen Dokumentation entnommen werden.

### Wartung

- Übersicht über Lizenzinformationen
- Einfache Logging-Konfiguration

## Visual Rules Server: Build-Komponente

### Build-Verwaltung

- Bauen und Testen von verwalteten Regelprojekten
- Erstellen von OSGi-kompatiblen Regelbibliotheken
- Veröffentlichung von Regel-Bibliotheken auf Maven-Repositories
- Veröffentlichung von Rule Services auf Visual Rules Server
- Überwachung von Build-Prozessen
- Konfigurierbares Audit-Logging, um in der Web UI Konfigurationsänderungen und Zugriffsversuche verfolgen zu können
- Bereitstellung von Metriken zur Überwachung des Betriebs mit JMX, Prometheus, Graphite und Grafana
- Verfügbar als eigenständige Spring-Boot-Applikation (mit eingebettetem Tomcat) oder als Webarchiv zur Bereitstellungen auf einem Applikationsserver
- Zwölf-Faktoren-App, die Cloud-fähig und skalierbar ist
- Docker-Support für die Entwicklung

### APIs und Erweiterungen

- REST API zur Verwaltung von Build-Konfigurationen, zum Starten von Build-Prozessen während der Entwicklung oder für ein Release und zur Verwaltung von Maven-Repositories als RESTful Web Services

## Visual Rules Execution Platform

### Visual Rules Server: Execution-Komponente

- Performante Ausführungsumgebung und Verwaltungssystem für Geschäftsregeln
- Bereitstellung von Regeln als SOAP und RESTful Web Services zur Integration von Regelausführungen in Kunden-Applikationen (z.B. .NET Applikationen)
- Einfaches Bereitstellen von Regeln direkt aus dem Visual Rules Modeler, Builder oder Visual Rules Server heraus
- Hot Deployment neuer und bestehender Regeln ohne Unterbrechung oder Neustart des Servers
- Mehrere Versionen desselben Regelwerks gleichzeitig verfügbar
- Spezifische Schnittstellenbeschreibung für jede Regel inklusive Datentypen
- Versionsgenaues Abhängigkeitsmanagement zwischen Regelbibliotheken und anderen Java-Bibliotheken
- Unterstützung für Metadaten und dynamisches Routing von Regelanfragen
- Erzeugung und Abruf von Statistiken mehrerer Regelausführungen zur aggregierten Anzeige im Visual Rules Modeler
- Verfügbar als eigenständige Spring-Boot-Applikation (mit eingebettetem Tomcat) oder als Webarchiv zur Bereitstellungen auf einem Applikationsserver
- Zwölf-Faktoren-App, die Cloud-fähig und skalierbar ist
- Docker-Support für die Entwicklung
- Unterstützung externer Datenbanken zur Speicherung von Regelbibliotheken und Statistiken (siehe Systemanforderungen)
- Optionale Integration mit einem Authentifizierungs-/Autorisierungssystem (inkl. LDAP-Anbindung): auf OAuth 2.0/OpenID Connect basierendes System wie Keycloak oder Bosch IoT Permissions
- Flexible Installation: Konfiguration der Nutzung nur eines oder mehrerer Mandanten
- Niedriger Installationsaufwand durch automatische Initialisierung der Datenbank und des Identity Management
- Moderne, ansprechende Benutzeroberfläche
- Konfigurierbares Audit-Logging, um in der Web UI Konfigurationsänderungen und Zugriffsversuche verfolgen zu können
- Bereitstellung von Metriken zur Überwachung des Betriebs mit JMX, Prometheus, Graphite und Grafana
- Möglichkeit, modellbasierte Konfigurationen und Ressourcen zur Laufzeit zu überschreiben, indem global verwaltete und wiederverwendbare Server-Ressourcen eingesetzt werden
- Bereitstellung einer formellen Spezifikation von Regeln als Web Services in unterschiedlichen Formaten (WSDL für SOAP-basierte Anwendungen, WADL für leichtgewichtige XML/JSON basierte Anwendungen)
- Regelausführung für Testzwecke in der Web UI

### APIs und Erweiterungen

- REST API zur Verwaltung von Regeln als RESTful Web Services

### Execution Core

- Integration von Regelverwaltung und -ausführung in Java-Applikationen
- Fluent Java API zum Verwalten und Ausführen von Regeln und zur Konfiguration der Ausführungsumgebung
- Hot Deployment neuer und bestehender Regeln
- Mehrere Versionen von Regelbibliotheken können gleichzeitig verwendet werden
- Versionsgenaues Abhängigkeitsmanagement zwischen Regelbibliotheken und Java-Bibliotheken
- Unterstützung externer Datenbanken zur Speicherung von und zum Zugriff auf Bibliotheken (siehe Systemanforderungen)
- Kann in Verbindung mit einem Visual Rules Server für das Regel-Management benutzt werden

## Visual Rules DatabaseIntegrator

- Direkter Zugriff von den Regeln auf relationale Datenbanken
- Import von Datentypen aus relationalen Datenbanken
- Definition und Verwaltung von JDBC und JNDI Verbindungen
- Zugriff auf Daten mit SQL SELECT Aktionen
- Datenänderung mit SQL INSERT, UPDATE und DELETE Aktionen
- Transaktionssteuerung mit COMMIT und ROLLBACK Aktionen
- Aufruf von Stored Procedures und Stored Functions mit CALL Aktion
- Erstellen und Löschen von Datenbankelementen (z.B. Tabellen) mit DDL Aktion

## Visual Rules Builder

- Vollständige Integration von Regeln in Maven-basierte Build- und Deploymentverfahren
- Maven Plugin zur Erzeugung von Regelcode
- Maven Plugin zur Validierung von Regelmodellen
- Maven Plugin zur Ausführung von Tests und Test-suites
- Maven Plugin zur Erzeugung von Dokumentation von Testergebnissen
- Maven Plugin zum Paketieren von Regelartefakten (JARs mit Regelcode, einschließlich WSDL, WADL, XSDs und Regelmodelldateien)
- Maven Plugin zum Bereitstellen der Artefakte auf dem Visual Rules Server
- Mit Maven gebaute Regelartefakte sind OSGi kompatibel

## Technische Limitierungen

Hinsichtlich Setup und Verwendung von Visual Rules gibt es einige Limitierungen, die es bei der Integration und der Arbeit mit Visual Rules zu beachten gilt. Die derzeit gültigen Limitierungen, die unten stehend aufgeführt sind, können jederzeit ohne Angabe von Gründen geändert werden. Die Verwendung von Visual Rules ohne Beachtung der Limitierungen ist möglich, wird aber weder empfohlen, noch offiziell unterstützt. Nicht alle Limitierungen wurden explizit geprüft, sie müssen aber auf jeden Fall eingehalten werden.



Ein paar Worte zu den Begriffen, die in der folgenden Tabelle verwendet werden:

Ein Visual Rules Server besteht aus mehreren Komponenten wie Gateway, Registry, Team, Build und Execution. Einige der Komponenten unterstützen horizontale Skalierung in einem Cluster. Die einzelnen Entitäten eines solchen Komponenten-Clusters werden *Instanzen* genannt, z.B. *Execution Instanz*.

Element	Limit-Typ	Wert
Regelprojekte	pro Workspace	200
Regelmodelle	pro Regelprojekt	20
	pro Build-Ausführung	200
Regelpakete	pro Regelmodell	200
	Tiefe pro Regelpaket	20
Modellpfad (rulemodel/package/rule)	Länge (in Buchstaben)	200
<ul style="list-style-type: none"> <li>Regeln</li> <li>Datenelemente (z.B. Ein-/Ausgabedaten, interne Daten)</li> <li>Kundenspezifische Servicetypen</li> <li>Kundenspezifische Aktionstypen</li> <li>Kundenspezifische Funktionen</li> </ul>	Jeder Elementtyp pro Regelmodell	1000
Strukturen	pro Regelprojekt (inkl. wiederverwendete Strukturen)	10000
	Attribute pro Struktur	1000
	Tiefe pro Struktur	20
Zustandsabläufe	pro Regelmodell	200
	Zustände pro Zustandsablauf	200
	Transitionen pro Zustandsablauf	1000
Regelelemente	pro Ablaufregel	2000
Tabellenzellen	pro Entscheidungstabelle	2000
Regeltests & Testsuiten	Regeltests pro Regelprojekt	1000
	Testsuiten pro Regelprojekt	200
	Regeltests pro Regel	200

Element	Limit-Typ	Wert
	Testfälle pro Regeltest	200
	Regeltests/Testsuiten pro Testsuite	200
	Regeltests pro Build-Ausführung	10000
Binäre Code-Artefakte (vra, jar)	Physische Größe	256 MB
Ausführungsergebnis-Artefakte (vrexecution, vrstatistics)	Physische Größe	64 MB
Ausdrücke	Länge (in Buchstaben)	10000
Mandanten	Pro Team/Execution Instanz	40
Repositories	Pro Mandant	20
Branches	Pro Mandant	2000
Projekte	Pro Branch	200
Dateien	Pro Projekt	2000
Buildprozesse (Fortlaufend + Release)	Pro Mandant	200
Build History	Pro Buildprozess	20
Anzahl konfigurierbarer Maven Repositories, Visual Rules Server	Jeder Typ pro Team Instanz	10
Rule Services	Pro Visual Rules Server	4000
Abhängige Bibliotheken	Pro Visual Rules Server	4000
Rule Service Ausführungen	Parallel pro Execution Instanz	400 *
Unterschiedliche Rule Services	Gleichzeitig geladen auf einer Execution Instanz	100 *
Rule Service Bibliothek Dateien	Physische Größe	40 MB
Regelmodell Metadaten	Anzahl Metadaten-Werte	20

**\* Vermerk:**

- Abhängig vom verfügbaren Speicher bzw. von den Speichereinstellungen (32bit/64bit)
- Abhängig von der CPU Leistung
- Vorausgesetzt wurden eine typische Größe und Komplexität eines Regelmodells und seiner Abhängigkeiten

**Einschränkungen bei Mandantenbetrieb:**

- Benutzerdefinierte Servicetypen und Funktionen unterstützen möglicherweise keine volle Datentrennung
- Eingeschränkte Datentrennung bei der Verwendung von externen Maven Repositories

## Datenschutz

Diese Dokumentation liefert dem Visual Rules Betreiber Hinweise für die Erstellung einer Datenschutzerklärung nach [EU Datenschutz-Grundverordnung \(DSGVO\)](#). Es muss eine Abwägung dokumentiert werden zwischen Nachvollziehbarkeit von Änderungen und dem Interesse am Schutz der eigenen Daten. Da die Verordnung keinen Unterschied in der Handhabung zwischen personenbezogenen und personenbeziehbareren Daten macht, werden sie in diesem Guide als Synonym verwendet.

### Allgemein

Generell berücksichtigt Visual Rules die Prinzipien "Privacy by Design" und "Privacy by Default". Im Folgenden einige allgemeine Hinweise zu allen Visual Rules Komponenten.

Anwendungsfall	Verwendung von personenbezogenen Daten	Umgang mit/Löschung von personenbezogenen Daten
Benutzeranmeldung	Für die Authentifizierung des Benutzers und Autorisierung bestimmter Funktionalitäten wird an verschiedenen Stellen eine Benutzeranmeldung (Mandant, Benutzername und Passwort) gefordert. Um die Daten zu einem Benutzer korrekt anzuzeigen, werden grundsätzliche Daten im System verarbeitet und gespeichert.	Siehe Datenschutzhinweise der Benutzerverwaltung.
Log-Dateien	Protokollierung der Datenverarbeitung und Fehleranalyse in verschiedenen Komponenten. Der Umfang kann über Logkonfiguration und der Zugriff organisatorisch eingeschränkt werden.	Löschen der Log-Dateien.
Datenverarbeitung in Regeln	Personenbezogene Daten, welche in Regelprojekten definiert oder verarbeitet werden, können nicht erkannt und deshalb nicht speziell behandelt werden.	Implementieren Sie entsprechende Schutzmaßnahmen innerhalb Ihrer Regeln, so dass die verarbeiteten personenbezogenen Daten vor Zugriffen geschützt sind und auf Bedarf gelöscht werden können.

### Visual Rules Modeler

Dies sind die Datenschutzhinweise für den Modeler.

Anwendungsfall	Verwendung von personenbezogenen Daten	Löschung von personenbezogenen Daten
Notizen	Um Notizen zuordnen und nachvollziehen zu können, wird der Benutzername des im Betriebssystem angemeldeten Benutzers gespeichert.	Notizen löschen.
Visual Rules Server Verbindung	Für die Verbindung zum Visual Rules Server muss Mandant, Benutzername und Passwort eingegeben werden. Diese können optional verschlüsselt gespeichert werden, so dass diese nicht jedes Mal wieder eingegeben werden müssen.	Löschen sie die Verbindungskonfiguration.



Anwendungsfall	Verwendung von personenbezogenen Daten	Löschung von personenbezogenen Daten
Veröffentlichung von Rule Services	Für die Veröffentlichung auf einem Visual Rules Server muss Mandant, Benutzername und Passwort eingegeben werden, falls dieser entsprechend konfiguriert ist. Diese können optional verschlüsselt gespeichert werden, so dass diese nicht jedes Mal wieder eingegeben werden müssen.	Entfernen Sie die Benutzerdaten im Eingabedialog.
Änderungskommentar	Änderungskommentare dienen der Nachvollziehbarkeit von Änderungen durch Bearbeiter an Regeln.	Um die Nachvollziehbarkeit sicherzustellen, ist eine Löschung dieser Daten nicht vorgesehen.
Ressourcenkonfiguration	Ressourcen dienen zur Konfiguration des Zugriffs auf Drittsysteme und Datenbanken innerhalb von Regeln. Es können URLs, Benutzernamen und Passwörter hinterlegt werden. <b>Da die im Regelprojekt abgelegten Daten unverschlüsselt abgelegt werden, nutzen Sie diese bitte nur zu Testzwecken und tragen Sie dann Dummywerte ein.</b> Bitte hinterlegen Sie die Authentifizierungsdaten für eine Ressource sicher im Visual Rules Server.	Entfernen Sie Benutzerdaten von Ressourcen in Regelprojekten.
Metadaten	Metadaten sind freie Schlüssel-Werte-Paare und nicht für die Ablage personenbezogener Daten vorgesehen. Diese Daten werden nicht verschlüsselt gespeichert.	Löschen Sie die Metadaten.

## Visual Rules Server (Team und Build)

Dies sind die Datenschutzhinweise für die Verwendung der Team- und Build-Komponente des Visual Rules Servers.

Anwendungsfall	Verwendung von personenbezogenen Daten	Löschung von personenbezogenen Daten
Repository: Revisionen	Revisionen dienen der Nachvollziehbarkeit von Änderungen durch Bearbeiter an Regeln.	Um die Nachvollziehbarkeit sicherzustellen, ist eine Löschung dieser Daten nicht vorgesehen.
Globale Build Konfiguration	Für das Deployment auf Visual Rules Server oder Maven Repositories müssen Benutzerdaten hinterlegt werden, so dass diese nicht jedes Mal wieder eingegeben werden müssen. Diese werden verschlüsselt abgelegt.	Bitte erstellen und hinterlegen Sie hierfür technische Nutzer.
Build und Logausgaben	Zwecks Nachvollziehbarkeit wird der Benutzer gespeichert, der einen Build gestartet hat, sowie die Logausgaben des Builds.	Löschen Sie die Builds in der UI und auf dem Dateisystem.
Audit Log	Das Audit Log dient der Nachvollziehbarkeit von Benutzeraktionen wie Konfigurationsveränderungen. Hierfür werden Benutzernamen und IP-Adressen gespeichert.	Um die Nachvollziehbarkeit sicherzustellen, ist eine Löschung dieser Daten nicht vorgesehen. Das Audit Log kann aber deaktiviert werden.

Anwendungsfall	Verwendung von personenbezogenen Daten	Löschung von personenbezogenen Daten
Zugriffsrechte	Der Zugriff auf Source-Repositories und Regelprojekte kann für einzelne Benutzer oder Gruppen festgelegt werden.	Pflegen Sie Zugriffsrechte auf Gruppenebene und löschen Sie Zugriffsrechte für einzelne Benutzer.


## Execution Platform

Dies sind die Datenschutzhinweise für die Verwendung der Execution Platform.

Anwendungsfall	Verwendung von personenbezogenen Daten	Löschung von personenbezogenen Daten
Server-Ressourcen	Server-Ressourcen dienen zur Konfiguration des Zugriffs auf Drittsysteme und Datenbanken innerhalb von Regeln. Diese werden in Form von Schlüssel-Wert Paaren, abgelegt wobei sensitive Werte wie Benutzernamen oder Passwörter optional verschlüsselt abgelegt werden können.	Löschen Sie die Schlüssel-Wert Paare oder die gesamte Server-Ressource.
Metadaten	Metadaten sind freie Schlüssel-Werte-Paare und nicht für die Ablage personenbezogener Daten vorgesehen. Diese Daten werden nicht verschlüsselt gespeichert.	Löschen Sie die Metadaten.
Veränderungsdetails	Es werden bei Bibliotheken und Ausführungen personenbezogene Daten zur Nachvollziehbarkeit gespeichert und angezeigt. Es muss eine Abwägung gemacht werden zwischen Nachvollziehbarkeit und dem Interesse der Bearbeiter am Schutz Ihrer Daten.	Eine einzelne Löschung von personenbezogenen Daten ist nicht vorgesehen, aber es kann auf das Schreiben des Audit Logs verzichtet werden.
Audit Log	Das Audit Log dient der Nachvollziehbarkeit von Benutzeraktionen wie Konfigurationsveränderungen. Hierfür werden Benutzernamen und IP-Adressen gespeichert.	Um die Nachvollziehbarkeit sicherzustellen, ist eine Löschung dieser Daten nicht vorgesehen. Das Audit Log kann aber deaktiviert werden.

## Benutzerverwaltung

Dies sind die Datenschutzhinweise für die Verwendung der Benutzerverwaltung.

Anwendungsfall	Verwendung von personenbezogenen Daten	Löschung von personenbezogenen Daten
Benutzerverwaltung	Bei der Anmeldung eines Benutzers am Visual Rules Server werden folgende Daten aus dem angebundenen Authentifizierungs-/Autorisierungssystem übernommen und gespeichert: Benutzername, Vorname, Nachname, erstes und letztes Login-Datum, Beschreibung, Rollen, Gruppen und Mandant.	Entitäten in der Benutzerverwaltung des Visual Rules Servers und im angebundenen Authentifizierungssystem löschen.  <div style="border: 1px solid #ccc; padding: 5px; background-color: #f9f9f9;">  Das Löschen im angebundenen Authentifizierungssystem führt nicht automatisch zum Löschen der Entität in der Benutzerverwaltung des Visual Rules Servers (und umgekehrt). </div>
Berechtigungsmanagement	Benutzern können im angebundenen Authentifizierungssystem Gruppen und Rollen mit speziellen Berechtigungen zugewiesen werden.	Zuweisungen im angebundenen Authentifizierungssystem löschen.
Audit Log	Das Audit Log dient der Nachvollziehbarkeit von Benutzeraktionen bei der Authentifizierung. Hierfür werden der Benutzername, der Status und Zeitpunkt des Logins gespeichert.	Um die Nachvollziehbarkeit sicherzustellen, ist eine Löschung dieser Daten nicht vorgesehen. Audit Log Einträge von gelöschten Benutzern enthalten lediglich eine nicht menschenlesbare Benutzer-ID. Das Audit Log kann vollständig deaktiviert werden.

## IoT Permissions

In Cloudumgebungen kann der Bosch IoT Permissions Service als angebundenes Authentifizierungs-/Autorisierungssystem für die Benutzerverwaltung des Visual Rules Servers verwendet werden. In diesem Fall gelten dieselben Datenschutzhinweise, wie für [Benutzerverwaltung](#) beschrieben. Es können dort optional weitere Benutzerdaten hinterlegt werden, die Visual Rules jedoch nicht auswertet: Bild, Geburtsdatum, Sprache, Telefon, Adresse und ggf. weitere. Die vollständigen Datenschutzhinweise können im [Bosch IoT Permissions Privacy Leaflet](#) eingesehen werden.

## Über Bosch Software Innovations

Die **Bosch Software Innovations GmbH**, das Software- und Systemhaus der Bosch-Gruppe, konzipiert, entwickelt und betreibt weltweit innovative Software- und Systemlösungen, die unsere Kunden sowohl im Internet of Things (IoT) als auch im klassischen Enterprise-Umfeld voranbringen. Unseren Fokus legen wir dabei auf die Themenfelder Mobilität, Energie, Fertigungsindustrie und Gebäude. Unsere IoT-Plattform – die Bosch IoT Suite – bildet die technologische Basis, die das Zusammenspiel von Geräten, Anwendern, Unternehmen und Partnern auf einer Plattform erlaubt. Dadurch wird die Entwicklung innovativer und zukunftsfähiger Lösungen für neue Geschäftsmodelle ermöglicht.

Bosch Software Innovations ist mit rund 800 Mitarbeitern weltweit mit Standorten in Deutschland (Berlin, Immenstaad am Bodensee und Stuttgart), in Singapur, China (Schanghai) und den USA (Palo Alto) vertreten.

Mehr Informationen unter [www.bosch-si.de](http://www.bosch-si.de), [www.twitter.com/BoschSI](https://www.twitter.com/BoschSI) und [www.blog.bosch-si.com](http://www.blog.bosch-si.com).

Die **Bosch-Gruppe** ist ein international führendes Technologie- und Dienstleistungsunternehmen mit weltweit rund 360 000 Mitarbeitern (Stand: 01.04.2015). Sie erwirtschaftete im Geschäftsjahr 2014 einen Umsatz von 49 Milliarden Euro<sup>1</sup>. Die Aktivitäten gliedern sich in die vier Unternehmensbereiche Mobility Solutions, Industrial Technology, Consumer Goods sowie Energy and Building Technology. Die Bosch-Gruppe umfasst die Robert Bosch GmbH und ihre rund 440 Tochter- und Regionalgesellschaften in rund 60 Ländern. Inklusive Handels- und Dienstleistungspartnern ist Bosch in rund 150 Ländern vertreten. Dieser weltweite Entwicklungs-, Fertigungs- und Vertriebsverbund ist die Voraussetzung für weiteres Wachstum. Im Jahr 2014 meldete Bosch weltweit rund 4 600 Patente an. Strategisches Ziel der Bosch-Gruppe sind Lösungen für das vernetzte Leben. Mit innovativen und begeisternden Produkten und Dienstleistungen verbessert Bosch weltweit die Lebensqualität der Menschen. Bosch bietet „Technik fürs Leben“.

Mehr Informationen unter [www.bosch.com](http://www.bosch.com), [www.bosch-presse.de](http://www.bosch-presse.de) und <http://twitter.com/BoschPresse>.

---

<sup>1</sup>Im Umsatzausweis 2014 sind die zwischenzeitlich komplett übernommenen bisherigen Gemeinschaftsunternehmen BSH Bosch und Siemens Hausgeräte GmbH (heute: BSH Hausgeräte GmbH) und ZF Lenksysteme GmbH (heute: Robert Bosch Automotive Steering GmbH) nicht enthalten.