

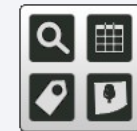


*j*ACOB

Eclipse

Verside

Online



Greifen Sie zuverlässig und mit hoher Leistung auf wichtige Daten zu

Mit jACOB und/oder Verside stellen Sie Ihre Datenbanken Anwendern in einem Netzwerk oder im Web zur Verfügung. jACOB ist eine schnelle, zuverlässige und bedienerfreundliche Serversoftware für die Verwaltung und Erstellung Ihrer Datenbanken.

***jACOB* ist Buzzwords compliant ;-)**

Eclipse

Der jACOB Designer wurde als Set von Eclipse Plugins umgesetzt

Eclipse BIRT

jACOB kann Reports welche mit dem ReportDesigner BIRT erstellt wurden ausführen und ausliefern.

Ad Hoc Reporting

Anwender können sich einfache Excellisten als Report zusammenstellen.

Live Grid

Skalierbar

JACOB unterstützt das Clustern von Anwendungen

Enterprise Scheduler

Hintergrundprozesse können als CRON Job pro Anwender, Instanz oder pro Cluster ausgeführt werden.

Hot Deployment

Anwendungen können im Betrieb ausgetauscht werden.

DHTML

Kann ein jACOB Entwickler getrost vergessen

Rich Internet Application

Sind in der Handhabung und Funktionalität Desktop Anwendungen immer ähnlicher wenn nicht sogar ebenbürtig

QBE

„Query By Example“ ist die Anfragesprache bei jACOB

I18N

Anwendungen können mit dem Designer vollständig lokalisiert

UI Plugin

JACOB kann durch eigenen UI-Plugins erweitert werden.

Ajax

Wird in jACOB eingesetzt um die Usability zum Maximum zu treiben.

Data Access Layer

Die Eigenarten der unterschiedlichen Datenbanken wird durch den Data Layer von jACOB komplett verborgen. Kein SQL.

SOAP

„Point and Click“ Integration von Apache SOAP im jACOB Designer

WebDAV

Dokumente innerhalb von jACOB können über WebDav bearbeitet werden.

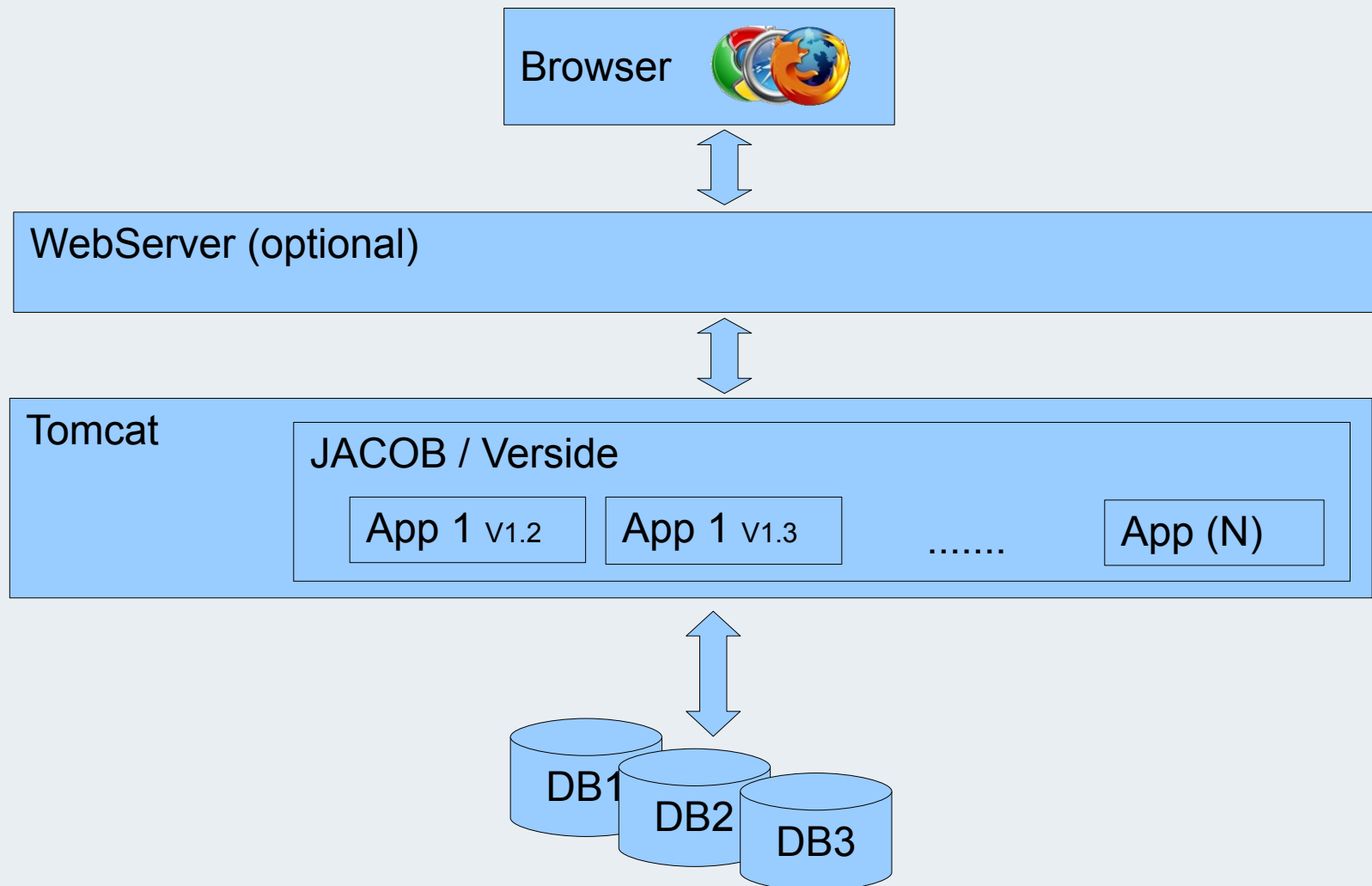
LDAP

jACOB unterstützt LDAP als eine weitere Datenquelle.

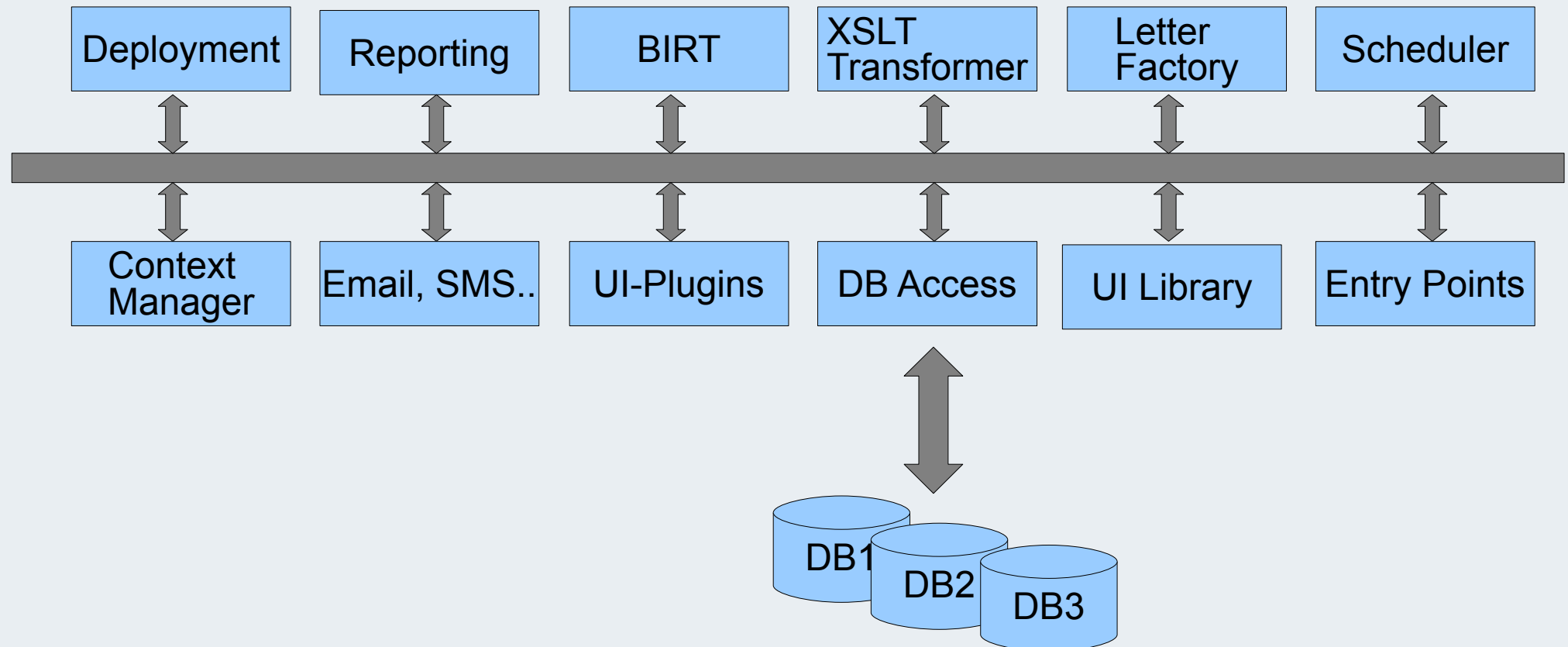
JavaScript

Hat für einen jACOB Entwickler keine Bedeutung.

Architektur Übersicht



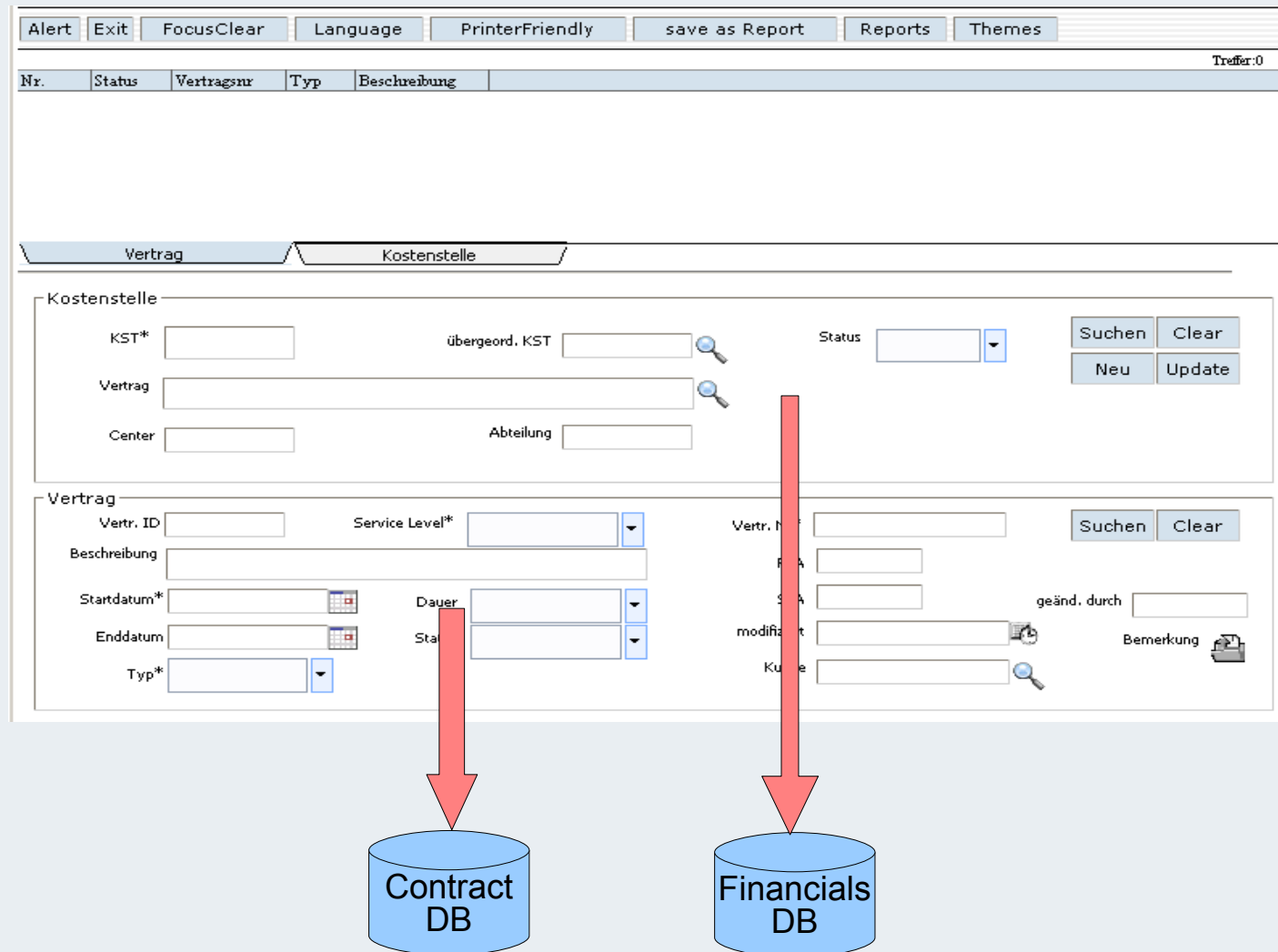
Komponenten



Data Layer

- Abstraktions- und Zugriffsschicht für Daten in tabellarischer Form. Insbesondere (aber nicht ausschließlich) auf Daten in relationalen Datenbanken
- Unterstützung von kommerziellen und Open Source Datenbanken
- Gleichzeitiger Zugriff auf unterschiedlichen Datenquellen
- Allgemein anwendbare QBE (Query By Example) Funktionalität
- Cache von Daten pro Benutzer
- Unterstützung von Transaktionen, ebenfalls „Nested Transaktionen“
- Allgemeiner Zugriff auf Administrations- und Monitoringdaten zur Steuerung und Überwachung der einzelnen Komponenten.

Applikation mit unterschiedlichen Datenquellen



Ad-Hoc Report Komponente

- Erstellung von Ad Hoc Reports wahlweise über Benutzerführung oder per API
- Speichern und Wiederaufruf von Reports in XML Format
- Erstellung von „on-Screen“ Berichten und Export nach Excel

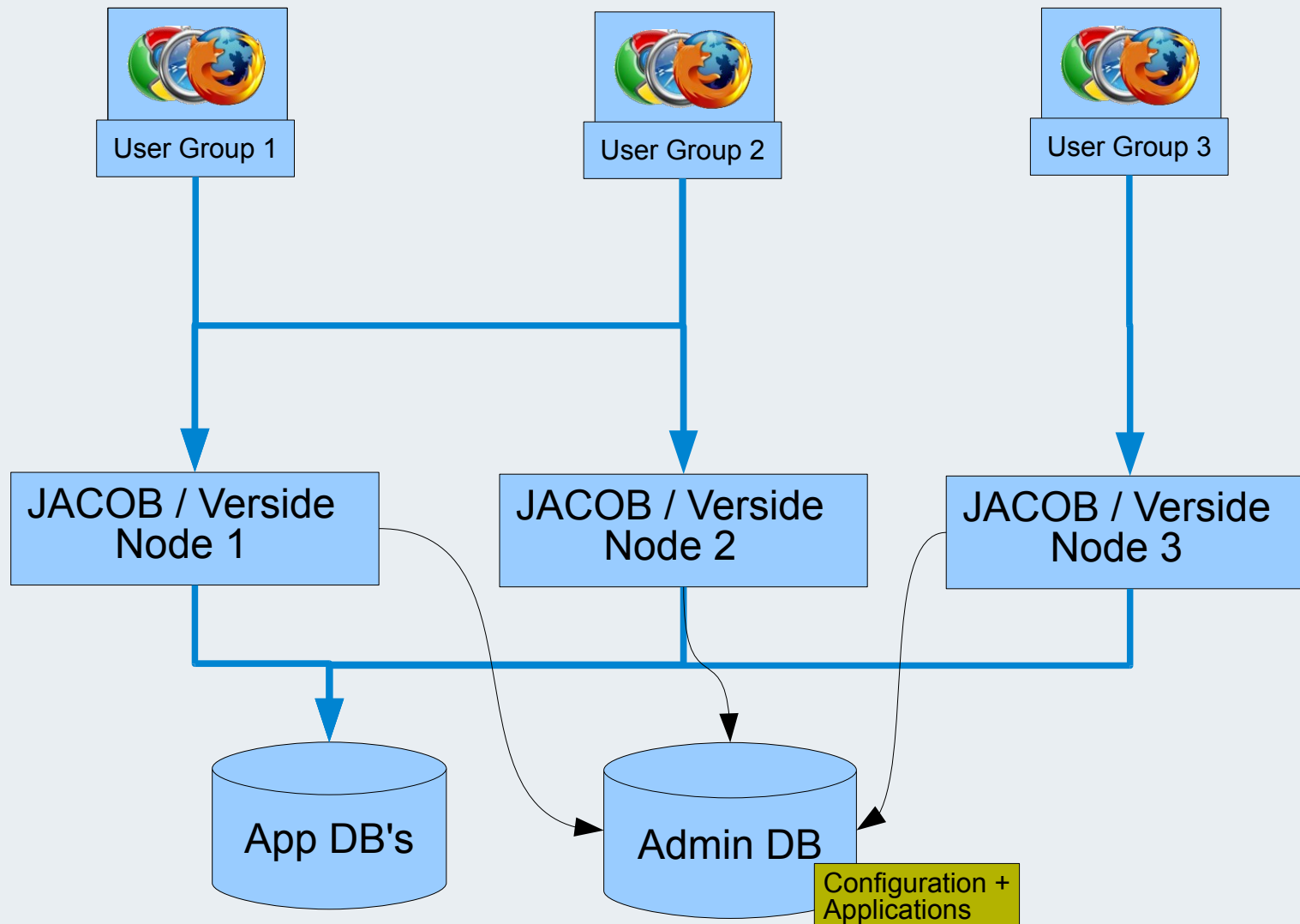
UI Library

- Client-unabhängige Bibliothek von UI-basierten Funktionen (z.B Erstellung von Formular basierten Dialogen)
- Erweiterbar durch applikationseigene Plugins
- Erweiterbar durch Engine Plugins

Deployment Komponenten

- Zur Verfügung stellen von Applikationen und Ressourcen
- Versionierung von Applikationen
- Hot-Deployment von neuen Funktionen und Versionen
- Gleichzeitiger Betrieb von unterschiedlichen Applikationen und von unterschiedlichen Versionen derselben Applikation
- Management von Live, Test und archivierten Applikationen und Versionen

Skalierbarkeit: Variante 1



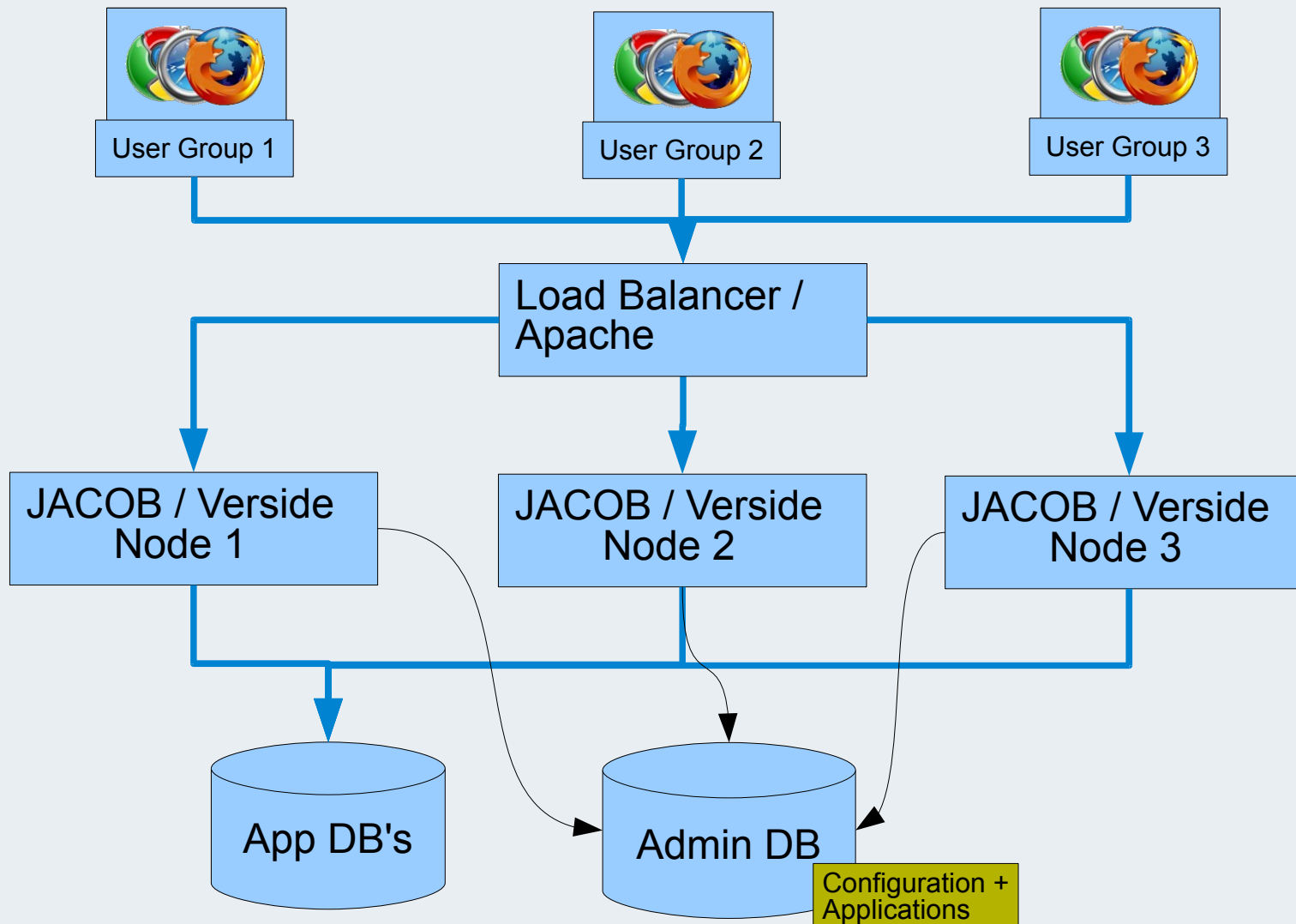
Mehrere jACOB Instanzen (nodes) können zu einem jACOB Cluster kombiniert werden.

JACOB Instanzen laufen im selben Cluster, wenn Sie sich die gleiche Konfigurationsdatenbank teilen.

Wird eine Applikation in einer Instanz deployed, steht sie in allen Instanzen zur Verfügung

User Sessions sind an eine Instanz gebunden, es findet kein Sharing statt.

Skalierbarkeit: Variante 2

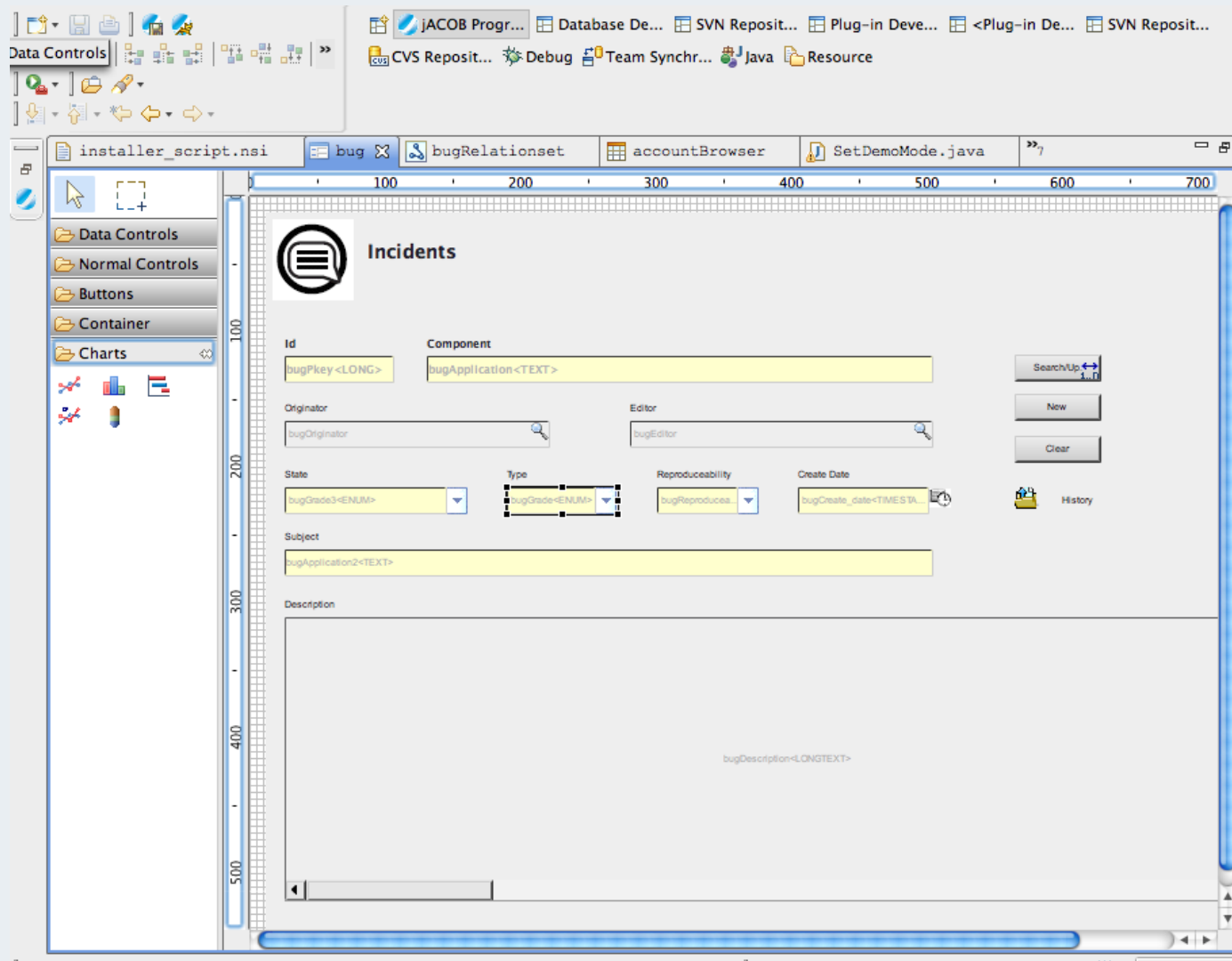


Load Balancer

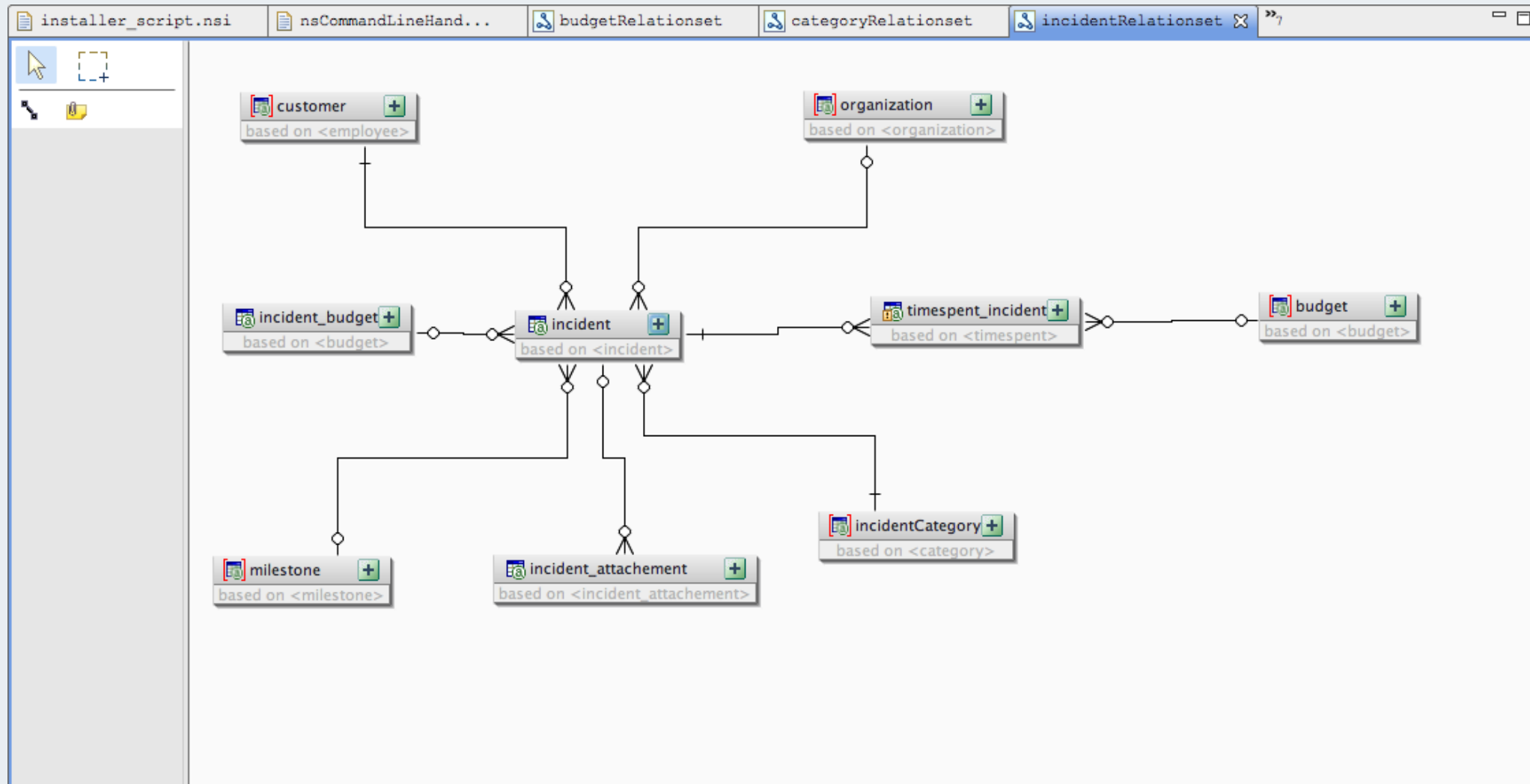
Mittels eines Load Balancer können Benutzeressions gleichmäßig über verschiedene Instanzen verteilt werden. Mögliche Vertreter sind Cisco oder Pound.

Eine Weitere Lösung ist die Verwendung der Tomcat Balancer Web Applikation welche die Anfragen Regelbasiert oder on Demand über die einzelnen Instanzen verteilt.

Eclipse UI Editor



Eclipse Relationset Editor



Eclipse DB Table Editor

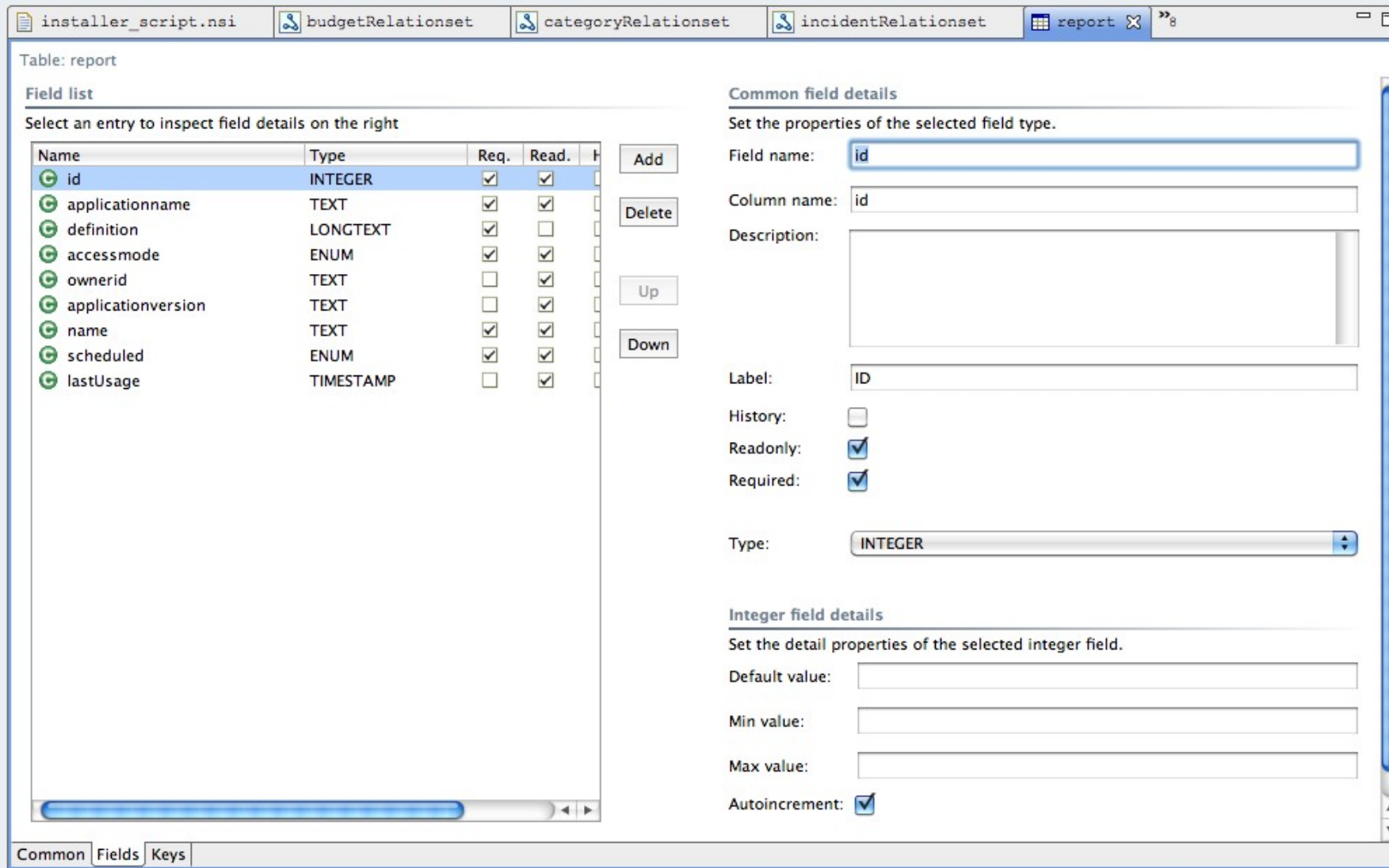


Table: report

Field list

Select an entry to inspect field details on the right

Name	Type	Req.	Read.	
id	INTEGER	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
applicationname	TEXT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
definition	LONGTEXT	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
accessmode	ENUM	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
ownerid	TEXT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
applicationversion	TEXT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
name	TEXT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
scheduled	ENUM	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
lastUsage	TIMESTAMP	<input type="checkbox"/>	<input checked="" type="checkbox"/>	

Buttons: Add, Delete, Up, Down

Common field details

Set the properties of the selected field type.

Field name: id

Column name: id

Description:

Label: ID

History: ☐

Readonly: ☒

Required: ☒

Type: INTEGER

Integer field details

Set the detail properties of the selected integer field.

Default value:

Min value:



Max value:

Autoincrement: ☒

Common Fields Keys

Verside.org Web UI Editor

Verside Demo Application 1.2 Abmelden



 

Tracker

Bug


Stammdaten

Account


 

Felder für : bug

- ☒ pkey
- ☐ history
- ☐ application
- ☒ title
- ☐ description



Bug

 Incidents

Id Component


Originator Editor




State Type Reproduceability Create Date

Subject

Description

Suchen **Neu** **Clear**

 History

Search **New** **Update** **Clear** **Delete**    Upload