

Kernsoftware Release-Notes

Jonathan Haas, Roland Schmitz

4. August 2016



Kappich Systemberatung

Inhaltsverzeichnis

1	Einleitung	4
2	Kernsoftware 3.9.1	5
2.1	Bugfixes	5
3	Kernsoftware 3.9.0	6
3.1	Neue Features	6
3.2	Änderungen	9
3.3	Bugfixes	10
3.4	Technische Details zur Verschlüsselung	10
3.4.1	Authentifizierung	10
3.4.2	Verschlüsselung	11
4	Kernsoftware 3.8.1	12
4.1	Bugfixes	12
5	Kernsoftware 3.8.0	13
5.1	Änderungen	13
6	Kernsoftware 3.7.4	14
6.1	Bugfixes	14
7	Kernsoftware 3.7.3	15
7.1	Neue Features	15
7.2	Bugfixes	15
8	Kernsoftware 3.7.2	17
8.1	Änderungen	17
8.2	Bugfixes	17
9	Kernsoftware 3.7.1	19
9.1	Bugfixes	19
10	Kernsoftware 3.7.0	21
10.1	Neue Features	21
10.2	Änderungen	24
10.3	Bugfixes	24

10.4 Datenmodelländerungen	25
11 Kernsoftware 3.6.5	26
11.1 Bugfixes	26
12 Kernsoftware 3.6.4	27
12.1 Bugfixes	27
13 Kernsoftware 3.6.3	29
13.1 Änderungen	29
13.2 Bugfixes	32
13.3 Datenmodelländerungen	33
14 Kernsoftware 3.6.2	34
14.1 Bugfixes	34
15 Kernsoftware 3.6.1	35
15.1 Bugfixes	35
16 Kernsoftware 3.6.0	36
16.1 Neue Features	36
16.2 Änderungen	38
16.3 Bugfixes	39
16.4 Datenmodelländerungen	40
17 Ältere Releases	41

1 Einleitung

Im folgenden Dokument werden die Release-Notes zur Kernsoftware in umgekehrter chronologischer Reihenfolge aufgelistet, so dass die Informationen zum letzten Release zuerst aufgeführt werden.

2 Kernsoftware 3.9.1

Release-Datum: 04.08.2016

2.1 Bugfixes

Folgende Probleme vorhergehender Versionen wurden behoben:

- NullPointerException bei verschlüsselten Verbindungen zwischen verschiedenen Rechnern.
- In bestimmten Situation zeigte die Combobox zur Auswahl der Portnummer im Login-Dialog des GTM ein fehlerhaftes Verhalten und falsches Layout.

3 Kernsoftware 3.9.0

Release-Datum: 28.07.2016

3.1 Neue Features

Folgende Erweiterungen gegenüber vorhergehenden Versionen wurden durchgeführt:

- Das Kommunikationsprotokoll zwischen Applikation und Datenverteiler sowie zwischen zwei Datenverteilern wurde um eine verbesserte Authentifizierung und Verschlüsselung ergänzt. Voraussetzung ist die Aktualisierung von Datenverteiler, Konfiguration und den DAF der verbundenen Applikationen. Verwendet wird das SRP6a-Verfahren zur Authentifizierung und AES-GCM zur Verschlüsselung.
- Im ClientDavInterface gibt es die neue Methode `getEncryptionStatus()`, mit der angefragt werden kann, ob die Verbindung aktuell verschlüsselt ist und welcher Algorithmus verwendet wird.
- Bei Verwendung der neuen Authentifizierung braucht statt dem Passwort in der *benutzerverwaltung.xml* der Konfiguration nur noch ein Überprüfungscode gespeichert werden. Aus dem Überprüfungscode kann weder das Passwort bestimmt werden, noch kann man sich mit Hilfe des Überprüfungscode authentifizieren.
 - **Wird mit der neuen Software ein neues Passwort gesetzt, wird nur noch der Überprüfungscode an die Konfiguration übertragen und dort gespeichert. Hierbei muss beachtet werden, dass danach die Authentifizierung mit dem bisherigen Verfahren für diesen Benutzer nicht mehr möglich ist, es müssen also vorher alle Applikationen und Datenverteiler, die diesen Benutzer zur Authentifizierung verwenden, aktualisiert werden.**

Soll ein neues Passwort im Klartext gesetzt werden, weil noch Applikationen oder gekoppelte Datenverteiler nicht aktualisiert wurden, kann entweder die Systemproperty `srp6.disable.verifier` gesetzt werden (JVM-Aufrufparameter: `-Dsrp6.disable.verifier`), oder das Passwort von einem nicht aktualisierten System aus gesetzt werden.

- Überprüfungscode sind an einen Benutzer gebunden. Es ist nicht möglich, den Überprüfungscode eines Benutzers in der *benutzerverwaltung.xml* bei einem anderen Benutzer zu ergänzen oder bei einer “Umbenennung” beizubehalten. In dem Fall ist immer ein neues Passwort zu vergeben. Wenn zwei Benutzer denselben Überprüfungscode besitzen, ist das in jedem Fall ein Fehler.
- Clientseitig können die in der lokalen Authentifizierungsdatei (*passwd*) gespeicherten Klartextpasswörter für die automatische Anmeldung von Applikationen durch Login-Token ersetzt werden. Ein Login-Token kann (nur) bei Kenntnis des Klartextpassworts und des Überprüfungscode berechnet werden und erlaubt eine Authentifizierung ohne das Klartextpasswort speichern zu müssen.

Ein Login-Token ist immer nur für einen bestimmten Benutzer gültig und nur solange, bis ein neues Passwort vergeben wird. Das bedeutet: Ein Login-Token wird immer ungültig wenn das Passwort eines Benutzers neu gesetzt wird, selbst wenn das eigentliche Klartextpasswort gleich bleibt. Aus dem Login-Token ist das Klartextpasswort nicht ableitbar und der Login-Token kann nicht zur Authentifizierung für die Benutzerverwaltung (im *UserAdministration*-Interface) oder im Urlasser-Dialog verwendet werden. Trotzdem sollte der Login-Token möglichst geheim gehalten werden, da damit offensichtlich eine Authentifizierung beim Datenverteiler möglich ist.

- Login-Token können mit dem Migrationswerkzeug berechnet werden, oder mit der Methode `SrpClientAuthentication.createLoginToken`.
- Beispielzeile einer *passwd*-Datei mit Login-Token:

```
Tester=SRP6~~~~ 46f981cc468a72d3726112e8f7b33
```

- Es ist jetzt möglich, in der Authentifizierungsdatei (*passwd*) vom Datenverteiler je Kommunikationspartner ein eigenes Passwort bzw. Login-Token anzugeben. Hierbei wird die Pid den Datenvertelers mit einem @-Zeichen an den Benutzernamen angehängt. Zusätzlich kann die Pid des lokalen Konfigurationsverantwortlichen angegeben werden um das Passwort zur Authentifizierung bei der Konfiguration festzulegen. Beispiel:

```
Tester=SRP6~~~~ 46f981cc468a72d3726112e8f7b33
Tester@dav.TestUZ=KlartextPasswort
Tester@dav.VRZ3=SRP6~~~~ 3ef530a0f5a669fd2c652339485bf
Tester@kv.test=SRP6~~~~ 7c36080fd21f2cfb4476897130885
```

Ist für einen Kommunikationspartner kein spezifischer Eintrag vorhanden, wird das Standard-Passwort (bzw. der Standard-Login-Token) ohne @-Suffix verwendet (sofern vorhanden).

- Beim Login mit Einmalpasswörtern gilt das selbe wie für normale Passwörter, es muss jetzt allerdings der Index des zu verwendenden Einmalpassworts an den Benutzernamen angehängt werden. Will sich beispielsweise der Benutzer "Operator" mit dem Einmalpasswort mit dem Index 54 einloggen, dann muss bei Verwendung der neuen Authentifizierung "Operator-54" als Benutzername eingegeben werden. Da der Passwortindex zur Authentifizierung jetzt erforderlich ist, bieten das Migrationswerkzeug sowie das **UserAdministration**-Interface jetzt neue Methoden, um beim Anlegen eines Einmalpassworts den zugehörigen Index mit zurückzugeben.

Im Gegensatz zu früheren Kernsoftwareversionen können die noch gültigen Einmalpasswörter in beliebiger Reihenfolge verwendet werden.

- Der Datenverteiler authentifiziert sich jetzt zusätzlich bei der Konfiguration. In der Regel sind hierbei keine Änderungen notwendig, es sollte aber sichergestellt sein, dass der Benutzername, mit dem der Datenverteiler gestartet wird (Aufrufparameter `-benutzer=`) ein gültiger Benutzer in der lokalen Konfiguration ist und dieser Admin-Rechte besitzt. Hierdurch wird sichergestellt, dass (nur) der Datenverteiler berechtigt ist, Benutzer zu authentifizieren und bspw. Einmalpasswörter als benutzt zu markieren. Es kann ggf. sinnvoll sein, einen eigenen Benutzer für den Datenverteiler anzulegen.
 - Fehlt dieser Benutzer wird eine Warnung beim Start ausgegeben und weiterhin die alte Authentifizierung verwendet.
- Das Passwort für die Parametrierung kann jetzt auch wie bei einem normalen Benutzer in der Konfiguration gespeichert werden, das Klartextpasswort sollte dann aus der `passwd`-Datei des Datenverters entfernt werden (nur falls die Parametrierung beim Start nicht auf dieselbe Datei zugreift). Das Aufrufargument `-parametrierungsBenutzer=` sorgt beim Datenverteiler weiterhin dafür, dass eine unter dem angegebenen Benutzer gestartete Parametrierung volle Zugriffsrechte bzgl. der Rechteprüfung hat.
- Die Benutzerverwaltungsfunktionen (Interface **UserAdministration**) wurden um Funktionen erweitert, die die neue Authentifizierung betreffen:
 - `setSrpVerifier`: Setzt den Überprüfungscode eines Benutzers direkt. Diese Methode kann verwendet werden, um den Überprüfungscode eines Benutzers zu setzen, ohne dass das Klartextpasswort bekannt ist. Dies ist beispielsweise sinnvoll, wenn Überprüfungscode zwischen verschiedenen Datenvertersystemen ausgetauscht werden sollen.
 - Die Methode `createSingleServingPassword` ist auf `@Deprecated` gesetzt worden, da sie den Passwortindex nicht zurück gibt und man sich dadurch nicht mehr ohne weiteres mit den erzeugten Einmalpasswörtern einloggen kann. Als Alternative gibt es die neue Methode `createOneTimePasswords`, die diesen Index zurück gibt.

- Weitere Methoden um (bei vorhandenen Rechten) einzelne Einmalpasswörter zu deaktivieren und den Überprüfungscode sowie Verschlüsselungsstatus abzufragen.
- Die Schnittstelle des `ClientDavInterface` und ähnlicher Klassen wurden erweitert, um neben Klartextpasswörtern auch die Authentifizierung mit Login-Token zu ermöglichen.
- Durch den Aufrufparameter `-authentifizierung=interaktiv` (statt `-authentifizierung=<Passwortdatei>`) kann das Passwort nun auch bei den meisten Konsolenanwendungen interaktiv abgefragt werden. Es ist daher zur Authentifizierung von Konsolenanwendungen nicht mehr erforderlich, sein Passwort in einer Datei zu speichern. Der interaktive Login funktioniert bei allen Applikationen, die die `ClientDavParameters` über die Aufrufargumente initialisieren. Das ist unter anderem bei allen Applikationen der Fall, die die `de.bsvrz.sys.funclib.application` verwenden.
- Im beiliegenden Tutorial wird erklärt, wie bestehende Systeme so umgestellt werden können, dass alle Passwörter verschlüsselt gespeichert werden.

3.2 Änderungen

Folgende Änderungen zu vorhergehenden Versionen wurden durchgeführt:

- Bei wiederholten Login-Versuchen mit falschen Zugangsdaten werden weitere Login-Versuche nun künstlich ausgebremst um Brute-Force-Angriffe zu erschweren.
- Verschiedene Sicherheitsverbesserungen bei der Verarbeitung von Telegrammen vom Datenverteiler.
- Verschiedene Sicherheitsverbesserungen beim Anmelden der lokalen Datenverteilerapplikation (`TransmitterLocalApplication`).
- Die Telegramme zum Austausch von Anmeldelisten werden jetzt mit der korrekten Länge versendet. Die direkte Kopplung mit Datenverteilern, die eine ältere Version der Kernsoftware als 3.1 (31. Mai 2007) verwenden ist dadurch nicht mehr möglich.
- Passwörter (und Login-Token) aus der Authentifizierungsdatei (`passwd`) werden nicht mehr im Speicher gehalten, sondern bei jedem Login-Versuch neu aus der Datei gelesen.

- Dadurch lassen sich jetzt auch Passwörter/Login-Token zur Authentifizierung bei anderen Datenverteilern im laufenden Betrieb ändern, indem die Authentifizierungsdatei (passwd) des Datenverteilers im laufenden Betrieb bearbeitet wird. Ein Neustart des Datenverteilers ist damit nicht mehr notwendig, wenn auf der Gegenseite neue Passwörter vergeben wurden.
- Die Performance beim applikationsseitigen Abfragen von Mengenelementen wurde stark verbessert, da nun alle noch nicht lokal bekannten Objekte in einer einzigen Anfrage ermittelt werden, statt für jedes fehlende Objekt nacheinander eine neue Anfrage zu stellen.
- Die Authentifizierung von KExDaV bei Remote-Datenverteilern wurde auf die neue Funktionalität umgestellt, es wird die gleiche Syntax der passwd-Datei wie bei der Datenverteilerkopplung verwendet und es sind jetzt auch Login-Token zur Anmeldung möglich.

3.3 Bugfixes

Folgende Probleme vorhergehender Versionen wurden behoben:

- Die Kombination von Einmalpasswörtern mit einer zweiten Verbindung hat zu Problemen geführt da die zweite Verbindung sich nicht ein zweites mal mit dem gleichen Passwort authentifizieren konnte. Die zweite Verbindung wird bei Verwendung von Einmalpasswörtern nun automatisch deaktiviert.

3.4 Technische Details zur Verschlüsselung

3.4.1 Authentifizierung

- Authentifizierungsverfahren: SRP6a (Implementierung: NimbusSRP)
- X-Routine: $H(s \mid H(u \mid ":" \mid KDF(p, s)))$
- Hashfunktion H: Standardmäßig SHA-256
- Schlüsselableitungsfunktion KDF: Standardmäßig PBKDF2WithHmacSHA256 mit 20.000 Iterationen und 256 Bit Rückgabe
- SRP-Primzahl N: Standardmäßig 1024 Bit aus RFC 5054¹

¹<https://tools.ietf.org/html/rfc5054#appendix-A>

3.4.2 Verschlüsselung

- Verschlüsselungsverfahren: AES-GCM
- Schlüssellänge: Standardmäßig 128 Bit
- GCM-Tag-Bits: Standardmäßig 96 Bit
- Ableitung der AES-Schlüssel aus dem gemeinsamen SRP-Sitzungsschlüssel S (bei AES-128):

$S^* = H(0x00000000 \mid S) \mid H(0x00000001 \mid S) \mid H(0x00000002 \mid S) \dots$

$= \text{Schlüssel}(\text{ClientServer}) \mid \text{Schlüssel}(\text{ServerClient}) \mid \text{Nonce-Salt}(\text{ClientServer}) \mid \text{Nonce-Salt}(\text{ServerClient})$

(\mid = Verkettung)

Nonce-Berechnung aus [RFC 5288](https://tools.ietf.org/html/rfc5288#section-3)² mit Verwendung von einem Telegrammzähler als explizitem Nonce-Teil.

²<https://tools.ietf.org/html/rfc5288#section-3>

4 Kernsoftware 3.8.1

Release-Datum: ??.??.2016

4.1 Bugfixes

Folgende Probleme vorhergehender Versionen wurden behoben:

- Nerz-FM-206. Bei den Methoden zum An- und Abmelden auf Online-Daten incl. kurzfristige Historie aus dem Archiv (`ArchiveRequestManager.unsubscribeReceiver()` und `ArchiveRequestManager.unsubscribeReceiver()`), wurde eine fehlerhafte `equals()`-Implementierung verwendet, die dazu führte, dass beim Abmelden Exceptions aufgetreten sind, wenn nicht die selben `DataDescription`-Objekte verwendet wurden, wie beim Anmelden.

5 Kernsoftware 3.8.0

Release-Datum: 31.05.2016

5.1 Änderungen

Folgende Änderungen zu vorhergehenden Versionen wurden durchgeführt:

- Da leere Strings als Meldungs-ID zu Problemen in der Betriebsmeldungsverwaltung führen, erzeugt die Funclib `OperatingMessage` beim Versand von Betriebsmeldungen nun automatisch zufällige, eindeutige IDs, sofern der Anwender der Klasse keine eigene Meldungs-ID vorgibt.
- Die Klasse `MessageTemplate` wurde um die Methoden `withIdFactory` und `withLevel` erweitert, mit denen ein Lambda-Ausdruck für die Erzeugung der Meldungs-IDs sowie ein Debug-Level für die lokale Ausgabe der Meldung vorgegeben werden kann.
- Die Klasse `OperatingMessage` besitzt nun die neue Funktion `setMessageId`, mit der eine ID manuell vorgegeben werden kann.

6 Kernsoftware 3.7.4

Release-Datum: 22.04.2016

6.1 Bugfixes

Folgende Probleme vorhergehender Versionen wurden behoben:

- Beim Setzen von Text-Attributen in Datensätzen wurde die Übergabe von `null` statt einer Zeichenkette bisher akzeptiert, aber der anschließende Versand eines solchen Datensatzes führte zu einer schwer diagnostizierbaren `NullPointerException`. Das Setzen von `null`-Strings in `Data.TextValue`-Objekten wird nun direkt mit einer `NullPointerException` quittiert.
- Beim Senden von Betriebsmeldungen mit der in Version 3.7.3 neu eingeführten Klasse `OperatingMessage` konnte es zu einer `NullPointerException` kommen, weil als Message-ID für Einmal-Meldungen `null` verwendet wurde. Statt `null` wird nun ein Leer-String verwendet.

7 Kernsoftware 3.7.3

Release-Datum: 20.04.2016

7.1 Neue Features

Folgende Erweiterungen gegenüber vorhergehenden Versionen wurden durchgeführt:

- Die `de.bsvrz.sys.funclib.operatingMessage` wurde um verschiedene Klassen und Methoden erweitert, um komfortabler Betriebsmeldungen erzeugen zu können. Hierzu gibt es die neuen Klassen `OperatingMessage` und `MessageTemplate`, die den Anwender beim Erstellen von Meldungen und Befüllen von Platzhaltern in Meldungen unterstützen. Im einfachsten Fall können damit Meldungen bspw. wie folgt erzeugt werden:

```
OperatingMessage.warning(  
    MessageType.APPLICATION_DOMAIN,  
    "Die Güte am Fahrstreifen ", fs, " kann nicht bestimmt werden.",  
    exception  
).send();
```

Alle über die neuen Funktionen erzeugten Meldungen werden auch über die Debug-Funktionen mit Level INFO ausgegeben.

7.2 Bugfixes

Folgende Probleme vorhergehender Versionen wurden behoben:

- Nerz-FM-194: In empfangene Online-Datensätze enthaltene Referenzen auf lokal noch nicht bekannte Konfigurationsobjekte konnten zu Blockierungen von Applikationen und insbesondere dem Archivsystem in anschließenden Debugausgaben führen. In der geänderten Implementierung werden derartige Referenzen vor Übergabe des Online-Datensatzes an die Applikation durch entsprechende Konfigurationsanfragen aufgelöst.

- Nicht beantwortete Konfigurationsanfragen führten clientseitig bei manchen Anfragen nach 10 Minuten nicht mehr zu einem Timeout sondern zu einer Blockierung des anfragenden Threads.
- KExDaV hat empfangene Daten, die als *nachgeliefert* markiert waren, ohne die *Nachgeliefert*-Kennung weitergeleitet. In der geänderten Implementierung wird die *Nachgeliefert*-Kennung von empfangenen Datensätzen in die weitergeleiteten Datensätze übernommen.

8 Kernsoftware 3.7.2

Release-Datum: 05.02.2016

8.1 Änderungen

Folgende Änderungen zu vorhergehenden Versionen wurden durchgeführt:

- Beim aktiven Aufbau von Verbindungen zwischen zwei Datenverteilern wartet der Datenverteiler jetzt auch zwischen zwei erfolglosen Verbindungsversuchen (Kommunikationsfehler oder Authentifizierungsproblem) mindestens die in 3.6.0 implementierte Wiederverbindungswartezeit (Aufrufparameter: `-wiederverbindungswartezeit=60s`, Standardwert: 60 Sekunden). Bisher betrug die Wartezeit in diesem Fall eine Sekunde.

8.2 Bugfixes

Folgende Probleme vorhergehender Versionen wurden behoben:

- Beim Backup der Konfiguration über die Funktion `DataModel.backupConfigurationFiles()` konnte es dazu kommen, dass ein gemeinsames Wurzelverzeichnis der zu sichernden Dateien nicht korrekt ermittelt wurde, wenn sich die Aufrufargumente der Konfiguration (`-verwaltung`, `-benutzerverwaltung`) innerhalb von symbolischen Links befinden.
- Nerz-Fehler-174: In den Versionen 3.7.0 und 3.7.1 der `de.bsvrz.sys.funclib.losb` war unbeabsichtigt eine nicht kompatible Änderung des Kommunikationsprotokolls bei der Serialisierung von Long-Werten (IDs) enthalten. Hierdurch kommt es bei *Protokolle und Auswertungen* (PuA) zu Kommunikationsproblemen, wenn der Server eine andere Version dieser Bibliothek bzw. der Kernsoftware verwendet, als der Client. Mögliche Fehlermeldungen, die dadurch entstehen können, sind unter anderem:

Kann Daten nicht deserialisieren:

```
java.io.StreamCorruptedException: invalid stream header: 00000000
```

und:

Fehler beim Abruf eines Protokolls:

```
[..]FailureException: Protokoll nicht vorhanden: -5986128308230684672
```

Die Änderung wurde rückgängig gemacht. Wenn bei der Deserialisierung ein mit den betroffenen Versionen fehlerhaft serialisierter Wert erkannt wird, wird nun eine Warnmeldung ausgegeben.

- Beim Terminieren einer eingehenden Datenverteiler-Datenverteiler-Verbindung aufgrund einer deaktivierten Verbindung konnte ein Deadlock entstehen, wodurch die TCP-Verbindung nicht ordnungsgemäß geschlossen wurde. Mit der Zeit konnten sich hierdurch viele nutzlose Verbindungen ansammeln und damit Systemressourcen blockieren.

9 Kernsoftware 3.7.1

Release-Datum: 07.01.2016

9.1 Bugfixes

Folgende Probleme vorhergehender Versionen wurden behoben:

- Nerz-Fehler-170: Bedingt durch die in Java-8 geänderten Implementierungen von HashMap und HashSet ist aufgefallen, dass die beim Export von Konfigurationsbereichen verwendete Reihenfolge von nachfolgend aufgelisteten Elementen abhängig von der HashMap- und HashSet-Implementierung sind:
 - Elemente von Mengen
 - Zugelassene Mengentypen von Mengendefinitionen
 - Mengen von Konfigurationsobjekten

Der Export verwendet jetzt an diesen Stellen immer eine alphabetische Reihenfolge.

- Nerz-Fehler-171: Beim erneuten Import eines Bereichs konnten manchmal unnötige Objektversionen entstehen, die durch Rundungsfehler bedingt waren, wenn bei Zahl-Attributen weniger Stellen, als durch den Skalierungsfaktor vorgesehen, verwendet wurden.
- Beim Zugriff mit der Methode `Data.NumberValue.doubleValue()` auf skalierte Attributwerte von Attributen, die einen Skalierungsfaktor kleiner als 1 haben, konnten in manchen Fällen geringe Rundungsfehler entstehen, die durch eine geänderte Berechnungsformel vermieden werden konnten.
- Beim wiederholten Import von Konfigurationsbereichen ohne inhaltliche Änderungen, konnte es zu einer unkritischen Vergrößerung der Konfigurationsdatei kommen, wenn an einzelnen Konfigurationsobjekten Default-Parameter angegeben waren.

- Beim Export von Konfigurationsbereichen war das Element <modell...> eine Stufe zu weit eingerückt, wenn keine unversionierten Datenmodelländerungen im Bereich vorlagen.

10 Kernsoftware 3.7.0

Release-Datum: 03.12.2015

10.1 Neue Features

- Applikationen, die die `de.bsvrz.sys.funclib.application` benutzen, erhalten einen neuen Aufrufparameter `-terminierenBeiException=(ja/nein)` (Standardwert `nein`, entspricht bisherigem Verhalten), mit dem festgelegt werden kann, ob Applikationen sich bei jedem unbehandelten Fehler beenden sollen (`ja`), oder wie bisher nur bei schwerwiegenden Fehlern wie `OutOfMemoryError` (`nein`).

Mit Version 3.6.3 führten unbehandelte Exceptions im Updater-Thread im `ClientReceiverInterface` zu einem kontrollierten Beenden der Datenverteiler-Verbindung und damit auch typischerweise zum Beenden des Programms. Mit diesem Mechanismus kann dieses Verhalten nun optional auch auf alle anderen Threads ausgeweitet werden. Dadurch können beliebige Applikationen bei Fehlern jetzt automatisch (beispielsweise über `StartStopp`) neu gestartet werden, selbst wenn diese keine eigene Fehlerbehandlung durchführen.

- Der Parametereditor im GTM verwendet jetzt zum Kopieren und Einfügen die System-Zwischenablage. Dadurch können Parameterdaten zwischen verschiedenen GTM-Instanzen und Datenverteilersystemen ausgetauscht werden. Außerdem ist es möglich, die Datensätze in Texteditoren einzufügen und dort zu bearbeiten. Für diese Funktionalität werden die Datensätze in das [JSON-Format](#)¹ konvertiert. Um diese Funktionalität bereitzustellen besitzt der GTM daher die neue Abhängigkeit `de.kappich.sys.funclib.json`.
- Das Programm zum Terminieren von Verbindungen am Datenverteiler unter `de.bsvrz.dav.dav.main.TerminateConnection` besitzt jetzt den neuen Aufrufparameter `-wartezeit=(Zeitdauer)` mit dem die übergebenen Applikationen zeitlich versetzt terminiert werden können, beispielsweise um eine Überlastung des Systems zu vermeiden.

¹https://de.wikipedia.org/wiki/JavaScript_Object_Notation

- Nerz-Ä-60: Die verwendete Art der Rechteprüfung (Deaktiviert, Alt, Neu) wird jetzt vom Datenverteiler am Datenverteilerobjekt über die Quell-Attributgruppe `atg.datenverteilerRechteprüfung` am Aspekt `asp.standard` publiziert.
- Nerz-Ä-81: Bei Verwendung der alten Rechteprüfung erhalten alle Benutzer jetzt implizit alle Rechte, die auch in der neuen Rechteprüfung implizit vergeben werden (Konfigurationsanfragen, -antworten, Versand der Applikationsfertigmeldung, Betriebsmeldungen)
- Nerz-Ä-84: Über die neue Parameter-Attributgruppe `atg.deaktivierteVerbindungen` am Datenverteilerobjekt können jetzt einzelne Verbindungen zu anderen Datenverteilern deaktiviert werden. Ausgehende Verbindungen zu den angegebenen Datenverteilern werden terminiert bzw. gar nicht erst aufgebaut, eingehende Verbindungen von den angegebenen Datenverteilern werden während der Authentifizierung nach einem Timeout von einer Minute (um zu verhindern, dass sofort ein erneuter Verbindungsversuch gestartet wird) mit einer entsprechenden Fehlermeldung terminiert.
- Nerz-Ä-84: Über Debug-Ausgaben und über die neue Attributgruppe `atg.datenverteilerKommunikationsZustand` publiziert der Datenverteiler am Datenverteilerobjekt den aktuellen Kommunikationszustand mit den konfigurierten oder verbundenen Nachbardatenverteilern. Mögliche Zustände sind in der nachfolgenden Tabelle ([Tabelle 10.1](#)) aufgeführt.

Zusätzlich sind auch Informationen zur IP-Adresse des verbundenen Datenvertailers sowie eine Fehlermeldung vorhanden, wenn es sich um einen Fehlerzustand handelt.

- Nerz-Ä-88: Beim Backup der Konfigurationsdaten über die Funktion `DataModel.backupConfigurationFiles` werden jetzt die Konfigurationsdateien in der Struktur gesichert, in der sie sich auch im laufenden System befinden, d.h. Unterverzeichnisse werden beibehalten. Auf diese Weise kann ein Backup wieder problemlos in ein laufendes System eingespielt werden. Außerdem werden Dateien von dynamischen Mengen jetzt auch gesichert.

Tabelle 10.1: Mögliche Zustände der Kommunikation mit anderen Datenverteilern

Zustand	Beschreibung
Nicht verbunden	Verbindung ist nicht verbunden (Der Datenverteiler wartet auf neuen Verbindungsversuch)
Ersatzverbindung (nicht verbunden)	Verbindung ist nicht verbunden (Es handelt sich um eine Ersatzverbindung, welche nicht erforderlich ist)
Deaktiviert (nicht verbunden)	Verbindung ist nicht verbunden, da sie über <code>atg.deaktivierteVerbindungen</code> deaktiviert wurde
Warte auf eingehende Verbindung	Es wird auf Verbindungsaufbau durch den anderen Datenverteiler gewartet
Fehler	Verbindung ist nicht verbunden und wurde durch einen Fehler terminiert
Verbindungsaufbau Authentifizierung	Die Verbindung wird gerade aufgebaut Der Datenverteiler authentifiziert sich beim anderen Datenverteiler
Verbunden	Die Verbindung wurde hergestellt und ist aktiv
Verbindungsabbau	Die Verbindung wird gerade geschlossen und bestehende Anmeldungen werden abgemeldet (zum Beispiel Aufgrund eines Fehlers oder weil ein Terminierungstelegramm gesendet wurde)

10.2 Änderungen

- UTF-8: Die Java-Quellcode-Dateien von allen Distributionspaketen wurden mit UTF-8 statt mit ISO-8859-1 kodiert.
- GPL 3: Verwendung der GPL-Lizenz (bzw. LGPL-Lizenz) in Version 3 entsprechend NERZ Änderungsmanagement Eintrag 95
 - Die Lizenz von Distributionspaketen der Kernsoftware, die bisher unter GPL 2 standen wurde auf GPL 3 geändert
 - Die Lizenz von Distributionspaketen der Kernsoftware, die bisher unter LGPL 2.1 standen wurde auf LGPL 3 geändert
- Java-8: Alle Distributionspakete der Kernsoftware wurden entsprechend NERZ-Änderungsmanagement Eintrag 97 mit Java-8 übersetzt. Zu beachten ist, dass zur Laufzeit auch Java-8 vorausgesetzt wird.

10.3 Bugfixes

- Ein Fehler in 3.6.4 führte dazu, dass die angemeldeten Datenidentifikationen einer Applikation über die Attributgruppe `atg.angemeldeteDatenidentifikationen` sofort bei jeder Änderung veröffentlicht wurden, statt in einem 5-Sekunden-Intervall, was zu einer Überlastung des System führen konnte.
- Ein Deadlock in der Konfiguration konnte zu einer Blockierung beim Schreiben von dynamischen Mengendateien führen. Das konnte dazu führen, dass die Konfiguration für einzelne Applikationen keine Antworten mehr sendete und beim Beenden weitergelaufen ist.
- Ein weiterer Deadlock in der Konfiguration konnte bei der Ermittlung der zu bereinigenden dynamischen Objekte (Parameter `atg.parameterEndgültigesLöschen`) auftreten, wenn zeitgleich Konfigurationsanfragen nach gelöschten Objekten durchgeführt wurden.
- Ein Fehler beim Import der Konfiguration konnte dazu führen, dass Zeilenumbrüche und Einrückungen von Versorgungsdateien (beim erneuten Export) verloren gehen.

10.4 Datenmodelländerungen

Folgende Änderungen an Konfigurationsbereichen wurden durchgeführt:

- kb.systemModellGlobal Version 36:
 - Erweiterung des Datenverteiler-Typs um die Attributgruppen `atg.datenverteilerRechteprüfung`, `atg.deaktivierteVerbindungen` und `atg.datenverteilerKommunikationsZustand` für die oben beschriebene neue Funktionalität (Nerz-Ä-81 und Nerz-Ä-84)
- kb.fachModellGlobal Version 9:
 - Ergänzung um Attributtypen für Uhrzeit und Wochentag

11 Kernsoftware 3.6.5

Release-Datum: 22.05.2015

11.1 Bugfixes

- Nerz-F-147: Der Zustand *keine Quelle* wurde nicht immer korrekt im CacheManager in der DAF-API gespeichert. Dadurch konnten veraltete Daten an einen neu angemeldeten Empfänger ausgeliefert werden, wenn sich dieser zu einem Zeitpunkt anmeldet, an dem die Quelle nicht erreichbar ist.
- Es konnte zu einem Deadlock in den Datenverteiler-Applikationsfunktionen kommen, wenn eine Applikation sich während des Verbindungsaufbaus (z.B. wegen eines Fehlers) beenden wollte oder die noch nicht fertig initialisierte Verbindung zum Datenverteiler terminieren wollte.

12 Kernsoftware 3.6.4

Release-Datum: 10.04.2015

12.1 Bugfixes

Folgende Probleme vorhergehender Versionen wurden behoben:

- Bei der im Rahmen von Nerz-Ä-28 eingeführten effizienteren Speicherung von dynamischen Mengen (siehe unten) konnte es zu fehlerhaften Mengendateien kommen, die beim erneuten Start der Konfiguration zu Exception geführt haben.
- Bei der Publikation von Datensätzen der Attributgruppe `atg.angemeldeteDatenidentifikationen` durch den Datenverteiler konnte es in besonderen Fällen zu Exceptions kommen, weil das Systemobjekt von einzelnen Anmeldungen nicht durch den Datenverteiler aufgelöst werden konnte. In diesen Fällen wird vom Datenverteiler jetzt eine entsprechende Warnung ausgegeben und der Datensatz ohne die problematischen Einträge publiziert.
- Eine fehlerhafte Überprüfung der aktivierten Datenmodellversion hat dazu geführt, dass die neuen Konfigurationsanfragen aus Nerz-Ä-28 (siehe unten) auch nach Aktualisierung des Systemmodells nicht immer verwendet wurden.
- Bei der expliziten Abfrage von allen Objekten vom Typ `Ganzzahlattributtyp` und Attributlistendefinitionen kam es zu Serialisierungsfehlern.
- Bei Objektanfragen an fremde Konfigurationen mit `ClientDavConnection.getDataModel(configAuthority)` konnte es beim Einsatz der neuen Konfigurationsanfragen aus Nerz-Ä-28 (siehe unten) dazu kommen, dass die entsprechenden Objekte aus der lokalen Konfiguration zurückgeliefert wurden.
- Es konnten Deadlocks auftreten, wenn mehrere Datenverteiler-Verbindungen in einer JVM aufgebaut wurden.
- In seltenen Fällen trat beim Beenden des Datenvertailers ein Deadlock auf, der dazu führte, dass der Datenverteiler sich nicht beenden konnte.

- Es konnte zu Deadlocks beim Start einer Applikationen kommen, wenn der Datenverteiler sich während der Authentifizierung der Applikation beendet hat.

13 Kernsoftware 3.6.3

Release-Datum: 08.03.2015

13.1 Änderungen

Folgende Änderungen zu vorhergehenden Versionen wurden durchgeführt:

- Nerz-F-14: Die Pids von dynamischen Objekten werden nun in der Konfiguration nach Simulationsvariante getrennt verwaltet. Dadurch ist es nun möglich, in Simulationen dynamische Objekte zu erstellen, deren Pid bereits außerhalb von Simulationen oder in anderen Simulationen verwendet wird.
 - Anfragen nach dynamischen Objekten anhand der Pid wie z. B. `DataModel.getObject(pid)` geben nun bevorzugt Objekte der eigenen Simulationsvariante zurück. Gibt es in der Simulation kein passendes Objekt können auch normale nicht-simulierte Objekte zurückgegeben werden, aber nur wenn der Typ des Objektes nicht in der Simulationsstrecke eingetragen ist.
 - Allgemein werden Simulationen besser voneinander getrennt, d. h. Objekte, die in der aktuellen Simulation ungültig sind, werden nun bei verschiedenen Konfigurationsanfragen ausgeblendet. Normale, nicht-simulierte Applikationen sehen keine in Simulationen erstellen Objekte mehr.
 - Damit die Konfiguration bei Konfigurationsanfragen die Simulationsvariante der anfragenden Applikation korrekt ermitteln kann, initialisieren Applikationen in Simulationen beim Login nun die Kommunikation mit der Konfiguration. Das führt dazu, dass solche Applikationen nun solange im Login blockieren bis die Simulation mindestens den Zustand "Vorstart" erreicht hat.
- Nerz-Ä-28: Es wurden konfigurationsseitig und clientseitig verschiedene Verbesserungen bei der Speicherung von dynamischen Objekten und Mengen durchgeführt.
 - Verbesserungen der Performance und des Speicherverbrauchs durch geeignete Datenstrukturen.

- Dynamische Mengen, die mit `verwaltung="kv..."` im Dateisystem gespeichert werden, werden konfigurationsseitig effizienter verwaltet. Beim Einfügen und Löschen von Elementen werden nur die Bereiche der Datei neu geschrieben, für die sich Änderungen ergeben haben.
- Die Restrukturierung der Konfigurationsdateien kann nun offline manuell gestartet werden, indem die Konfiguration mit den Aufrufparametern `-restrukturierung` und `-verwaltung=...` gestartet wird. Dies ist hilfreich, wenn sich in einer Konfigurationsdatei viele gelöschte dynamische Objekte befinden, da dadurch die Datenstrukturen optimiert werden.
- Die Geschwindigkeit der Restrukturierung wurde wesentlich verbessert.
- Ebenfalls ist eine Restrukturierung im laufenden Betrieb möglich. Hierzu kann am AOE-Objekt über die neue Parameter-Attributgruppe `atg.parameterEndgültigesLöschen` ein Cron-String definiert werden, der angibt, zu welchen Zeitpunkten (beispielsweise: wöchentlich oder täglich) eine Restrukturierung im Hintergrund stattfinden soll. Bei der Restrukturierung in laufenden Betrieb werden nur die dynamischen Objekte restrukturiert.
- Über denselben Parameter können auch Vorhaltezeiträume für dynamische Objekte und dynamische Mengen vorgegeben werden. Hiermit können gelöschte dynamische Objekte und historische Mengenreferenzen nach einer gewissen Zeit (beispielsweise 1 Jahr) vollständig und endgültig gelöscht werden, um nicht mehr benötigte Daten zu löschen, Speicherplatz freizugeben, und die Performance zu verbessern. Dynamische Objekte werden nur dann gelöscht, wenn sie von keinem gültigen Objekt mehr referenziert werden und wenn sie mindestens der Vorhaltezeitraum nach dem Löschen vergangen ist.
 - * Objekte werden im laufenden Betrieb als zu Löschen markiert und erst beim nächsten Neustart der Konfiguration vollständig gelöscht. Die zu löschenden Objekte werden je Bereich in einer Datei gespeichert, die sich im gleichen Ordner wie die Konfigurationsdateien befindet und die Dateiendung `“.0.index“` oder `“.1.index“` besitzt.
- Nerz-Ä-28: Es wurden weitere Konfigurationsanfragen implementiert, mit denen ermöglicht wird, dass die Konfiguration verschiedene Anfragen wie die Abfrage von Objekten anhand von PID, ID oder Typ oder das Erstellen und Löschen von Objekten asynchron ausführen kann. Hierdurch kann die Performance der Konfiguration verbessert werden.
- Nerz-Ä-50: Mit dem neuen Aufrufparameter `-zweiteVerbindung=ja` für Applikationen wird jetzt für die Durchführung von Konfigurationsanfragen eine zweite Datenverteilterverbindung aufgebaut. Hiermit kann in Hochlast-Situationen die Gefahr von Blockierungen vermieden werden. Diese Funktion befindet sich in der Erprobungsphase und ist noch standardmäßig deaktiviert.

- Diese zweite Verbindung wird durch eine Raute (#) gekennzeichnet, die an den Applikationsnamen angehängt wird.
 - Über den Parameter `-zweiteVerbindungPufferAnteil=<Wert>` lässt sich der Anteil an den Sende- und Empfangs- Puffern vorgeben, der der zweiten Verbindung zugewiesen wird. Der Wert muss zwischen 0,0 und 1,0 liegen und beträgt standardmäßig 0,01 (1%).
 - Für Applikationen, bei denen eine eigene Verbindung für Konfigurationsanfragen wenig Sinn machen würde, wie die Konfiguration selbst, wird der Parameter ignoriert.
- Änderungen an den `ClientDavParameters` waren bisher auch möglich, nachdem eine `ClientDavConnection` initialisiert wurde. Um Problemen vorzubeugen, die entstehen könnten, wenn im laufenden Betrieb wesentliche Parameter geändert werden, gibt es jetzt folgende Einschränkungen:
 - Wenn eine zweite Verbindung benutzt wird, wird im Konstruktor der `ClientDavConnection` eine Kopie der `ClientDavParameters` erzeugt. Weitere Änderungen an dem ursprünglichen Objekt wirken sich nicht auf die Verbindung aus. Das über `connection.getClientDavParameters()` zurückgegebene Objekt ist schreibgeschützt, der Aufruf von Setter-Methoden führt zu Exceptions.
 - Wenn keine zweite Verbindung benutzt wird, verhalten sich die Setter-Methoden der `ClientDavParameters` (aus Kompatibilitätsgründen) wie bisher. Beim nachträglichen Ändern von Parametern, die bereits einer `ClientDavConnection` zugewiesen wurden, wird aber nun eine Warnung ausgegeben. Betroffene Codestellen sollten so geändert werden, dass die `ClientDavParameters` vollständig gesetzt werden, bevor die `ClientDavConnection` erzeugt wird. Programme, die zur Initialisierung die Frameworks `de.bsvrz.sys.funclib.application` oder `de.kappich.sys.funclib.configurationtool` benutzen, sind in der Regel bereits kompatibel. Da die Verwendung von zwei Verbindungen in Zukunft Standard werden könnte, sollten alle Applikationen, sofern nötig, angepasst werden.
 - Unbehandelte Exceptions in der `update()`-Methode im `ClientReceiverInterface` führen jetzt zu einem kontrollierten Terminieren der Datenverteilerverbindung.
 - Da Eclipse nicht mit zirkulären Abhängigkeiten zwischen Plugins umgehen kann, wurden die Inhalte der Distributionspakete `de.bsvrz.sys.funclib.dataSerializer` und `de.bsvrz.sys.funclib.communicationStreams` in das Paket `de.bsvrz.dav.daf`

integriert. Die beiden Distributionspakete bleiben aus Kompatibilitätsgründen als leere Pakete erhalten.

- Das `DataModel`-Interface wurde um Methoden erweitert, welche mehrere Objekte gleichzeitig anhand der PID oder ID anfordern können. Hierdurch kann eine bessere Performance erreicht werden, als wenn jedes Objekt einzeln abgefragt würde.
- Das `ConfigurationAuthority`-Interface wurde um eine Methode erweitert, die den Default-Bereich des Konfigurationsverantwortlichen zurückgibt.

13.2 Bugfixes

Folgende Probleme vorhergehender Versionen wurden behoben:

- Wenn eine Simulation beendet wird, werden alle dazugehörigen dynamischen Objekte automatisch gelöscht. Beim Löschen wurden verschiedene Datenstrukturen konfigurationsseitig nicht aktualisiert, sodass bei Anfragen wie `DynamicObjectType.getElements()` oder `DataModel.getObjects(...)` solche gelöschten Objekte dennoch zurückgegeben werden konnten.
- Das Ändern von (änderbaren) Konfigurationsdaten von gelöschten Objekten konnte zu Fehlern in den Datenstrukturen der Konfigurationsdatei führen, wodurch es beim Zugriff auf diese gelöschten Objekte zu Exceptions kommen konnte. Aus diesem Grund dürfen Konfigurationsdaten nur noch für Objekte geändert werden, die noch nicht gelöscht wurden. Durch die in Nerz-Ä-28 implementierte Offline-Restrukturierung können beschädigte Konfigurationsdateien automatisch repariert werden. (Aufrufparameter: `-restrukturierung`)
- Beim Setzen von Konfigurationsdaten wird nun geprüft, ob der Typ die verwendete Attributgruppe unterstützt. Ist das nicht der Fall, wird nun eine Exception generiert, statt den Datensatz dennoch (versteckt) zu speichern.
- Beim Setzen von `ReferenceValues` im `Data`-Interface wird nun geprüft, ob der Attributtyp den Typ des zu speichernden Objekts unterstützt.
- `KExDaV` verhält sich besser, wenn die Remote-Konfiguration nicht antwortet oder keine Rechte zum Stellen von Konfigurationsanfragen vorhanden sind. In dem Fall wird die Verbindung jetzt sofort terminiert, statt im Hintergrund weiter zu laufen.
- In der Datenverteilerkommunikation konnte es zu Blockierungen kommen, wenn die Sende- und/oder Empfangspuffer kleiner als 3000 Bytes eingestellt wurden.
- Eine `NullPointerException` wurde korrigiert. Sie konnte in der alten Rechteprüfung auftreten, wenn keine Parameterdaten zu einer Berechtigungsklasse vorlagen.

- Der Import und Export der Einstellungen in der Generischen Netzdarstellung (GND) wurde korrigiert.
- Der Deserializer in `de.bsvrz.sys.funclib.dataSerializer` liefert jetzt keine `EOFExceptions` mehr, wenn kein `ByteArrayInputStream` sondern andere `InputStreams` benutzt werden (bzw. der `InputStream` bei `read(byte[])` nicht den kompletten Puffer füllt).
- Im laufenden Betrieb gelöschte dynamische Objekte wurden bis zum nächsten Neustart der Konfiguration nicht von der Methode zum Abruf aller Objekte mit der gegebenen `Pid` im Zeitbereich zurückgegeben (`DataModel.getObjects(pid, startTime, endTime)`).

13.3 Datenmodelländerungen

Folgende Änderungen an Konfigurationsbereichen wurden durchgeführt:

- `kb.systemModellGlobal`:
 - Erweiterung der Konfigurationsanfragen und -antworten um weitere Anfrage- bzw. Antworttypen mit denen ermöglicht wird, dass die Konfiguration die entsprechenden Anfragen asynchron ausführen kann.
 - Fehlerkorrektur: In der Attributliste `atl.aktivitätMengen` wurde der Attributtyp des Attributs `Mengentyp` geändert. Statt `att.typMengeReferenz` wird hier jetzt korrekte `att.mengenTypReferenz` verwendet. Konfigurationsbereiche mit der Versorgung von Zugriffsrechten und Defaultparameter dazu können mit einem einfachen Import aktualisiert werden.
- `kb.systemModellAoe`: Neue Parameter-Attributgruppe für AOE-Objekte mit der folgende zyklische Aktionen in der Konfiguration eingestellt werden können (siehe auch *Nerz-Ä-28* oben):
 - die Restrukturierung im laufenden Betrieb
 - das endgültige Löschen von dynamische Objekte
 - das endgültige Löschen von historische Elementreferenzen in dynamischen Mengen

14 Kernsoftware 3.6.2

Release-Datum: 20.03.2014

14.1 Bugfixes

Folgende Probleme vorhergehender Versionen wurden behoben:

- Durch ein Synchronisationsproblem im Datenverteiler konnte es in seltenen Fällen dazu kommen, dass in den vergebenen Datensatzindizes von versendeten Datensätzen ein Rücksprung enthalten war, der dann beim Empfang durch eine Applikationen zu Fehlermeldungen (*Empfangener Datensatz hat ungültigen Datensatzindex*) und zu Blockierungen führen konnte.

15 Kernsoftware 3.6.1

Release-Datum: 13.03.2014

15.1 Bugfixes

Folgende Probleme vorhergehender Versionen wurden behoben:

- GTM-Plugin für generische Netzdarstellung hat beim Start eine `NullPointerException` geworfen. Außerdem war ein Fehler im entsprechenden Start-Skript.
- Beim Wechsel einer Quellenmeldung vom bisherigen Quelldatenverteiler auf einen anderen Datenverteiler, der zwischen bisherigem Quelldatenverteiler und einem Empfangsdatenverteiler lag, konnte es zu Kommunikationsausfällen kommen.
- Der Start des Datenverteilers mit dem Aufrufargument `-warteAufParametrierung=ja` hat in Version 3.6.0 fälschlicherweise nicht mehr dazu geführt, dass die Authentifizierung von normalen Applikationen solange verzögert wird, bis die Parametrierung ihre Fertigmeldung versendet hat.
- Bei fehlgeschlagenen Verbindungsversuchen einer Applikation zum Datenverteiler mit der Methode `ClientDavConnection.connect()` blieb ein Thread mit Namen `SendControlNotifier` bestehen.

16 Kernsoftware 3.6.0

Release-Datum: 25.02.2014

Status: Diese Version ist als Beta-Release zu betrachten, weil in der Testphase zwei sehr selten aufgetretene Probleme zwar bearbeitet, aber noch nicht abschließend geklärt werden konnten:

- Alle paar Wochen: Terminieren des Archivsystems.
- Alle paar Wochen: Datenverteilung über mehrere Datenverteiler funktioniert nach Kommunikationsunterbrechungen nicht korrekt.

16.1 Neue Features

Folgende Erweiterungen gegenüber vorhergehenden Versionen wurden durchgeführt:

- Nerz-Ä-16: Performance-Verbesserungen bei der Verwendung von ByteArrays in den Datenverteiler-Applikationsfunktionen. In den verschiedenen Implementierungsvarianten des Data-Interfaces wurden neue Methoden zum Setzen eines `byte[]`, `short[]`, `int[]`, etc. integriert, die das Data-Array auf die Größe des übergebenen byte-Arrays (etc.) einstellen und dann die Werte des übergebenen byte-Arrays kopieren ohne dabei die Defaultwerte zu setzen. In den Datenverteiler-Applikationsfunktionen und anderen Teilen der Kernsoftware wurden entsprechende Anwendungsfälle in der Kommunikation mit der Konfiguration, dem Archivsystem und PuA auf die neuen Methoden umgestellt.
- Nerz Ä-21: Der Datenverteiler wurde so erweitert werden, dass er vor erneuten Verbindungsversuchen zur Kopplung von Datenverteilern eine einstellbare Zeit wartet. Die Wartezeit kann über den neuen Aufrufparameter `-wiederverbindungsWartezeit=60s` eingestellt werden. Defaultwert ist 60 Sekunden.
- Nerz-ID Ä-22: Im Datenverteiler wurde eine Erweiterung durchgeführt, die bewirkt, dass der Datenverteiler versucht, die Menge "Applikationen" am entsprechenden Konfigurationsobjekt des Datenvertailers zu aktualisieren, wenn Applikationen sich anmelden bzw. wieder abmelden. Damit dies funktioniert, muss die lokale AOE die Verantwortung zur Speicherung der Elementzugehörigkeit der Menge

Applikationen besitzen. Dies kann bei der Versorgung der Datenverteilerobjekte durch das XML-Attribut “verwaltung” bei der Definition der konkreten Menge realisiert werden.

- Nerz-Ä-6: Bei versionierten Konfigurationsänderungen werden für die betroffenen Objekte neue Konfigurationsobjekte mit neuer ID und gleicher Pid in der Konfiguration erzeugt. Um eine Archivanfrage für ein so geändertes Objekt über den Versionswechsel hinaus durchzuführen, muss die anfragende Applikation bisher die verschiedenen Konfigurationsobjekte mit der gewünschten Pid im Anfragezeitbereich ermitteln und alle gefundenen Objekte in der Archivanfrage angeben.
 - Das API für Archivanfragen wurde so erweitert, dass automatisch die im Zeitbereich gültigen Objekte ermittelt und in der Anfrage ans Archivsystem übertragen werden, wenn die anfragende Applikation die Pid des gewünschten Objekts in der Anfrage angibt.
 - Der Systemprotokollierer wurde so erweitert, dass er das neue API für Archivanfragen verwendet.
 - Der Generische-Test-Monitor (GTM) wurde so erweitert, dass er das neue API bei Archivanfragen verwendet und in der Objektspalte, das jeweilige historische Objekt darstellt und im Tooltip angezeigt.
- Nerz Ä-26/Ä-58: Unversionierte Datenmodelländerungen. Die Konfiguration kann beim Import von Attributgruppen kleinere Änderungen von einzelnen Attributen, die nicht zu einer Veränderung der bei der Serialisierung benötigten Anzahl von Bytes führen, ohne Versionierung und damit auch ohne die Vergabe von neuen Objekt-IDs durchführen. Nur bei größeren Datenmodelländerungen, die bei der Serialisierung zu strukturellen Änderungen oder zu einer Veränderung der benötigten Anzahl von Bytes führen, wird nach wie vor der Mechanismus der Versionierung eingesetzt werden.
- Nerz-F-67: Neues Commandline-Tool, mit dem man die jeweils aktivierbare Version einer oder mehrerer Konfigurationsdateien anzeigen kann. Über einen Aufrufparameter kann man dem Tool einzelne oder mehrere Konfigurationsdateien oder den Namen eines Ordners mit zu berücksichtigenden Konfigurationsdateien übergeben. Das Tool gibt zu jedem Konfigurationsbereich die zuletzt zur Aktivierung freigegebene Version und auf Wunsch weitere Informationen (gesteuert über entsprechende Ausgabe-Flags) aus.

Aufruf:

```
java de.bsvrz.puk.config.configFile.util.ConfigFileInfo  
[Optionen]... [Konfigurationsdateien/Ordner]...
```

Gültige Optionen: (Umlaute können bei Bedarf durch ae, oe, ue ersetzt werden)

- kurzInfo=ja** Kurzinfos ausgeben
- beschreibung=ja** Beschreibungen ausgeben
- übernahmeVersion=ja** Zur Übernahme freigegebene Version ausgeben
- aktivierungVersion=nein** Zur Aktivierung freigegebene Version nicht ausgeben
- abhängigkeiten=ja** Abhängigkeiten ausgeben
- änderungen=ja** Änderungsvermerke ausgeben
- zusammenfassung=nein** Zusammenfassung über benötigte Konfigurationsbereiche nicht ausgeben (nur bei mehreren Dateien)
- alles=ja** Alles oben genannte ausgeben
- terminalBreite=80** Breite der Ausgabe anpassen
- ausführlich=ja** Ausführliche Ausgabe der Abhängigkeiten und Änderungsvermerke
- Nerz-Ä-55: Sicherung der Konfiguration für nur einen Konfigurationsverantwortlichen.
 - Die Erstellung von Sicherungskopien der Konfigurationsbereiche wurde in der Konfiguration dahingehend erweitert, dass es optional auch möglich ist, nur die Konfigurationsbereiche eines vorgegeben Konfigurationsverantwortlichen zu sichern.
 - Erweiterung des API in den Datenverteiler-Applikationsfunktionen zur Steuerung des Sicherungsmechanismus um eine weitere Methode, die einen zusätzlichen Parameter aufnimmt, über den ein Konfigurationsverantwortlicher vorgegeben werden kann, dessen Konfigurationsbereiche gesichert werden sollen.
 - Erweiterung des Command-Line-Tools zum einfachen manuellen Anstoßen der Sicherung um einen entsprechenden (optionalen) Aufrufparameter.

16.2 Änderungen

Folgende Änderungen zu vorhergehenden Versionen wurden durchgeführt:

- Nerz-F-13: Die Implementierung der `getElements()` Methode von dynamischen Typen wurde so geändert, dass sie die für die jeweilige Simulation relevanten Objekte zurück liefert und nicht mehr die vorhandenen Elemente aller Simulationsvarianten zurückliefert.

- Nerz-Ä-51: Die Implementierung der Anmeldung als Sender in den DAF wurde so erweitert, dass mehrere Anmeldungen als Sender für die gleiche Datenidentifikation zugelassen werden.
- Nerz-Ä-54: Wenn eine Applikation bei der Anmeldung auf Daten keine Rechte dazu hatte, dann blieb die Anmeldung nicht bestehen, sondern wurde automatisch zurückgenommen.
 - Refactoring der Java-Klassen `ConnectionsManager` und `SubscriptionsManager`, d.h. Überarbeitung der Struktur der Klassen, so dass sie modularer, klarer und übersichtlicher organisiert sind, ohne dabei das aktuelle Verhalten zu ändern.
 - Änderung des Verhaltens des Datenverteilers, sodass Anmeldungen, die wegen fehlenden Rechten nicht aktiviert werden können, im Datenverteiler erhalten bleiben und nach entsprechender Änderung der Rechte automatisch (d.h. ohne erneute Anmeldung der betroffenen Applikation) aktiviert werden.
 - Realisierung einer Analysemöglichkeit im Datenverteiler, mit der Applikationen und Testfälle den Status von einzelnen Anmeldungen abfragen können.
- Nerz-Ä-12: Über eine neue API-Methode `getNumArchiveQueries()` in der Klasse `ArchiveRequestManager` kann eine Applikation die aktuell im Archivsystem eingestellte maximale Anzahl von gleichzeitig für eine Applikation bearbeiteten Archivfragen zur Laufzeit abfragen.
- Nerz-Ä-17: Änderung der Namen von Objekten. Die Konfiguration wurde so geändert, dass der Name eines noch nicht aktivierten Konfigurationsobjekts geändert werden kann, auch wenn der Name laut Datenmodell nicht änderbar ist.
- Nerz-F-96 `MessageSender` funktioniert erst nach einer unbestimmten Zeit. Die applikationsseitig verwendbare Bibliothek zum Senden von Betriebsmeldungen (Klasse `MessageSender` im Distributionspaket `de.bsvrz.sys.funclib.operatingMessage`) wurde so erweitert, dass zu versendende Nachrichten bis zu einer vorgebbaren Anzahl (Defaultwert: 100 Meldungen) zwischengespeichert werden, wenn die Betriebsmeldungsverwaltung nicht (mehr) verfügbar ist. Zwischengespeicherte Meldungen werden automatisch ohne weiteres Zutun der Applikation versendet, sobald die Betriebsmeldungsverwaltung (wieder) verfügbar ist.

16.3 Bugfixes

Folgende Probleme vorhergehender Versionen wurden behoben:

- Korrektur eines Problems: In der Konfiguration wurde in manchen Fällen das Aufrufargument zum Ignorieren der Abhängigkeiten zwischen Konfigurationsbereichen `-ignoriereFehlerDerAbhaengigkeitenBeiKonsistenzpruefung=ja` zu spät gesetzt.
- Korrektur eines Problems: Im Osi3-Modul von KEx-TLS war ein Logikfehler bei der Auswertung der Einstellung `osi3.reduzierungAufSteuerbyte`, der dann aufgetreten ist, wenn nicht die Defaulteinstellung verwendet wurde.
- Nerz-F-28: Im `UncaughtExceptionHandler` des `StandardApplicationRunners` werden beim Auftreten von nicht behandelten Exceptions jetzt Ausgaben mit Debug-Level FEHLER statt WARNUNG verwendet. In allen Applikationen der Kernsoftware, die einen eigenen `UncaughtExceptionHandler` haben, wurde eine analoge Anpassung durchgeführt.
- Nerz-F-65: Beim Import von aktualisierten Versorgungsdateien, in denen Defaultparameter enthalten waren, in denen auf neue importierte Objekte referenziert wurde, kam es zu Exceptions und abgebrochen Importvorgängen. Die Ursache für diese Probleme wurde korrigiert.

16.4 Datenmodelländerungen

Folgende Änderungen an Konfigurationsbereichen wurden durchgeführt:

- `kb.metaModellGlobal`: Neue Attributgruppe am Konfigurationsbereich ergänzt, die Informationen über unversionierte Datenmodelländerungen enthält.
- `kb.tmVewBetriebGlobal`: Erweiterung der SWE Betriebsmeldungsverwaltung gemäß NERZ Änderungsantrag 34 und eines entsprechenden konkreten Vorschlags um Kriterien für die Filterung. Um den Filter in einer Anwendung auch tatsächlich verwenden zu können, wurde der AttributTyp `att.vewBetriKriterium` um den Zustand `Meldungstext` erweitert.
- `kb.systemModellObsolet`: Anpassung des Änderungskommentars und der Beschreibungen.
- `kb.systemModellGlobal`: Erweiterung der Archivanfragen um eine Möglichkeit die maximale Anzahl Archivanfragen pro Applikation zu ermitteln.
- `kb.systemModellGlobal`: Korrektur des Datenmodells für die neuen Zugriffsrechte.
- `kb.objekteTestSystem`: Bei der Menge `Applikationen` an den Datenverteiler-Objekten wurde festgelegt, dass die Elementzugehörigkeit von der AOE `kv.testKonfiguration` verwaltet wird, damit die Applikationsobjekte im Beispielsystem zur Laufzeit auch in die Menge aufgenommen und entfernt werden können.

17 Ältere Releases

Die Release-Notes zu älteren Kernsoftware-Releases bis Release 3.5.5 befinden sich in einer [separaten Datei](#)¹.

¹[release-notes-alt.html](#)