

# AK VRZ

## Verkehrsrechnerzentralen

### SW-Architektur

#### Segment 01 - Datenverteiler

#### SW-Einheit Datenverteiler Applikationsfunktionen

Ersteller:



Autor:

Dipl.-Ing. H.C. Kniß  
Dipl.-Ing. F. Labtaini  
Dipl.-Inform. R. Schmitz  
Dipl.-Ing. C. Westermann

Version:

1.0

Stand

15.03.2002

Status:

akzeptiert

PID:

SE-02.01.01.00.00-SWArc-1.0

Submodell:

Systementwicklung

Dokument:

SE-02.01.01.00.00-SwArc-1.0 [SwArc DAF].doc

VS-Einstufung:

----

Projekt ID AG:

AK VRZ

Projekt ID AN:

BAST.06.006 / BAST.07.007

Kappich+Kniß Systemberatung Verkehr und Technik

Im Auftrag der Bundesanstalt für Straßenwesen BAST

## 1 Allgemeines

### Verteilerliste

Entfällt. Dokumentverteilung entsprechend aktuellem Projektverteiler.

### Versionsübersicht

Nr.	Datum	Version	Änderungsgrund	Bearbeiter
1	31.01.02	0.1	Ersterstellung	Knifß Schmitz Westermann
2	15.02.02	0.2	Einarbeitung der Änderungsvorschläge vom 19. PAWG	Schmitz
3	15.03.02	1.0	Überführung in den Status "akzeptiert"	Westermann

Tabelle 1-1: Versionsübersicht

### Änderungsübersicht

Nr.	Version	geändertes Kapitel	Beschreibung der Änderung
1	0.1	alle	Ersterstellung
2	0.2	alle	Rechtschreibkorrektur
3	0.2	3.1 3.4	Langbezeichnungen in Tabelle 3-1 und Tabelle 3-2 korrigiert.
4	0.2	3.2.1.x	Bezeichnung Modul in Überschriften aufgenommen.
5	0.2	3.2.1.3 3.2.1.4	Die Funktionalitäten der Zerlegung von großen Datenblöcken in mehrere Datentelegramme beim Senden und die Rekombination von zusammengehörigen Datentelegrammen zu einem Datenblock in das Modul Protokollsteuerung DAF-DaV verschoben.
6	0.2	3.2.2.3	Fehlerbehandlung detailliert und als zusätzliche Fehlerquelle die Kommunikation mit der Konfiguration aufgeführt. Abbildung 3-4 überarbeitet.
7	0.2	3.2.2.4.1	Vorgabe der Simulationsvariante verdeutlicht.
8	0.2	3.2.2.4.2	Reaktion auf die Sendesteuerung bei impliziter Anmeldung verdeutlicht.
9	0.2	3.2.2.4.2	Es wurde ergänzt, dass Änderungsinformationen mit dem Datentelegramm an den Datenverteiler übertragen werden. Ein Fußnote wurde ergänzt, die beschreibt, wie der Datenverteiler mit den Markierungen umgeht.
10	0.2	3.2.2.4.4	Überschrift "Entgegennahme von angemeldeten und historischen Daten" in "Entgegennahme von angemeldeten Daten" geändert.
11	0.2	3.2.2.4.5	Historische Daten durch zwischengespeicherte Daten ersetzt.
12	0.2	3.2.2.4.6	Letzten Absatz gestrichen.

Nr.	Version	geändertes Kapitel	Beschreibung der Änderung
13	0.2	3.2.2.4.7	"keine Berechtigung" als mögliche Fehlerkennung ergänzt. Zugriff auf die Attribute verdeutlicht.
14	0.2	3.2.2.5	Erkennung von bestimmten Fehlerzuständen ergänzt.
15	0.2	3.2.2.5.1	Schnittstelle durch Applikationsschnittstelle ersetzt.
16	0.2	3.2.2.5.3	"Aufgaben" durch "Funktion" ersetzt.
17	0.2	3.2.2.6	Verdeutlicht, dass die Interaktion mit der "zugeordneten" Datenhaltung stattfindet.
18	0.2	3.2.2.7	Verdeutlicht, dass die Zustandsinformationen von beliebigen Datenverteilern im Netz abgerufen werden können.
19	0.2	5	Anforderungszuordnung gemäß PAWG überarbeitet.
20	0.2	3.2.1.2	Differenzierung des aktiven und passiven Verbindungsaufbaus über TCP zur allgemeinen Verwendung des TCP-Kommunikationskanals in TCP-Clients und auch TCP-Servern.

Tabelle 1-2: Änderungsübersicht

## Kurzbeschreibung

In diesem Produkt wird die SW-Architektur der SW-Einheit Datenverteiler Applikationsfunktionen beschrieben.

## Inhalt

<b>1 Allgemeines</b>	<b>2</b>
Verteilerliste	2
Versionsübersicht	2
Änderungsübersicht	2
Kurzbeschreibung	3
Inhalt	4
Abkürzungen	6
Definitionen	6
Verzeichnis der Tabellen	6
Verzeichnis der Abbildungen	6
Referenzierte Dokumente	7
<b>2 Lösungsvorschläge</b>	<b>8</b>
<b>3 Modularisierung/Datenbankentwurf</b>	<b>9</b>
3.1 Übersicht der SW-Komponenten, SW-Module, Prozesse und Datenbanken	9
3.2 Einzelbeschreibungen	10
3.2.1 Komponente Kommunikation	10
3.2.1.1 Modul Datenrepräsentation	10
3.2.1.2 Modul TCP-Kommunikationskanal	11
3.2.1.3 Modul Telegrammverwaltung	11
3.2.1.4 Modul Protokollsteuerung DAF-DaV	12
3.2.1.5 Modul HMAC-MD5 Verschlüsselung	14
3.2.2 Modul Verwaltung	15
3.2.2.1 Plausibilitätsprüfung der Parameter der Applikationsschnittstelle	15
3.2.2.2 Auswertung der Aufrufparameter	15
3.2.2.3 Verwaltung der logischen Verbindung zum Datenverteiler	15
3.2.2.4 Funktionen der Datenverteilung	17
3.2.2.4.1 Verwaltung von Empfangs- und Sendeanmeldungen	17
3.2.2.4.2 Senden von Anwendungsdaten	19
3.2.2.4.3 Sendesteuerungen an Applikation weiterleiten	19
3.2.2.4.4 Entgegennahme von angemeldeten Daten	19
3.2.2.4.5 Zwischenspeicherung von Daten	20
3.2.2.4.6 Lesen von Daten	20
3.2.2.4.7 Zugriff auf Datensätze	20
3.2.2.5 Konfiguration	21
3.2.2.5.1 Stellvertreterobjekte	22
3.2.2.5.2 Metamodell	22
3.2.2.5.3 Abfrage des Metamodells	22
3.2.2.5.4 Abfrage von Konfigurationsobjekten	22
3.2.2.5.5 Lesen von konfigurierenden Daten	22
3.2.2.5.6 Schreiben von konfigurierenden Daten	23

**Allgemeines**

---

3.2.2.5.7	Erzeugen und Löschen von Objekten	23
3.2.2.5.8	Änderung von Mengen	23
3.2.2.5.9	Persistente Zwischenspeicherung von Konfigurationsdaten	24
3.2.2.6	Archiv	24
3.2.2.7	Zustandsinformationen eines Datenverteilers	25
3.3	Dynamisches Ablaufmodell	25
3.4	Kritikalität der SW-Komponenten/SW-Module/Prozesse/Datenbanken	25
3.5	Sonstige Entwurfsentscheidungen	25
<b>4</b>	<b>Schnittstellen</b>	<b>26</b>
4.1	Externe Schnittstellen der SW-Einheit	26
4.2	Interne Schnittstellen der SW-Einheit	26
<b>5</b>	<b>Anforderungszuordnung</b>	<b>27</b>

## Abkürzungen

siehe Dokument "Abkürzungen".

## Definitionen

siehe Dokument "Glossar".

## Verzeichnis der Tabellen

Tabelle 1-1: Versionsübersicht	2
Tabelle 1-2: Änderungsübersicht	3
Tabelle 3-1: Identifizierung der SW-Komponenten und Module	10
Tabelle 3-2: Kritikalität der SW-Komponenten/SW-Module/Prozesse/Datenbanken	25
Tabelle 4-1: Übersicht der externen Schnittstellen der SW-Einheit	26
Tabelle 4-2: Interne Schnittstellen der SW-Einheit	26
Tabelle 5-1: Zuordnung der Anforderungen an die SW-Module	28

## Verzeichnis der Abbildungen

Abbildung 3-1: Strukturierung der SW-Komponenten und Module	9
Abbildung 3-2: Schnittstellen der SW-Module	10
Abbildung 3-3: Verbindungsaufbau zwischen Applikation und Datenverteiler	14
Abbildung 3-4: Zustände der logischen Datenverteilerverbindung	17

## Referenzierte Dokumente

[Afo]	Anwenderforderungen AK VRZ, Dokument "SE-02.00.00.00.00-Afo", aktueller Stand
[SysArc]	Systemarchitektur AK VRZ, Dokument "SE-02.00.00.00.00-SysArc", aktueller Stand
[TanfGes]	Technische Anforderungen an das Gesamtsystem AK VRZ, Dokument "SE-02.00.00.00.00-TAnf", aktueller Stand
[TanfDaV]	Technische Anforderungen an den Datenverteiler AK VRZ, Dokument "SE-02.01.00.00.00-TAnf", aktueller Stand
[DatK]	Datenkatalog AK VRZ, Dokument "SE-02.00.00.00.00-DatK", aktueller Stand
[SSÜb]	Schnittstellenübersicht AK VRZ, Dokument "SE-02.00.00.00.00-SSÜb", aktueller Stand
[SSB]	Schnittstellenbeschreibung AK VRZ, Dokument "SE-02.00.00.00.00-SSB", aktueller Stand
[MARZ]	Merkblatt für die Ausstattung von Verkehrsrechnerzentralen und Unterzentralen, Ausgabe 1999.
[VMOD97]	Der Bundesminister des Innern, Entwicklungsstandard für IT-Systeme des Bundes Vorgehensmodell, Juni 1997, KBSt, Koordinations- und Beratungsstelle der Bundesregierung für Informationstechnik in der Bundesverwaltung.
[IEEE754]	IEEE Std 754-1985 Reaffirmed 1990, IEEE Standard for Binary Floating-Point Arithmetic < <a href="http://standards.ieee.org/reading/ieee/std_public/description/busarch/754-1985_desc.html">http://standards.ieee.org/reading/ieee/std_public/description/busarch/754-1985_desc.html</a> >
[HMAC]	Krawczyk, H., Bellare, M., and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication", RFC 2104, Februar 1997. < <a href="http://rfc.net/rfc2104.html">http://rfc.net/rfc2104.html</a> >
[MD5]	Rivest, R., "The MD5 Message-Digest Algorithm", RFC 1321, April 1992. < <a href="http://rfc.net/rfc1321.html">http://rfc.net/rfc1321.html</a> >

## 2 Lösungsvorschläge

Kriterien für die Zerlegung der SW-Einheit in Module waren insbesondere Austauschbarkeit und Wiederverwendbarkeit von Modulen sowie die Erweiterbarkeit der SW-Einheit durch weitere Module mit gleicher Schnittstelle. Weitere wesentliche Kriterien waren Abgeschlossenheit, schwache Kopplung der Module untereinander und enge Bindung innerhalb eines Moduls.

Wenn mehrere Module gemeinsam einen Funktionskreis implementieren, wurden sie zu Komponenten zusammengefasst.

Die sich aus dieser Vorgehensweise herauskristallisierte Modularisierung wird in den folgenden Kapiteln dargestellt.



### 3 Modularisierung/Datenbankentwurf

#### 3.1 Übersicht der SW-Komponenten, SW-Module, Prozesse und Datenbanken

Die SW-Einheit "Datenverteiler Applikationsfunktionen" setzt sich aus den Modulen

- Verwaltung
- Datenrepräsentation
- Protokollsteuerung DAF-DaV
- Telegrammverwaltung
- TCP-Kommunikationskanal
- HMAC-MD5 Verschlüsselung

zusammen.

Die folgende Abbildung zeigt die weitere Strukturierung der Module in Komponenten:

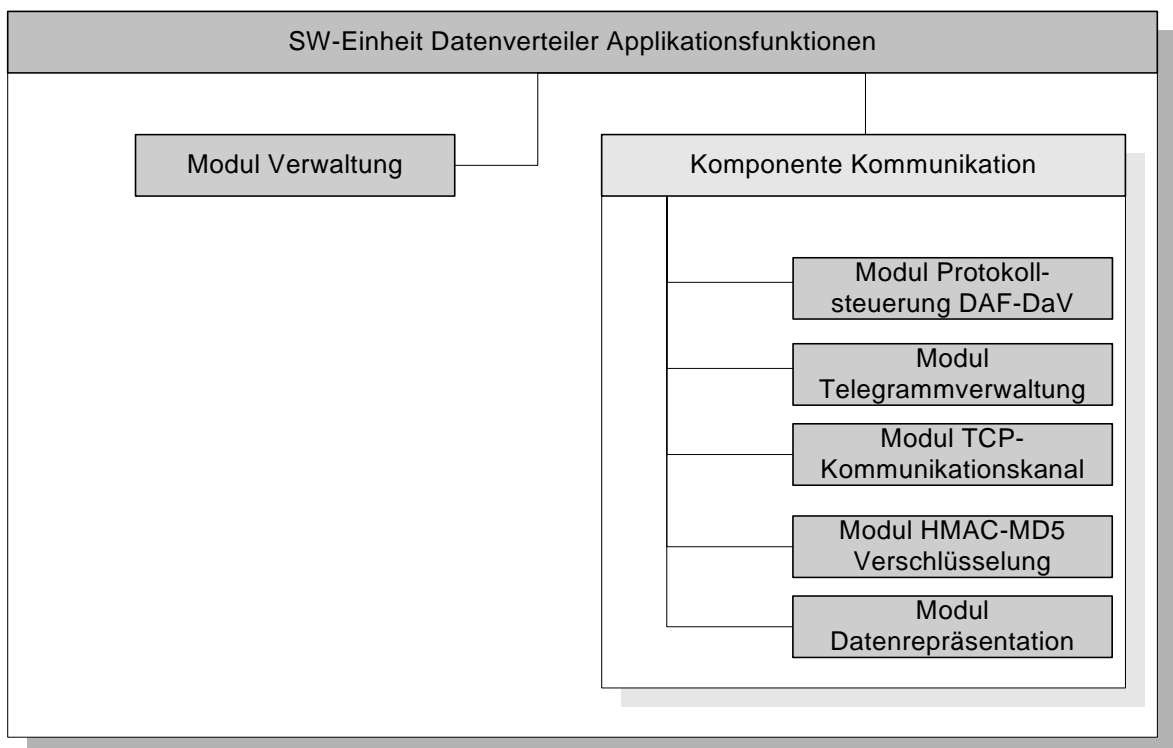


Abbildung 3-1: Strukturierung der SW-Komponenten und Module

Identifikator	Langbezeichnung
SE-01.01.01.00.01	Modul Verwaltung
SE-01.01.01.01.00	Komponente Kommunikation
SE-01.01.01.01.01	Modul Protokollsteuerung DAF-DaV
SE-01.01.01.01.02	Modul Telegrammverwaltung

Identifikator	Langbezeichnung
SE-01.01.01.01.03	Modul TCP-Kommunikationskanal
SE-01.01.01.01.04	Modul HMAC-MD5 Verschlüsselung
SE-01.01.01.01.05	Modul Datenrepräsentation

Tabelle 3-1: Identifizierung der SW-Komponenten und Module

### 3.2 Einzelbeschreibungen

Die folgende Abbildung zeigt die SW-Module und SW-Komponenten sowie die interne Kommunikationsstruktur der SW-Einheit:

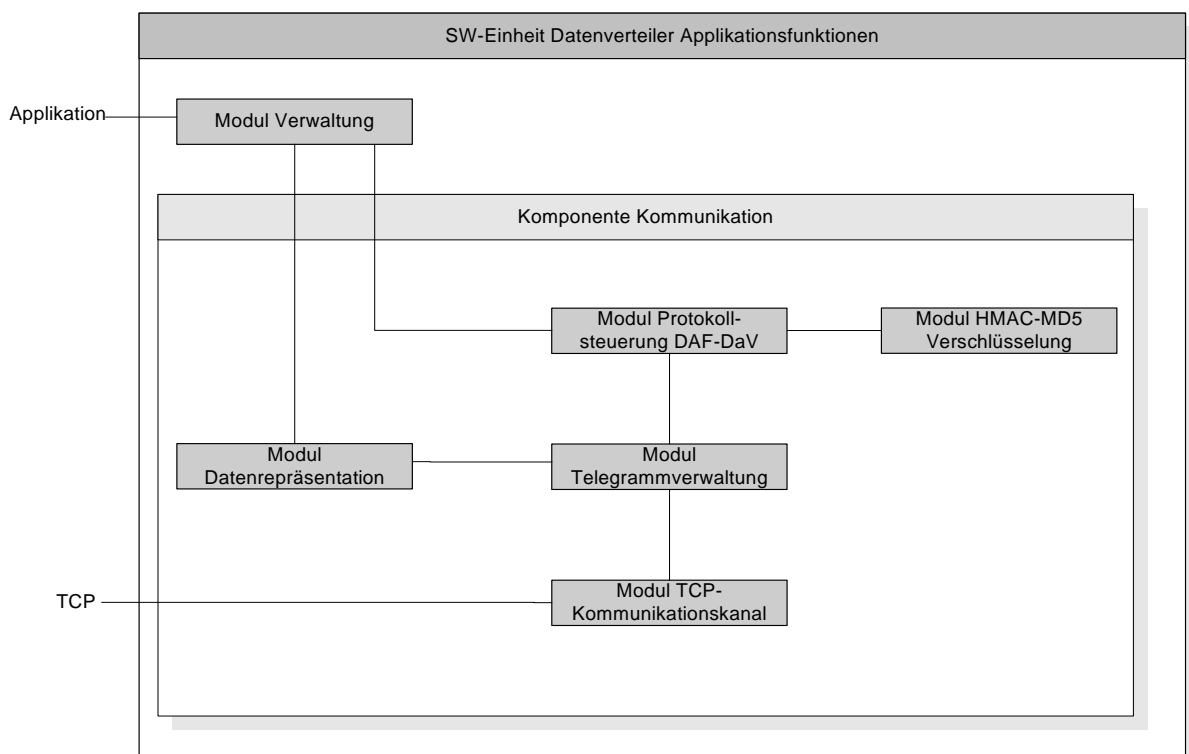


Abbildung 3-2: Schnittstellen der SW-Module

#### 3.2.1 Komponente Kommunikation

Die Komponente Kommunikation fasst die Module zusammen, die den Funktionsbereich der Kommunikation mit dem Datenverteiler abdecken. Die enthaltenen Module implementieren zusammen die Schnittstelle "DatenverteilerApplikationsfunktionen-Datenverteiler".

##### 3.2.1.1 Modul Datenrepräsentation

Das Modul Datenrepräsentation realisiert die eindeutige plattformneutrale Abbildung der in Telegrammen zu übertragenden Daten auf einen Bytestrom.

Es werden Funktionen zur Verfügung gestellt, die die verschiedenen Grunddatentypen auf einen Bytestrom konvertieren und umgekehrt.

Ganze Zahlen und Festkommazahlen werden in Telegrammen durch ein, zwei, vier oder acht Bytes im Big-Endian-Format dargestellt. Dabei werden die Zahlen so in Bytes zerlegt, dass das höherwertige Byte zuerst übertragen wird.

Fließkommazahlen werden durch 64 Bits dargestellt. Dabei wird das [IEEE754] Format verwendet. Die 64 Bits werden im Telegramm im Big-Endian-Format dargestellt.

Für die plattformneutrale Darstellung von Textzeichen wird das ISO 8859-1 Format verwendet.

### 3.2.1.2 Modul TCP-Kommunikationskanal

Das Modul TCP-Kommunikationskanal implementiert die Anbindung der Datenverteiler-Applikationsfunktionen zum Datenverteiler über TCP. Es abstrahiert die benötigten Funktionalitäten von TCP und stellt über eine allgemein verwendbare, einfache Schnittstelle einen verbindungsorientierten, gesicherten Kommunikationskanal zur Verfügung. Mit Hilfe der in TCP enthaltenen Quittierung und Wiederholung von Telegrammen wird die Erkennung und Korrektur von Übertragungsfehlern realisiert.

Das Modul enthält Funktionen zum Aufbau und zur Terminierung von TCP-Verbindungen und stellt für den Datenaustausch Streams zur Verfügung, über die Daten gelesen und geschrieben werden können. Weiterhin enthält es Funktionen zur Parametrierung der Ein- und Ausgabepuffer der zugrundeliegenden TCP-Implementierung.

Beim Aufbau von TCP-Verbindungen wird zwischen dem aktiven und dem passiven Verbindungsaufbau unterschieden. Der aktive Verbindungsaufbau wird in TCP-Clients benutzt, die den Aufbau einer TCP-Verbindung zu einem bestimmten Kommunikationspartner anstoßen. Der passive Verbindungsaufbau wird von TCP-Servern benutzt, die auf Verbindungswünsche von beliebigen TCP-Clients warten.

Der TCP-Kommunikationskanal wird als eigenes Modul realisiert, weil die Kommunikation durch andere Protokolle austauschbar sein muss. Außerdem kann dieses Modul bei der SW-Einheit Datenverteiler wiederverwendet werden.

### 3.2.1.3 Modul Telegrammverwaltung

Das Modul Telegrammverwaltung nutzt den TCP-Kommunikationskanal um die Kommunikation mit zwischen Applikationsfunktionen und Datenverteiler weiter zu abstrahieren und implementiert folgende richtungsunabhängige Funktionen der Kommunikation:

- Initialisierung. Die internen Datenstrukturen werden initialisiert und die TCP-Verbindung wird über das Modul TCP-Kommunikationskanal initialisiert.
- Interne Darstellung der verschiedenenartigen Telegramme mit den enthaltenen Informationen als Objekte und deren Serialisierung und De-Serialisierung. Die Telegramme, die in der Kommunikation zwischen Datenverteiler-Applikationsfunktionen und Datenverteiler eingesetzt werden sind in der Schnittstellenbeschreibung [SSB] zur Schnittstelle Datenverteiler-Applikationsfunktionen-Datenverteiler enthalten. Die für die Serialisierung und De-Serialisierung erforderliche Konvertierung der in den Systemtelegrammen enthalten Grunddatentypen wird mit Hilfe des Moduls Datenrepräsentation durchgeführt.
- Beim Empfang von Telegrammen aus dem TCP-Kommunikationskanal ist der formale Aufbau zu prüfen und im Fehlerfall eine entsprechende Nachricht an die Protokollsteuerung abzusetzen.
- Verwaltung von zwei priorisierten Telegrammpuffern in der zu sendende bzw. empfangene Telegramme bis zur Weiterverarbeitung und Übergabe an andere Module zwischengespeichert werden.

- Versand von Telegrammen im Sendepuffer über den TCP-Kommunikationskanal. Telegramme mit einer höheren Priorität werden bei der Weiterverarbeitung bevorzugt behandelt. Bei einem vollen Telegrammpuffer wird über eine Flusskontrolle die jeweilige Quelle der Telegramme solange blockiert, bis wieder Platz für neue Telegramme in dem jeweiligen Puffer freigeworden ist.
- Entgegennahme von Telegrammen vom TCP-Kommunikationskanal und Zwischenspeicherung im Empfangspuffer.
- Überwachung des Kommunikationskanals durch:
  - Senden von Keep-Alive-Telegrammen, wenn eine bestimmte Zeit kein Telegramm versendet wurde.
  - Timeout für den Empfang von Telegrammen. Wenn eine bestimmte Zeit kein Telegramm empfangen wurde, wird eine entsprechende Nachricht an die Protokollsteuerung abgesetzt.
  - Durchsatzprüfung, wenn der Sendepuffer eine parametrierbare Zeit einen parametrierbaren Füllstand überschreitet. Ist der ermittelte Durchsatz über eine parametrierbare Zeitspanne unterhalb einer parametrierbaren Grenze, wird die entsprechende Nachricht an die Protokollsteuerung abgesetzt..
- Abbruch der Kommunikation. Der TCP-Kommunikationskanal wird sofort geschlossen und noch im Puffer befindliche Telegramme werden gelöscht.
- Terminierung der Kommunikation. Der Aufruf der Terminierung führt dazu, dass Telegramme, die sich noch im Sendepuffer befinden an den Datenverteiler übertragen werden und anschließend der TCP-Kommunikationskanal geschlossen wird. In dieser Phase werden empfangene Telegramme ignoriert.

Die Telegrammverwaltung wird als eigenes Modul entworfen, weil sie bei der SW-Einheit Datenverteiler wiederverwendet werden kann und verschiedene zusammenhängende Funktionalitäten durch eine einfache Schnittstelle transparent zur Verfügung stellt.

#### 3.2.1.4 Modul Protokollsteuerung DAF-DaV

Das Modul Protokollsteuerung ist das Bindeglied der Komponente Kommunikation zum Modul Verwaltung.

Es stellt für die Interaktion mit dem Datenverteiler eine Funktionsschnittstelle zur Verfügung, die die technischen Aspekte der Kommunikation gegenüber der Verwaltung kapselt und implementiert die Abbildung auf Telegramme und Kommunikationsabläufe.

Die Protokollsteuerung grenzt sich zur Telegrammverwaltung ab, weil hier die aus Sicht der Datenverteiler-Applikationsfunktionen speziellen und nicht wiederverwendbaren Funktionen der Kommunikation mit dem Datenverteiler enthalten sind.

Folgende Funktionen und Abläufe werden in der Protokollsteuerung implementiert:

- Initialisierung und Verhandlung der Protokollversion.

Mit Hilfe der Telegrammverwaltung wird die TCP-Verbindung zum Datenverteiler hergestellt. Anschließend wird dem Datenverteiler in einem Protokollversionsanfrage-Telegramm mitgeteilt, welche Versionen des Kommunikationsprotokolls unterstützt werden. Als Reaktion wird vom Datenverteiler ein Protokollversionsantwort-Telegramm erwartet in dem die zu verwendende Version des Protokolls enthalten ist.
- Authentifizierung gegenüber dem Datenverteiler und Verhandlung der Verbindungsparameter.

Zur Authentifizierung wird ein Authentifizierungsbereitschafts-Telegramm übertragen. Als Antwort wird ein vom Datenverteiler generierter Zufallstext im Authentifizierungsaufforderungs-Telegramm erwartet. Dieser Text wird dem Modul "HMAC-MD5 Verschlüsselung" verschlüsselt. Als geheimer Schlüssel dient dabei das Passwort des Benutzers. Der verschlüsselte Text wird mit weiteren Informationen zum Datenverteiler übertragen. Als Antwort wird vom Datenverteiler ein Authentifizierungsannahme-Telegramm oder ein Authentifizierungsablehnungs-Telegramm erwartet. Im Falle einer Authentifizierungsablehnung wird der Authentifizierungsvorgang abgebrochen und kann bei Bedarf von der Verwaltung mit anderen Authentifizierungsinformationen wiederholt initiiert werden. Im Falle einer Authentifizierungsannahme werden anschließend die Verbindungsparameter verhandelt. Dazu sendet die Protokollsteuerung ein Verbindungsparameteranfrage-Telegramm an den Datenverteiler und erwartet ein Verbindungsparameterantwort-Telegramm mit der Festlegung der Verbindungsparameter vom Datenverteiler.

- Abbruch der Verbindung

Diese Funktion führt zum sofortigen Abbruch der Verbindung zum Datenverteiler mit Hilfe der Abbruch-Funktion der Telegrammverwaltung.

- Terminierung der Verbindung

Zur Terminierung der Verbindung zum Datenverteiler wird ein Applikationsabmeldungs-Telegramm übertragen und die Kommunikationsverbindung terminiert. Dabei wird durch die Telegrammverwaltung sichergestellt, dass alle noch im Sendepuffer befindliche Telegramme vor dem Schließen der Kommunikationsverbindung zum Datenverteiler übertragen werden.

- Parametrierung des Datenvertellers

Zur An- bzw. Abmeldung von zu empfangenden bzw. zu sendenden Daten wird ein entsprechendes Telegramm (Empfangsanmeldung, Sendeanmeldung, Empfangsabmeldung, Sendeanmeldung) an den Datenverteiler übertragen.

- Entgegennahme von Sendesteuerungs-Telegrammen und Weiterleitung der enthaltenen Information (ob Daten, die von der Applikation zum Senden angemeldet wurden, vom Datenverteiler benötigt werden oder nicht ) an die Verwaltung.

- Senden von Anwendungsdaten

Zu sendende Datenblöcke werden in Telegramme zerlegt und an die Telegrammverwaltung weitergeleitet.

- Empfang von Anwendungsdaten

Von der Telegrammverwaltung entgegengenommen Telegramme werden zu Datenblöcken rekombiniert und an das Modul Verwaltung weitergeleitet.

- Zerlegung von großen Datenblöcken in mehrere Datentelegramme beim Senden und Rekombination von zusammengehörigen Datentelegrammen zu einem Datenblock beim Empfang.

- Telegrammlaufzeitermittlung

Zur Messung von Telegrammlaufzeiten wird ein Telegrammlaufzeitanfrage-Telegramm mit der aktuellen Systemzeit an den Datenverteiler übertragen. Als Antwort wird ein Telegrammlaufzeitantwort-Telegramm erwartet, in dem die ursprünglich übertragene Systemzeit enthalten ist. Durch Vergleich der Zeit zwischen dem Absenden des Anfrage-Telegramms und dem Erhalt der Antwort wird die Laufzeit für Hin- und Rückweg zusammen ermittelt. An-

rerseits kann auch vom Datenverteiler eine solche Messung durchgeführt werden, d.h. bei Empfang eines Telegrammlaufzeitanfrage-Telegramms wird die enthaltene Systemzeit in einem Telegrammlaufzeitanfrage-Telegramm zurück übertragen.

- Fehlerbehandlung

Fehlerzustände, die von der Protokollsteuerung erkannt werden oder von der Telegrammverwaltung gemeldet werden, führen zu einem Abbruch der Verbindung und zu einer Benachrichtigung des Moduls "Verwaltung".

Folgendes Sequenzdiagramm zeigt die Interaktionen zwischen den Modulen beim Verbindungsaufbau:

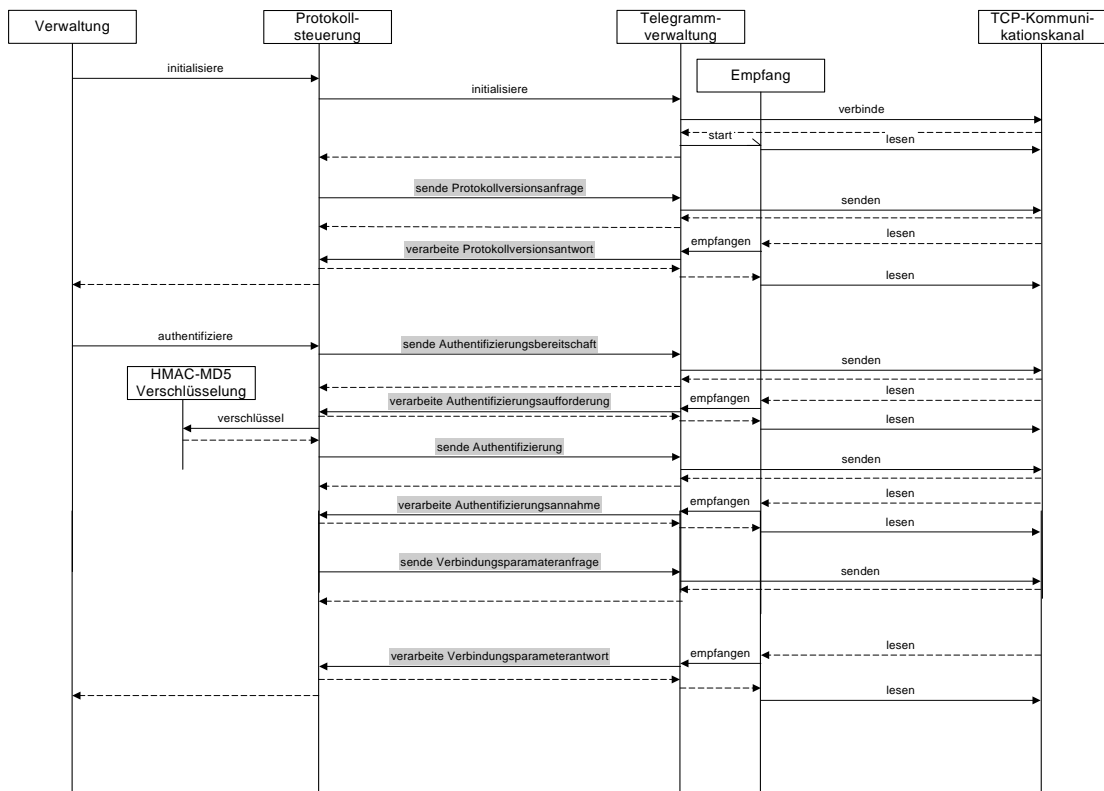


Abbildung 3-3: Verbindungsaufbau zwischen Applikation und Datenverteiler

### 3.2.1.5 Modul HMAC-MD5 Verschlüsselung

Das Modul "HMAC-MD5 Verschlüsselung" implementiert ein Verfahren zur Nachrichtenauthentifizierung mittels der kryptographischen Hashfunktion MD5. Es wird vom Modul Protokollsteuerung zur sicheren Authentifizierung gegenüber dem Datenverteiler eingesetzt. Das Verfahren bildet aus einer Nachricht und einem geheimen Schlüssel eine Signatur, die über ein unsicheres Medium übertragen werden kann und vom Empfänger der Nachricht auf Echtheit überprüft werden kann.

Das HMAC Verfahren ist im Detail in [HMAC] beschrieben. Es kann mit verschiedenen Hashfunktionen benutzt werden. Hier wird es mit dem kryptographischen Verfahren MD5 verwendet, dass in [MD5] beschrieben ist.

### 3.2.2 Modul Verwaltung

Das Modul Verwaltung implementiert die Schnittstellen "DatenverteilerApplikationsfunktionen-Starter" und "DatenverteilerApplikationsfunktionen-Applikation", d.h. sie stellt Funktionen zur Verwaltung der logische Datenverteilerverbindung und zum Datenaustausch über den Datenverteiler sowie Funktionen zum Zugriff auf die Konfiguration und das Archiv bereit. Diese können von beliebigen Applikationen benutzt werden.

#### 3.2.2.1 Plausibilitätsprüfung der Parameter der Applikationsschnittstelle

Bei allen Funktionen und Methoden der Schnittstelle "DatenverteilerApplikationsfunktionen-Applikation" sind Plausibilitätsprüfungen bei allen Parametern entsprechend der Schnittstellenbeschreibung [SSB] durchzuführen, sofern die möglichen Fehler nicht durch die eingesetzte Programmiersprache oder Laufzeitumgebung ausgeschlossen werden können.

Erkannte Fehler führen zu entsprechenden Ausnahmen, die von der Applikation behandelt werden können.

#### 3.2.2.2 Auswertung der Aufrufparameter

Das Modul Verwaltung implementiert die Starterschnittstelle der Datenverteiler-Applikationsfunktionen durch Auswertung der in der Schnittstellenbeschreibung [SSB] festgelegten Parameter und stellt außerdem Funktionen zur Verfügung, um die einzelnen Parameter zu setzen und abzufragen. Nicht spezifizierte Parameter werden auf Standard-Werte gesetzt.

#### 3.2.2.3 Verwaltung der logischen Verbindung zum Datenverteiler

Für jede Verbindung zu einem Datenverteiler verwaltet das Modul Verwaltung ein logisches Verbindungsobjekt, das sich in einem der folgenden Zustände befinden kann:

- `AußerBetrieb`

Es besteht keine Kommunikationsverbindung zum Datenverteiler.

- `Initialisiert`

Es besteht eine Kommunikationsverbindung zum Datenverteiler, aber es hat noch keine erfolgreiche Authentifizierung stattgefunden.

- `InBetrieb`

Es besteht eine Kommunikationsverbindung zum Datenverteiler und es hat eine erfolgreiche Authentifizierung stattgefunden.

Zur Verwaltung der logischen Verbindung stehen folgende Funktionen bereit, die von der Applikation aufgerufen werden können:

- Initialisierung der Verbindung

Diese Funktion kann nur im Zustand `AußerBetrieb` verwendet werden. Die internen Datenstrukturen der Verwaltung werden initialisiert und über das Modul Protokollsteuerung wird die Kommunikation mit dem Datenverteiler initialisiert. Wenn dies erfolgreich ist, erfolgt ein Übergang in den Zustand `Initialisiert`. Die zu verwendende Kommunikationsadresse des Datenverters kann beim Aufruf angegeben werden. Wenn keine Adresse angegeben wird, wird die Adresse, die über die Starterschnittstelle spezifiziert wurde, benutzt.

- Terminierung der Verbindung

Diese Funktion kann in den Zuständen `Initialisiert` und `InBetrieb` verwendet werden. Sie führt zu einer Terminierung der Kommunikation über das Modul `Protokollsteuerung`. Anschließend werden die internen Datenstrukturen auf den Initialzustand zurückgesetzt. Die logische Verbindung wird in den Zustand `AußerBetrieb` überführt.

- Abbruch der Verbindung

Diese Funktion kann in den Zuständen `Initialisiert` und `InBetrieb` verwendet werden. Sie führt zu einem sofortigen Abbruch der Kommunikation über das Modul `Protokollsteuerung`. Die internen Datenstrukturen werden auf den Initialzustand zurückgesetzt. Die logische Verbindung wird in den Zustand `AußerBetrieb` überführt.

- Authentifizierung

Diese Funktion kann im Zustand `Initialisiert` verwendet werden. Über das Modul `Protokollsteuerung` wird die Authentifizierung durchgeführt. Bei erfolgreicher Authentifizierung erfolgt ein Zustandswechsel in den Zustand `InBetrieb`. Erst nach erfolgreicher Authentifizierung kann über das logische Verbindungsobjekt auf die weiteren Funktionen des Moduls `Verwaltung` zugegriffen werden. Wenn die Authentifizierung nicht erfolgreich war, bleibt die Verbindung im Zustand `Initialisiert`. Die zu verwendenden Authentifizierungsinformationen (Benutzername und Passwort) können beim Aufruf angegeben werden. Wenn keine Authentifizierungsinformationen angegeben wurden, dann werden die entsprechenden Parameter, die über die Starterschnittstelle spezifiziert wurden, benutzt.

- Fehlerbehandlung

Beim Auftreten von Situationen, die zum Fehlverhalten der Datenverteiler-Applikationsfunktionen führen können, wird die Verbindung in den Zustand `AußerBetrieb` überführt, eine evtl. bestehende Verbindung mit dem Datenverteiler wird abgebrochen und die internen Datenstrukturen werden auf den Initialzustand zurückgesetzt. Die unerwartete Terminierung der Verbindung wird der Applikation über ein `Beobachter-Objekt` mitgeteilt.

Auslöser für die Fehlerbehandlung können sein:

- Fehler in der Kommunikation, die von der `Protokollsteuerung` gemeldet werden
- Fehler in der Kommunikation mit der Konfiguration (siehe Kapitel 3.2.2.5 "Konfiguration")

Das folgende Bild zeigt die möglichen Übergänge zwischen den Zuständen der logischen Verbindung:



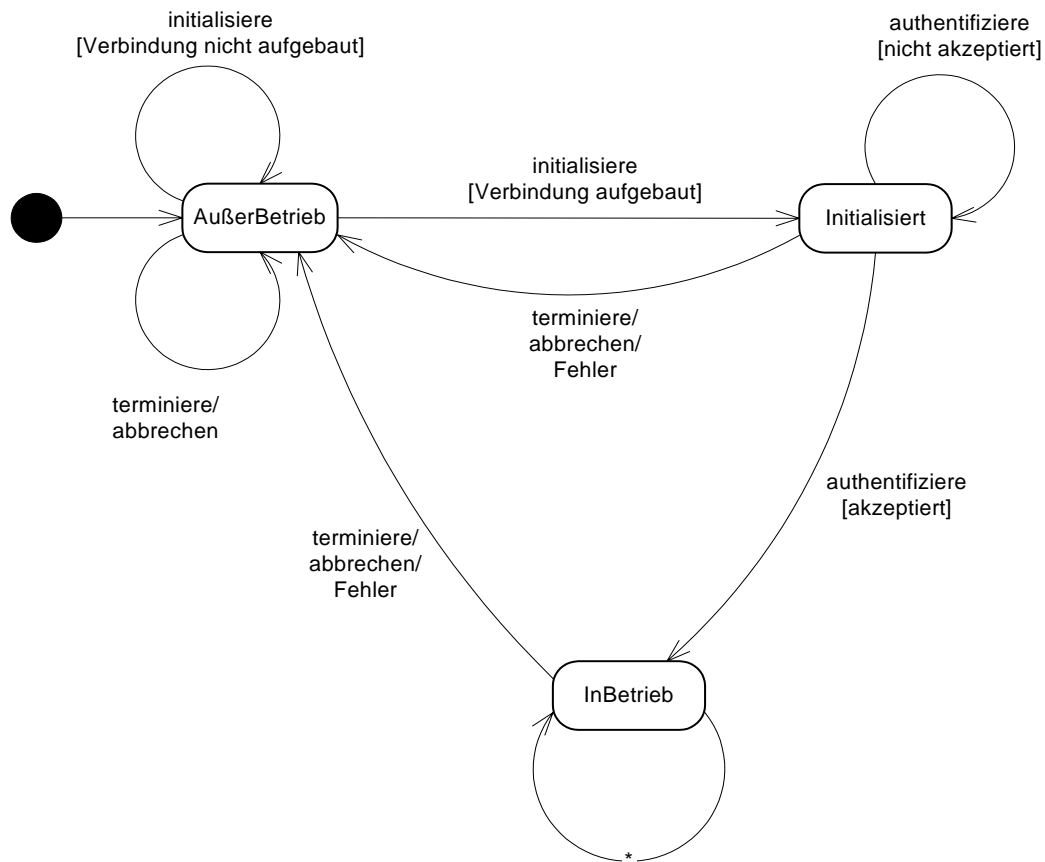


Abbildung 3-4: Zustände der logischen Datenverteilerverbindung

### 3.2.2.4 Funktionen der Datenverteilung

Bezüglich des Austauschs von Daten über den Datenverteiler realisiert das Modul Verwaltung folgende Funktionen:

#### 3.2.2.4.1 Verwaltung von Empfangs- und Sendeangabenungen

Die Applikation kann über entsprechende Methoden Daten zum Senden bzw. Empfangen an- und abmelden. Dabei muss sie die anzumeldenden Daten identifizieren. Dies geschieht durch Angabe folgender Informationen:

- Systemobjekt dem die Daten zugeordnet sind
- Attributgruppe der Daten
- Aspekt der Daten
- Simulationsvariante

Das Modul Verwaltung speichert die Anmeldeinformationen und gibt die Anmeldungen mit Hilfe der Komponente Kommunikation an den Datenverteiler weiter.

Bei der Anmeldung beim Datenverteiler müssen Aspektmanipulationen, die über die Starter-schnittstelle gesteuert werden können, berücksichtigt werden.

Sofern die Simulationsvariante nicht explizit bei der Anmeldung durch die Applikation vorgegeben wurde, wird die über die Starterschnittstelle spezifizierte Simulationsvariante benutzt.

Gespeichert werden je Anmeldung

- Systemobjekt dem die Daten zugeordnet sind
- Attributgruppe der Daten
- Aspekt der Daten (interne und externe Repräsentation; wenn der Aspekt, unter dem die Daten angemeldet sind, manipuliert wurde, werden die Daten innerhalb der Applikation weiter "normal" und bei Versand bzw. Empfang mit der "externen Repräsentation" behandelt.
- Simulationsvariante
- Beobachterobjekt der Applikation.<sup>1</sup>
- Ob die Daten zum Senden oder zum Empfangen angemeldet sind
  - Bei Sendeanmeldungen müssen zusätzlich folgende Informationen vorgehalten werden:
    - Anmeldung als Quelle oder einfacher Sender
    - Ob Sendesteuerungen gewünscht werden
  - Bei Empfangsanmeldungen müssen zusätzlich folgende Informationen vorgehalten werden:
    - Anmeldung als Senke oder einfacher Empfänger
    - ob nachgelieferte Daten gewünscht sind
    - ob ein von Quelle generierter Datensatz nur dann empfangen werden soll, wenn er sich gegenüber dem vorherigen Zustand verändert hat (Delta-Kennung)
    - optional welche Attribute der Attributgruppe benötigt werden
    - Zeitbereich für Anmeldung mit Historie
    - Vorhaltezeitraum für die Zwischenspeicherung von empfangenen Daten
    - ob es sich um eine implizite Anmeldung (siehe Kapitel 3.2.2.4.6 "Lesen von Daten") handelt, wann die Daten zuletzt gelesen wurden und nach welcher Zeit eine implizite Anmeldung wieder abgemeldet werden kann

Bei der Anmeldung von Daten wird pro Datenidentifikation nur eine Anmeldung an den Datenverteiler abgesetzt. Wenn bei einer Empfangsanmeldung die Daten bereits beim Datenverteiler angemeldet sind und der aktuelle Zustand bereits vorliegt, wird die Applikation durch Aufruf der Aktualisierungsfunktion des angegebenen Beobachterobjekts benachrichtigt. Dabei wird der aktuelle Zustand übergeben.

Bei einer Anmeldung mit Historie wird zusätzlich ein Archivfrage (s.u.) für den gewünschten Zeitbereich angestoßen, wenn die Daten des gewünschten Zeitbereichs nicht bereits im Zwischenspeicher vorliegen.

<sup>1</sup> Bei Empfangsanmeldungen dient das Beobachterobjekt dazu, Aktualisierungen beim Empfang von Daten an die Applikation weiterzuleiten. Bei Sendeanmeldungen dient das Beobachterobjekt dazu, der Applikation mitzuteilen, dass die angemeldeten Daten vom Datenverteiler benötigt werden bzw. nicht mehr benötigt werden. Bei Sendeanmeldungen als Quelle ist die Angabe des Beobachterobjekts optional. Wenn kein Beobachterobjekt angegeben wird, sendet der Datenverteiler keine Sendesteuerung und die Daten können von der Quelle ohne Aufforderung gesendet werden.

Bei einer Abmeldung werden die zugehörigen Anmeldeinformationen aus der internen Anmeldeverwaltung entfernt und wenn keine weitere Anmeldung mit der gleichen Identifikation besteht, wird mit Hilfe der Kommunikation eine Abmeldung der beim Datenverteiler angemeldeten Daten abgesetzt.

Bei impliziten Anmeldungen wird eine automatische Abmeldung der Daten durchgeführt, wenn über eine Zeitdauer, die beim Lesen der Daten angegeben werden muss, kein weiterer Lesevorgang stattgefunden hat.

#### 3.2.2.4.2 Senden von Anwendungsdaten

Zu sendende Datensätze werden mit Hilfe der Komponente Kommunikation gesendet, wenn sie schon angemeldet sind. Wenn keine Sendeanmeldung vorliegt, wird eine implizite Anmeldung als einfacher Sender durchgeführt und auf eine Sendesteuerung vom Datenverteiler gewartet. Wenn der Datenverteiler in der Sendesteuerung zum Senden auffordert, wird der Datensatz gesendet und anschließend wieder abgemeldet. Wenn der Datenverteiler in der Sendesteuerung zum nicht Senden auffordert, werden die Daten wieder abgemeldet und eine Ausnahme generiert.

Wenn die Daten als Quelle angemeldet wurden, werden die Anwendungsdaten vor dem Senden mit einem monoton aufsteigenden Datensatzindex versehen. Dieser Index wird je Datenidentifikation vom Modul verwaltet. Anhand dieses Indexes kann später die Reihenfolge der Datensätze festgestellt werden. Der Index besteht aus 64 Bits, die höchstwertigen 32 Bits enthalten den Zeitpunkt der Initialisierung der logischen Verbindung in Sekunden seit 1970, die folgenden 30 Bits werden als laufende Nummer bei jedem Sendevorgang um eins inkrementiert. Beim ersten Sendevorgang enthält die laufende Nummer den Wert 1. Die niedrigstwertigen 2 Bits werden immer mit 0 besetzt.

Eine weitere Aufgabe beim Versenden von Daten, die als Quelle angemeldet sind, besteht darin, Änderungsinformationen zum vorherigen Datensatz zu ermitteln. Die einzelnen Attribute und Attributlisten auf der höchsten Ebene der Attributgruppendefinition werden dabei auf Änderung untersucht und entsprechend markiert. Anschließend wird der Datensatz mit den ermittelten Änderungsmarkierungen an die Komponente Kommunikation zum Versand übergeben<sup>2</sup>.

#### 3.2.2.4.3 Sendesteuerungen an Applikation weiterleiten

Beim Empfang eines Sendesteuerungstelegramms wird der Inhalt von der Verwaltung ausgewertet und das Beobachterobjekt der entsprechenden Sendeanmeldung wird benachrichtigt, ob Daten gesendet werden können oder nicht.

#### 3.2.2.4.4 Entgegennahme von angemeldeten Daten

Von der Komponente Kommunikation empfangene Datensätze werden vom Modul Verwaltung entgegengenommen und im Zwischenspeicher abgelegt (siehe Kapitel 3.2.2.4.5 "Zwischenspeicherung von Daten"). Außerdem werden Daten, die infolge einer Empfangsanmeldung mit Historie aus dem Archiv abgefragt wurden, entgegengenommen und im Zwischenspeicher abgelegt.

Bei einer Anmeldung ohne Historie wird das Beobachterobjekt der entsprechenden Anmeldung über den Aufruf der Aktualisierungsmethode benachrichtigt.

---

<sup>2</sup> Die Änderungsmarkierungen werden vom Datenverteiler ausgewertet und an Applikationen, die sich zum Empfang mit Delta-Kennung angemeldet haben, nur dann weitergeleitet, wenn die angemeldeten Attribute sich verändert haben.

Bei einer Anmeldung mit Historie wird vor der Benachrichtigung des Beobachterobjekts geprüft, ob alle Daten des bei der Empfangsanmeldung angegebenen Zeitraums und die aktuellen Daten vom Datenverteiler lückenlos vorliegen. Wenn beide Bedingungen erfüllt sind, wird das Beobachterobjekt der entsprechenden Anmeldung benachrichtigt. Dabei wird der aktuelle Datensatz übergeben. Lücken werden anhand des in jedem Datensatz enthaltenen Datensatzindex erkannt.<sup>3</sup>

#### 3.2.2.4.5 Zwischenspeicherung von Daten

Das Modul verwaltet einen Zwischenspeicher, in dem empfangene Datensätze vorgehalten werden.

Duplikate von Datensätzen, die dadurch entstehen können, dass Datensätze über verschiedene Wege zur Applikation gelangen, werden vom Zwischenspeicher anhand des in jedem Datensatz enthaltenen Datensatzindex erkannt und eliminiert.

Über die Angabe eines Vorhaltezeitraums kann die Applikation bei der Anmeldung der Daten spezifizieren, ob und wie lange zwischengespeicherte Daten neben dem aktuellen Datensatz gespeichert werden müssen. Historische Daten, die nicht mehr im Vorhaltezeitraum liegen, werden gelöscht. Die entsprechende Prüfung, ob die historischen Daten einer Anmeldung zu löschen sind, findet nach jeder Aktualisierungsnachricht an das entsprechende Beobachterobjekt statt.

#### 3.2.2.4.6 Lesen von Daten

Die Verwaltung stellt eine Funktion bereit, mit der auf den aktuellen Zustand von Daten zugegriffen werden kann.

Wenn die Daten noch nicht angemeldet sind, wird eine implizite Anmeldung (als Empfänger) durchgeführt<sup>4</sup>. Wenn der aktuelle Stand der gewünschten Daten noch nicht in den Datenverteiler-Applikationsfunktionen vorliegt wird darauf gewartet. Nach dem Empfang der aktuellen Daten werden diese zurückgegeben.

Über eine weitere Funktion ist der Zugriff auf die im Zwischenspeicher abgelegten historischen Daten möglich.

#### 3.2.2.4.7 Zugriff auf Datensätze

Datensätze werden durch Objekte implementiert, über die ein direkter Zugriff auf folgende Informationen möglich ist:

- System-Objekt
- Attributgruppe
- Aspekt
- Simulationsvariante

<sup>3</sup> Duplikate von Datensätzen, die dadurch entstehen können, dass hier Datensätze über verschiedene Wege zur Empfangsapplikation gelangen, werden vom Zwischenspeicher (siehe Kapitel 3.2.2.4.5 "Zwischenspeicherung von Daten") erkannt und eliminiert.

<sup>4</sup> Implizite Anmeldungen beim Lesen von Daten führen dazu, dass bei weiteren Lesevorgängen keine Wartezeiten entstehen. Wenn keine weiteren Lesevorgänge innerhalb einer angegebenen Zeit durchgeführt werden, werden die impliziten Anmeldungen wieder zurückgenommen (siehe Kapitel 3.2.2.4.1 "Verwaltung von Empfangs- und Sendeanmeldungen").

- Laufende Nummer
- Fehlerkennung  
Mit der Fehlerkennung wird gekennzeichnet, ob der Datensatz übertragen wurde. Folgende Kennungen sind möglich: Datensatz ist vorhanden; Datenquelle ist vorhanden aber es ist kein Datensatz verfügbar; Datenquelle nicht vorhanden; keine Berechtigung;
- Zeitstempel
- Kennzeichnung, ob es sich um einen nachgelieferten Datensatz handelt
- etc.

Über die oben erhaltenen Informationen ist es mit Hilfe der Konfiguration möglich, generisch auf weitere Eigenschaften des vorliegenden Datensatzes zuzugreifen.

Über das System-Objekt kann auf sämtliche Eigenschaften des Objekts, wie z.B. der Name, zugegriffen werden.

Es besteht die Möglichkeit über die einzelnen Attribute des Datensatzes zu iterieren und jeweils die Eigenschaften (z.B. den Namen) und den Wert des Attributs zu erfragen. Wenn der Name des gewünschten Attributs bekannt ist, kann direkt über den Namen auf das Attribut und den Wert des Attributs zugegriffen werden.

Zu jedem Attribut können über den zugeordneten Attributtyp der konfigurierte Wertebereich, die Einheit eine mögliche Skalierung und weitere Informationen abgefragt werden.

Wenn ein Datensatz geschrieben werden soll, kann über eine weitere Funktion ein neues Datensatzobjekt erzeugt werden, das bereits die Struktur der gewünschten Attributgruppe aufweist. Die Zuweisung eines zu sendenden Attributwertes erfolgt beispielsweise über die Angabe des Attributnamens.

### 3.2.2.5 Konfiguration

Über die logische Verbindung mit dem Datenverteiler bietet das Modul Verwaltung einen Zugriff auf die zugeordnete Konfiguration. Hierüber kann auf die Metainformationen des Datenmodells und auf alle anderen Konfigurationsdaten zugegriffen werden.

Die Funktion implementiert eine gezielte Interaktion mit der Konfiguration über folgende Attributgruppen:

- KonfigurationsAnfrage
- KonfigurationsAntwort

Die Verwaltung meldet sich selbst intern als Sender der Attributgruppe "KonfigurationsAnfrage" für das Konfigurationsobjekt der zugeordneten Datenhaltung<sup>5</sup> an. Außerdem meldet sich die Verwaltung selbst intern als Empfänger der Attributgruppe "KonfigurationsAntwort" für das dynamische Konfigurationsobjekt der Applikation an.

Die Anmeldungen auf die beiden Attributgruppen werden einmal beim Übergang der logischen Verbindung in den Zustand `InBetrieb` durchgeführt und die Sendesteuerung vom Datenverteiler für die KonfigurationsAnfrage wird abgewartet.

<sup>5</sup> Die Objekt-ID dieses Objekts wird nach erfolgreicher Authentifizierung beim Aufbau der Kommunikationsverbindung vom Datenverteiler zur Verfügung gestellt (siehe [SSB] Systemtelegramm Authentifizierungsannahme)

Wenn der Datenverteiler in der initialen oder einer späteren Sendesteuerung zum nicht Senden auffordert oder bei der Interaktion mit der Konfiguration unerwartete Antworten bei bestimmten Anfragen empfangen werden, dann wird ein Fehler der logischen Verbindung generiert<sup>6</sup>.

#### 3.2.2.5.1 Stellvertreterobjekte

Die von der Konfiguration verwalteten System-Objekte werden seitens der Applikation durch sogenannte Stellvertreterobjekte dargestellt. Die Stellvertreterobjekte werden an der Applikationsschnittstelle der Datenverteiler-Applikationsfunktionen an allen Stellen benutzt, an denen ein System-Objekt spezifiziert werden muss. Seitens der Kommunikation mit dem Datenverteiler werden Systemobjekte im allgemeinen durch eine 64-Bit Objekt-ID identifiziert.

Über die Stellvertreterobjekte implementiert das Modul Verwaltung Funktionen zum Zugriff auf die Eigenschaften von Objekten und deren Zusammenhänge. Allgemeine Eigenschaften von System-Objekten werden durch entsprechende Methoden der jeweiligen Stellvertreter-Klassen implementiert. Der Zugriff auf die speziellen Eigenschaften der verschiedenartigen System-Objekte wird über ein generisches Interface realisiert.

#### 3.2.2.5.2 Metamodell

Das Metamodell ist die Beschreibung des dem System zugrundeliegenden Datenmodells. Diese Beschreibung wird in der Konfiguration selbst, also im Datenmodell gespeichert. Die verschiedenen Objekt-Typen des Metamodells werden von dem Modul Verwaltung als Klassen von Stellvertreterobjekten realisiert, die Methoden zum Zugriff auf die speziellen Eigenschaften der Objekte des Metamodells implementieren.

#### 3.2.2.5.3 Abfrage des Metamodells

Die Beschreibung des Datenmodells ist Grundlage für die Funktion der Datenverteiler-Applikationsfunktionen. Sie wird einmal beim Übergang der logischen Verbindung in den Zustand `InBetrieb` abgefragt und persistent gespeichert. Bei der Abfrage wird über einen Zeitstempel der Stand des bisher gespeicherten Metamodells angegeben, damit jeweils nur die Änderungen gegenüber dem lokal gespeicherten Stand übertragen werden müssen. Es werden die für die Datenverteiler-Applikationsfunktionen relevanten Konfigurationsobjekte des Metamodells mit ihrer ID und den vom Typ abhängigen Eigenschaften übertragen und in entsprechenden Stellvertreterobjekten gespeichert.

#### 3.2.2.5.4 Abfrage von Konfigurationsobjekten

Die Verwaltung implementiert Methoden, um auf beliebige System-Objekte zuzugreifen. Dabei können Objekte durch die Angabe ihrer ID, ihrer PID, ihrer Zugehörigkeit zu einem bestimmten Typ oder einer Menge spezifiziert werden. Entsprechende Anfragen werden an die Konfiguration übertragen und als Ergebnis werden die jeweiligen Objekte mit ihrer ID und den grundlegenden Eigenschaften erwartet. Aus der Antwort werden entsprechende Stellvertreterobjekte erzeugt und über die Applikationsschnittstelle zur Verfügung gestellt.

#### 3.2.2.5.5 Lesen von konfigurierenden Daten

Es werden Funktionen zur Verfügung gestellt, mit denen beliebige konfigurierende Daten gelesen werden können. Dazu wird in einer KonfigurationsAnfrage das Konfigurationsobjekt und die

---

<sup>6</sup> Folge eines Fehlers ist ein Abbruch der Verbindung mit dem Datenverteiler und eine entsprechende Benachrichtigung der Applikation wie in Kapitel 3.2.2.3 "Verwaltung der logischen Verbindung zum Datenverteiler" beschrieben.

Attributgruppe des gewünschten Datensatzes spezifiziert. Als Ergebnis wird der gewünschte Datensatz erwartet. Dieser wird zur Beantwortung weiterer Anfragen am Stellvertreter-Objekt zwischengespeichert und über die Applikationsschnittstelle zur Verfügung gestellt.

#### 3.2.2.5.6 Schreiben von konfigurierenden Daten

Zum Schreiben von Konfigurationsdaten werden Funktionen zu Verfügung gestellt, mit denen ein Datensatz einer konfigurierenden Attributgruppe erzeugt und an die Konfiguration übertragen werden kann. Als Ergebnis wird von der Konfiguration eine Quittung erwartet, aus der erkennbar ist, ob die Änderung durchgeführt wurde oder nicht. Im Fehlerfall wird eine entsprechende Ausnahme generiert.

#### 3.2.2.5.7 Erzeugen und Löschen von Objekten

Objekte von dynamischen Typen können zur Laufzeit erzeugt und gelöscht werden. Objekte von konfigurierenden Typen können auch zur Laufzeit erzeugt und gelöscht werden, allerdings wirkt sich die Änderung hier nicht sofort aus sondern erst mit Aktivierung der nächsten Konfigurationsversion. Die Verwaltung stellt entsprechende Funktionen zum Erzeugen und Löschen von dynamischen und konfigurierenden Objekten bereit. Außerdem implementiert sie Funktionen, mit denen sie die Erzeugung und das Löschen von dynamischen Objekten überwachen, in den betroffenen Stellvertreterobjekten entsprechend berücksichtigen und die Applikation bei Bedarf informieren kann.

Beim Erzeugen von Objekten wird die Objekt-ID von dem Modul Verwaltung vorgegeben. Damit sichergestellt ist, dass die Objekt-ID systemweit eindeutig vergeben wird, muss vor dem Erzeugen eine ID-Abfrage durchgeführt werden. Als Antwort wird von der Konfiguration ein Bereich von freien Objekt-IDs erwartet, die bei den folgenden Objekt-Erzeugungen verwendet werden können.

Zum Erzeugen des Objektes wird dann eine entsprechende Anfrage an die Konfiguration übertragen. Enthalten sind die ID des neuen Objekts und die grundlegenden Eigenschaften. Als Ergebnis wird von der Konfiguration eine Quittung erwartet, aus der erkennbar ist ob, das Objekt erzeugt wurde oder nicht. Im Fehlerfall wird eine entsprechende Ausnahme generiert.

Zum Löschen von Objekten wird eine entsprechende Anfrage mit der ID des zu löschenden Objekts an die Konfiguration übertragen. Als Ergebnis wird von der Konfiguration eine Quittung erwartet, aus der erkennbar ist, ob das Objekt gelöscht wurde oder nicht. Im Fehlerfall wird eine entsprechende Ausnahme generiert.

Das Erzeugen und Löschen von dynamischen Objekten wird von der Konfiguration in der Attributgruppe "DynamischeObjekte" über den Datenverteiler publiziert. Die Verwaltung meldet sich als Empfänger dieser Attributgruppe beim Datenverteiler an und erhält bei jeder Änderung (Objekt erzeugt, Objekt gelöscht) einen entsprechenden Datensatz, der dann zur Aktualisierung der betroffenen Stellvertreterobjekte führt.

#### 3.2.2.5.8 Änderung von Mengen

Dynamischen Mengen können zur Laufzeit verändert werden. Konfigurierende Mengen können auch zur Laufzeit verändert werden, allerdings wirkt sich die Änderung hier nicht sofort aus sondern erst mit Aktivierung der nächsten Konfigurationsversion. Die Verwaltung stellt entsprechende Funktionen zum Hinzufügen und Entfernen von Objekten einer Menge bereit. Außerdem implementiert sie Funktionen, mit denen sie die Veränderung von dynamischen Objekten überwachen, in den betroffenen Stellvertreterobjekten entsprechend berücksichtigen und die Applikation bei Bedarf informieren kann.

Zum Hinzufügen oder Entfernen eines Objekts einer Menge wird eine entsprechende Anfrage an die Konfiguration übertragen. Als Ergebnis wird von der Konfiguration eine Quittung erwartet,

aus der erkennbar ist, ob die Operation erfolgreich war oder nicht. Im Fehlerfall wird eine entsprechende Ausnahme generiert.

Änderungen von dynamischen Mengen werden von der Konfiguration in der Attributgruppe "DynamischeElemente" über den Datenverteiler publiziert. Die Verwaltung meldet sich als Empfänger dieser Attributgruppe beim Datenverteiler an und erhält bei jeder Änderung (Objekt hinzugefügt, Objekt entfernt) einen entsprechenden Datensatz, der dann zur Aktualisierung der betroffenen Stellvertreterobjekte führt.

### 3.2.2.5.9 Persistente Zwischenspeicherung von Konfigurationsdaten

Die in der Verwaltung intern gespeicherten Konfigurationsdaten können bei Bedarf beim Beenden der Applikation persistent gespeichert und beim Neustart wieder eingelesen werden.

Die Daten werden in dem von der Applikation spezifizierten Verzeichnis in einer Datei abgelegt. Als Dateiname wird die PID der Datenhaltung, die der Konfiguration zugeordnet ist, benutzt.

Beim Übergang der logischen Verbindung in den Zustand `InBetrieb` werden die Konfigurationsversionen für jeden Konfigurationsverantwortlichen der Konfiguration abgefragt und mit den in der für die entsprechende Konfiguration gespeicherten Datei enthaltenen Versionen verglichen. Es werden nur die Konfigurationsdaten aus der gespeicherten Datei übernommen, für die sich keine Abweichung der Version ergeben hat.

Die Konfiguration wird nur bei einer ordnungsgemäßen Terminierung der logischen Datenverteilerverbindung gespeichert. Wenn gezielt oder durch einen Kommunikationsfehler abgebrochen wurde, wird sie nicht gespeichert.

### 3.2.2.6 Archiv

Über die logische Verbindung mit dem Datenverteiler bietet das Modul Verwaltung einen Zugriff auf die zugeordnete Datenhaltung. Hierüber können archivierte Daten gelesen werden.

Die Funktion implementiert eine gezielte Interaktion mit der Datenhaltung über folgende Attributgruppen:

- ArchivAnfrage
- ArchivAntwort

Die Verwaltung meldet sich selbst intern als Sender der Attributgruppe "ArchivAnfrage" für das Konfigurationsobjekt der zugeordneten Datenhaltung an. Außerdem meldet sich die Verwaltung selbst intern als Empfänger der Attributgruppe "ArchivAntwort" für das dynamische Konfigurationsobjekt der Applikation an.

Die Anmeldungen auf die beiden Attributgruppen werden einmal beim Übergang der logischen Verbindung in den Zustand `InBetrieb` durchgeführt und die Sendesteuerung vom Datenverteiler für die ArchivAnfrage wird abgewartet.

Beim Aufruf der Archivabfragefunktion wird überprüft, ob Absetzungen von Archivfragen möglich sind. Wenn dies nicht der Fall ist, wird eine entsprechende Ausnahme generiert. Ansonsten werden die Parameter der Abfrage über die Attributgruppe "ArchivAnfrage" an die zugeordnete Datenhaltung übertragen.

Über die Attributgruppe ArchivAntwort werden anschließend die angeforderten Archivdaten empfangen und an den Interessenten weitergeleitet.



### 3.2.2.7 Zustandsinformationen eines Datenverteilers

Zur Abfrage der Zustandsinformationen eines beliebigen Datenverteilers im Netz stehen folgende Funktionen bereit, die über das dem jeweiligen Datenverteiler zugeordnete Konfigurationsobjekt aufgerufen werden können:

- Abfrage der aktuell verbundenen Applikationen. Hierzu wird eine Konfigurationsabfrage der dynamischen Menge mit dem Namen "Applikationen", ausgehend vom jeweiligen Konfigurationsobjekt des Datenverteilers, durchgeführt und eine Liste der enthaltenen Applikationen zurückgegeben.
- Abfrage der Telegrammlaufzeiten vom Datenverteiler zu den verbundenen Applikationen und zurück. Dazu wird eine Abfrage der Attributgruppe "TelegrammLaufzeiten" am Konfigurationsobjekt des Datenverteilers durchgeführt und das Ergebnis in einem Feld zurückgegeben.

### 3.3 Dynamisches Ablaufmodell

Die SW-Einheit Datenverteiler-Applikationsfunktionen ist eine Funktionsbibliothek die in Applikationen eingebunden wird und zusammen mit der SW-Einheit der jeweiligen Applikation einen Prozess darstellt. Sie kommuniziert über eine TCP-Verbindung mit dem Datenverteiler, zu dem eine starke dynamische Bindung besteht. Wenn der Datenverteiler nicht erreichbar ist, kann die Kommunikationsverbindung nicht aufgebaut werden. Wenn der Datenverteiler bei bestehender Verbindung nicht mehr erreichbar ist, wird dies von den Datenverteiler-Applikationsfunktionen erkannt und die Kommunikationsverbindung abgebrochen. Wenn die Kommunikationsverbindung nicht zustande kommt oder abgebrochen wird, können die Datenverteiler-Applikationsfunktionen ihre Aufgabe nicht erfüllen. Eine entsprechende Nachricht wird an die Applikation übergeben, die daraufhin entscheiden kann ob sie sich terminiert oder erneut initialisiert.

### 3.4 Kritikalität der SW-Komponenten/SW-Module/Prozesse/Datenbanken

Die Kritikalität der einzelnen SW-Komponenten und Module ist in der folgenden Tabelle aufgeführt:

Identifikator	Langbezeichnung	Kritikalität
SE-01.01.01.00.01	Modul Verwaltung	mittel
SE-01.01.01.01.00	Komponente Kommunikation	mittel
SE-01.01.01.01.01	Modul Protokollsteuerung Datenverteiler-Applikationsfunktionen-DaV	mittel
SE-01.01.01.01.02	Modul Telegrammverwaltung	mittel
SE-01.01.01.01.03	Modul TCP-Kommunikationskanal	mittel
SE-01.01.01.01.04	Modul HMAC-MD5 Verschlüsselung	mittel
SE-01.01.01.01.05	Modul Datenrepräsentation	mittel

Tabelle 3-2: Kritikalität der SW-Komponenten/SW-Module/Prozesse/Datenbanken

### 3.5 Sonstige Entwurfsentscheidungen

Es wurden keine sonstigen Entwurfsentscheidungen getroffen.

## 4 Schnittstellen

### 4.1 Externe Schnittstellen der SW-Einheit

Die folgende Tabelle gibt eine Übersicht der externen Schnittstellen der SW-Einheit. Dabei sind pro Schnittstelle die beiden Kommunikationspartner und eine Kurzbeschreibung zum Zweck der Schnittstelle angegeben.

Die detaillierte Beschreibung der Schnittstellen erfolgt in [SSB].

Schnittstellename	Zugeordnetes Strukturelement	Kurzbeschreibung
DatenverteilerApplikationsfunktionen- Starter	SW-Modul Verwaltung	Aufrufschnittstelle Der Starter ist die aufrufende SW-Einheit (z.B. Startskript) einer Applikation. Die Applikation nimmt die Aufrufparameter vom Starter entgegen und reicht sie ohne eigene Interpretation an die Datenverteilerapplikationsfunktionen weiter.
DatenverteilerApplikationsfunktionen- Datenverteiler	SW-Komponente Kommunikation	Telegrammschnittstelle zwischen den Datenverteilerapplikationsfunktionen und dem Datenverteiler.
DatenverteilerApplikationsfunktionen- Applikation	SW-Modul Verwaltung	Funktionsschnittstelle zum Datenverteiler Die Schnittstelle wird von Applikationen benutzt, um die Funktionalität des Datenverters über die Bibliothek "DatenverteilerApplikationsfunktionen" zu nutzen.

Tabelle 4-1: Übersicht der externen Schnittstellen der SW-Einheit

### 4.2 Interne Schnittstellen der SW-Einheit

Die folgende Tabelle zeigt die internen Schnittstellen der SW-Einheit. Dabei wird jede Schnittstelle namentlich aufgeführt und es wird durch Kreuze im rechten Teil der Tabelle gekennzeichnet, zwischen welchen Modulen die Schnittstelle angesiedelt ist. Die Beschreibung der Schnittstellen erfolgt in [SSB].

Schnittstellename	Komponente Modul	Kommunikation					
		Verwaltung	Protokollsteuerung	Datenrepräsentation	HMAC-MD5 Verschlüsselung	Telegrammverwaltung	TCP-Kommunikationskanal
Verwaltung-Kommunikation.Protokollsteuerung		X	X	---	---	---	---
Kommunikation.Datenrepräsentation-Modul		X	---	X	---	X	---
Kommunikation.Protokollsteuerung-Kommunikation.Telegrammverwaltung		---	X	---	---	X	---
Kommunikation.Protokollsteuerung-Kommunikation.HMACMD5Verschlüsselung		---	X	---	X	---	---
Kommunikation.Telegrammverwaltung-TCPKommunikationskanal		---	---	---	---	X	X

Tabelle 4-2: Interne Schnittstellen der SW-Einheit

## 5 Anforderungszuordnung

In der folgenden Tabelle ist die Zuordnung der Anforderungen aus den technischen Anforderungen (s. [TanfDaV]) auf die SW-Einheit, Komponenten, Subkomponenten bzw. Module angegeben:

Anforderung	Komponente	Modul	SW-Einheit Datenverteiler- Applikationsfunktionen						
			Verwaltung	Kommunikation					
				Protokoll- steuerung	Daten- repräsentation	HMAC-MD5 Ver- schlüsselung	Telegramm- verwaltung	TCP-Kommuni- kationskanal	
TDaV-1	Allgemeine Funktionsanforderungen	X	---	---	---	---	---	---	---
TDaV-2	Transparenz	X	---	---	---	---	---	---	---
TDaV-3	Abfrage und Änderung von Konfigurationsdaten		X	---	---	---	---	---	---
TDaV-4	Abfrage Archivdaten		X	---	---	---	---	---	---
TDaV-5	Mehrere Datenhaltungen	---	X	---	---	---	---	---	---
TDaV-6	Konfigurationsinformationen	---	X	---	---	---	---	---	---
TDaV-7	Undefinieren von Aspekten	---	X	---	---	---	---	---	---
TDaV-8	Kommunikationsmechanismen	---	X	---	---	---	---	---	---
TDaV-9	Kommunikation über TCP	---	---	---	---	---	---	---	X
TDaV-10	Austauschbarkeit des Kommunikationsmechanismus	---	---	---	---	---	---	---	X
TDaV-11	Pufferung und Flusskontrolle	---	---	---	---	---	X	---	---
TDaV-12	Prüfung der Zugriffsrechte zur Laufzeit <sup>7</sup>	---	---	---	---	---	---	---	---
TDaV-13	Plausibilitätsprüfungen Schnittstellenübergänge	---	X	X	---	---	---	---	---
TDaV-14	Effizienz der Telegrammverarbeitung	X	---	---	---	---	---	---	---
TDaV1-1	Gesamtfunktion	X	---	---	---	---	---	---	---
TDaV1-2	Logisches Verbindungsobjekt	X	---	---	---	---	---	---	---
TDaV1-3	Kommunikationsverbindung	---	---	---	---	---	---	---	X
TDaV1-4	Mehrere logische Verbindungen	X	---	---	---	---	---	---	---
TDaV1-5	Initialisierung Kommunikationsverbindung	X	---	---	---	---	---	---	---
TDaV1-6	Initialisierung logische Verbindung	---	X	---	---	---	---	---	---
TDaV1-7	Verhandlung Protokollversion	---	---	X	---	---	---	---	---
TDaV1-8	Authentifizierung	---	---	X	---	---	---	---	---
TDaV1-9	Verhandlung Verbindungsparameter	---	---	X	---	---	---	---	---
TDaV1-10	Terminierung	X	---	---	---	---	---	---	---
TDaV1-11	Abfrage Konfigurationsobjekte	---	X	---	---	---	---	---	---
TDaV1-12	Abfrage Metainformation	---	X	---	---	---	---	---	---
TDaV1-13	Stellvertreter für Konfigurationsobjekte	---	X	---	---	---	---	---	---

<sup>7</sup> Diese Anforderung an das Segment 01-Datenverteiler wird durch die SW-Einheit Datenverteiler abgedeckt.

Anforderung	Komponente	Modul	SW-Einheit Datenverteiler- Applikationsfunktionen						
			Verwaltung	Kommunikation					
				Protokoll- steuerung	Daten- repräsentation	HMAC-MD5 Ver- schlüsselung	Telegramm- verwaltung	TCP-Kommuni- kationskanal	
TDaV1-14	Zwischenspeicherung von Konfigurations- informationen	---	X	---	---	---	---	---	---
TDaV1-15	Persistente Speicherung von Konfigurati- onsinformationen	---	X	---	---	---	---	---	---
TDaV1-16	Anmeldung über Parametertelegramme	---	---	X	---	---	---	---	---
TDaV1-17	Anmeldung auf Eingangsdaten	---	X	---	---	---	---	---	---
TDaV1-18	Anmeldung mit Historie	---	X	---	---	---	---	---	---
TDaV1-19	Vorhaltezeitraum	---	X	---	---	---	---	---	---
TDaV1-20	Datenfilterung	---	X	---	---	---	---	---	---
TDaV1-21	Benachrichtigungsobjekte	---	X	---	---	---	---	---	---
TDaV1-22	Mehrfachanmeldungen	---	X	---	---	---	---	---	---
TDaV1-23	Anmeldung Ausgangsdaten	---	X	---	---	---	---	---	---
TDaV1-24	Entgegennahme von angemeldeten Daten	X	---	---	---	---	---	---	---
TDaV1-25	Lesen von aktuellen Daten	X	---	---	---	---	---	---	---
TDaV1-26	Lesen von zwischengespeicherten Daten	---	X	---	---	---	---	---	---
TDaV1-27	Lesen von Archivdaten	---	X	---	---	---	---	---	---
TDaV1-28	Datenfilterung	---	X	---	---	---	---	---	---
TDaV1-29	Datenkonvertierung	X	---	---	---	---	---	---	---
TDaV1-30	Versenden von Daten	X	---	---	---	---	---	---	---
TDaV1-31	Markierung nachgelieferter Daten	X	---	---	---	---	---	---	---
TDaV1-32	Datenkonvertierung	X	---	---	---	---	---	---	---
TDaV1-33	explizite Abmeldung	X	---	---	---	---	---	---	---
TDaV1-34	implizite Abmeldung	---	X	---	---	---	---	---	---
TDaV1-35	Konvertierung der Datendarstellung	X	---	---	---	---	---	---	---
TDaV1-36	Änderung von dynamischen Objekten	---	X	---	---	---	---	---	---
TDaV1-37	Anmeldung auf Konfigurationsänderungen	---	X	---	---	---	---	---	---
TDaV1-38	Telegrammpufferung	---	---	---	---	---	X	---	---
TDaV1-39	Flusskontrolle	---	---	---	---	---	X	---	---
TDaV1-40	Durchsatzprüfung	---	---	---	---	---	X	---	---
TDaV1-41	Priorisierung von Telegrammen	---	---	---	---	---	X	---	---
TDaV1-42	Kommunikationsüberwachung	---	---	---	---	---	X	---	---
TDaV1-43	Telegrammlaufzeitermittlung	X	---	---	---	---	---	---	---
TDaV1-44	Abruf Datenverteilerzustandsinformationen	---	X	---	---	---	---	---	---
TDaV1-45	Schnittstelle DatenverteilerApplikationsfunktionen-	---	X	---	---	---	---	---	---
TDaV1-46	Bestimmung Telegrammgrenzen	---	---	---	---	---	X	---	---
TDaV1-47	Darstellung Grunddatentypen	---	---	---	X	---	---	---	---
TDaV1-48	Zerlegung umfangreicher Transaktionen	---	---	---	---	---	X	---	---
TDaV1-49	Generischer Zugriff	---	X	---	---	---	---	---	---
TDaV1-50	Objektorientierung	---	X	---	X	X	X	X	X
TDaV1-51	Fehlerbehandlung	---	X	---	X	X	X	X	X

Tabelle 5-1: Zuordnung der Anforderungen an die SW-Module