

**Schlussbericht für den Zeitraum: 01.12.1998 bis 30.11.2000
zu dem aus Haushaltsmitteln des BMWi über die
AiF geförderten Forschungsvorhaben**

**Entwurf und Test von
Kommunikationsschnittstellen für die digitale
Leittechnik von Mittelspannungsanlagen**

Mannheim, 9. April 2001

Inhaltsverzeichnis

1	ZUSAMMENFASSUNG	1
2	EINFÜHRUNG	2
2.1	SEKUNDÄRTECHNIK	2
2.2	NORMUNGSAKTIVITÄTEN	5
2.2.1	IEC	5
2.2.2	IEEE	6
2.3	ZIEL DER ARBEIT	7
3	FUNKTIONSSPEZIFIKATION	8
4	PROJEKTIERUNGSLISTE	9
5	MODELLIERUNG ENTSPRECHEND IEC 61850	11
5.1	FUNKTIONSANALYSE	11
5.2	AUFGABENSTELLUNG	11
5.3	DAS IEC 61850 DATENMODELL	11
5.4	DIE MODELLIERUNG	18
6	MODELLIERUNG ENTSPRECHEND GOMSFE (UCA.2)	21
6.1	AUFGABENSTELLUNG	21
6.2	DAS GOMSFE (UCA) DATENMODELL	21
6.2.1	<i>Eigenschaften</i>	21
6.2.2	<i>Die Hierarchie</i>	23
6.3	DIE MODELLIERUNG	25
7	VERGLEICH UND ERGEBNISSE DER MODELLIERUNG	26
7.1	VERGLEICH	26
7.2	ERGEBNISSE	29
8	SIMULATION	33
8.1	AUFGABE	33
8.2	ERGEBNIS	33
9	PILOTSYSTEME MIT PROTOTYPEN	34
9.1	DER AUFBAU	34
9.2	FMS/PROFIBUS	35
9.3	MMS/ETHERNET	37
9.4	ERGEBNISSE	39
10	TESTPHASE	39
10.1	ANFORDERUNGEN	39
10.2	TESTVERFAHREN UND ERGEBNIS	40

10.2.1	<i>Konformitätstest</i>	40
10.2.2	<i>Protokolltest</i>	41
10.2.3	<i>Interoperabilitätstest</i>	41
10.2.4	<i>Abnahmetests</i>	43
10.3	TESTMANAGEMENT	43
10.3.1	<i>Allgemeines</i>	43
10.3.2	<i>Testszenarioszenarien</i>	43
10.3.3	<i>Testspezifikation</i>	43
10.4	AKZEPTANZ DER TESTS	44
10.4.1	<i>Testprotokoll oder Zertifizierung</i>	44
10.4.2	<i>Verantwortlichkeiten</i>	45
10.5	TESTTOOLS	45
10.5.1	<i>Analysen</i>	45
10.5.2	<i>Simulatoren</i>	46
10.6	KONFORMITÄTSTEST NACH IEC 61850	47
10.7	TESTERGEBNISSE	49
11	DER NUTZEN	50
12	AUSBLICK	51
13	VERÖFFENTLICHUNGEN IM RAHMEN DES PROJEKTES	52
13.1	VERÖFFENTLICHUNGEN MIT VORTRAG	52
13.2	VERÖFFENTLICHUNGEN OHNE VORTRAG	52
13.3	VORTRÄGE	53
13.4	AUSSTELLUNGEN	53
14	VERWENDETE ABKÜRZUNGEN	54
15	LITERATUR	55
16	ANHANG A: ANFORDERUNGEN	57
16.1	SYSTEMEIGENSCHAFTEN	57
16.1.1	<i>Identifikation</i>	57
16.1.2	<i>Systemanlauf</i>	57
16.1.3	<i>Überprüfung der Funktionsblöcke</i>	57
16.1.4	<i>Wiederanlauf von Funktionsblöcken</i>	57
16.1.5	<i>Datenerhaltung</i>	58
16.1.6	<i>Zeitverhalten</i>	58
16.1.7	<i>Zeitsynchronisierung</i>	58
16.1.8	<i>Informationssperren</i>	58
16.2	MESSWERTE	59
16.2.1	<i>Umfang</i>	59
16.2.2	<i>Messwertberuhigung</i>	59
16.2.3	<i>Umrechnung</i>	59
16.2.4	<i>Nullpunktunterdrückung</i>	59
16.3	MELDUNGEN	59
16.3.1	<i>Umfang</i>	59

16.3.2	Zeitzuordnung.....	59
16.3.3	Unterdrückung Flattermeldungen	59
16.3.4	Unterdrückung kurzzeitiger Meldungen	60
16.3.5	Störstellungsunterdrückung.....	60
16.3.6	Sammelmeldungsbildung	60
16.3.7	Detailinformationen.....	60
16.3.8	Schalterfallmeldung.....	60
16.3.9	Verwaltung von Meldeabbildern	60
16.4	STEUERUNG.....	60
16.4.1	Umfang	60
16.4.2	Rückmeldung	61
16.4.3	Doppelbetätigungssperre.....	61
16.4.4	Befehlslaufzeitüberwachung	61
16.4.5	Feldverriegelung.....	61
16.4.6	Zulässigkeitsprüfung.....	61
16.4.7	Blockierung der Befehlsrichtung	61
16.4.8	Befehlsherkunft	61
16.4.9	Nachführung	61
16.5	PROTOKOLLIERUNG / ARCHIVIERUNG (OPTIONAL).....	62
16.5.1	Archivierung	62
16.5.2	Protokollierung.....	62
16.6	SCHUTZ	62
16.6.1	UMZ-Schutz.....	62
16.6.2	Informationsumfang.....	62
16.6.3	Rückwärtige Verriegelung (optional)	63
16.6.4	Distanzschutz (optional).....	64
16.6.5	Erdschlusserfassung und Quittierung (optional).....	64
16.6.6	Automatische Wiedereinschaltung - AWE (optional)	64
16.7	PARAMETRIERUNG (OPTIONAL).....	64
16.7.1	Off-line-Parametrierung.....	64
16.7.2	On-line-Parametrierung.....	64
16.7.3	Fernaktivieren.....	64
16.7.4	Lesen von Parametern / Erkennen von Parametersätzen	65
16.8	STÖRSCHRIEBE (OPTIONAL).....	65
16.8.1	Datenumfang	65
16.8.2	Zuordnung	65
16.8.3	Fernauswertung.....	65
17	ANHANG B: PROJEKTIERUNGSLISTE	66
18	ANHANG C: MODELLIERUNG IEC 61850	79
18.1	ABSTRAKTES DATENMODELL NACH IEC 61850.....	79
18.2	MAPPING DATENMODELL IEC 61850 AUF MMS	86
18.3	MAPPING DATENMODELL IEC 61850 AUF PROFIBUS/FMS.....	99
18.4	ABSTRAKTES DIENSTMODELL NACH IEC 61850	109
18.5	MAPPING DIENSTMODELL IEC 61850 AUF MMS.....	114
18.6	MAPPING DIENSTMODELL IEC 61850 AUF PROFIBUS/FMS	119

19	ANHANG D: MODELLIERUNG GOMSFE	125
20	ANLAGE E: TESTSPEZIFIKATION	171
20.1	ANWENDUNGSBEREICH	171
20.2	ALLGEMEINES	171
20.2.1	<i>Aufbau des Standards</i>	172
20.3	PRÜFUNGEN AN EINEM STATIONSLEITGERÄT	172
20.4	PRÜFUNGEN AN EINEM FELDDGERÄT.....	172
20.4.1	<i>Allgemeines</i>	172
20.4.2	<i>Aufbau und Abbau von Kommunikationsverbindungen</i>	173
20.4.3	<i>Prüfungen an Datenobjekten</i>	174
20.4.4	<i>Services</i>	201
20.4.5	<i>Befehlssequenzen</i>	208
20.4.6	<i>Generalabfrage</i>	210
20.4.7	<i>Modellierung der Common Data Classes</i>	211
20.4.8	<i>Uhrzeitsynchronisierung</i>	215
20.4.9	<i>Performance</i>	216
20.5	PRÜFUNGEN AM GESAMTSYSTEM.....	216
20.5.1	<i>Störung eines Feldgerätes</i>	216
20.6	VERGLEICH PROFIBUS/FMS UND ETHERNET/MMS	217
20.7	NEGATIVTEST.....	217
20.7.1	<i>Attribute der Common Data Classes</i>	217
20.8	TESTSZENARIEN	221
20.8.1	<i>Leistungsschalter in Störstellung</i>	221
20.8.2	<i>Leistungsschalter blockiert</i>	221
20.8.3	<i>Doppelter Schaltbefehl</i>	221
20.8.4	<i>Falsche Adresse</i>	221
20.8.5	<i>Kommunikation unterbrochen</i>	222
20.8.6	<i>Netzbelastung</i>	222
20.8.7	<i>Beispiel eines Schaltvorgangs</i>	222
20.9	PRÜFUNGEN UNTER AUßERORDENTLICHEN BETRIEBSBEDINGUNGEN	225
20.9.1	<i>Betrieb bei extremen Temperaturen</i>	225
20.9.2	<i>Betrieb bei Änderung oder Störung der Versorgungsspannung</i>	226
20.9.3	<i>Störungen oder Unterbrechungen des Stationsbusses</i>	227
21	ANHANG F: ENTWICKLUNG EINES TESTSYSTEMS.....	228
21.1	EINFÜHRUNG.....	228
21.2	ANALYSE.....	228
21.3	MODULE DES TESTSYSTEMS.....	230
21.3.1	<i>Das Modul Server</i>	230
21.3.2	<i>Das Modul Client</i>	231
21.3.3	<i>Das Modul Prozesssimulation</i>	231
21.4	DATENBANKEN UND PROTOKOLLIERUNG.....	232
21.4.1	<i>IEC-Datenbank</i>	232
21.4.2	<i>DUT-Datenbank</i>	232
21.4.3	<i>Worksheet-Datenbank</i>	233
21.4.4	<i>Batch-Datenbank</i>	233

21.4.5	<i>Das Modul Datenbankzugriff</i>	233
21.4.6	<i>Das Modul Protokollierung</i>	233
21.5	DAS MODUL BLOCKSCHALTBILD	233
21.6	DAS MODUL GERÄTEIDENTIFIZIERUNG.....	234
21.7	DAS MODUL TESTSZENARIEN	234
21.7.1	<i>Einführung</i>	234
21.7.2	<i>Konformitätsüberprüfung</i>	234
21.7.3	<i>Kommunikations- und Sequenzprüfung</i>	235
21.7.4	<i>Das Modul Negativtest</i>	236
21.8	ÜBERSICHT DER ANWENDUNGSFÄLLE.....	236
21.8.1	<i>Details der Anwendungsfälle</i>	240
22	ANHANG G: ISO/OSI REFERENZMODELL	242
22.1	ARCHITEKTUR DES ISO/OSI REFERENZMODELLS	242
22.2	FUNKTIONEN DER SCHICHTEN	243
22.3	PROTOKOLLE UND PROFILE.....	246
23	ANHANG H: MMS.....	249
24	LITERATUR ANHANG	250

1 Zusammenfassung

Derzeit wird im TC 57 der IEC ein einheitliches Daten- und Dienstmodell für die digitale Leit- und Schutztechnik erarbeitet, das international in der Normenreihe IEC 61850 genormt werden soll. Parallel zu den IEC Arbeiten und mit einem ähnlichen Scope führt in den USA das Electric Power Research Institute (EPRI) im Auftrag der amerikanischen Energieversorgungsunternehmen ein Forschungsprojekt mit der Bezeichnung Utility Communication Architecture (UCA) durch. Im Rahmen des hier vorliegenden Forschungsvorhabens wurden die Modelle verglichen und ihre Leistungsfähigkeit sowie die Machbarkeit überprüft. Die zunächst theoretisch gewonnenen Ergebnisse wurden anhand von speziell hierfür entwickelten Testsystemen mit realen Geräten verifiziert.

Die Funktionsanalyse von Feldgeräten verschiedener Hersteller hat gezeigt, dass die Anforderungen der EVU, die in den Empfehlungen der VDEW für die digitale Stationsleittechnik festgehalten sind, zum großen Teil abgedeckt werden. Der Vergleich amerikanischer und europäischer Anforderungen an die Stationsleittechnik hat jedoch deutliche Unterschiede aufgezeigt. Dementsprechend sind die Ansätze für das Daten- und Dienstmodell des UCA-Entwurfes und des IEC-Normentwurfes unterschiedlich und nur mit größerem Aufwand harmonisierbar.

Ein Problem des amerikanischen Ansatzes ist aus europäischer Sicht die geräteorientierte Modellierung. Das bedeutet, dass jedes neue funktionserweiterte Gerät, das in Zukunft auf dem Markt angeboten würde, neu modelliert und in dem Standard aufgenommen werden müsste. So gut die angewandten Techniken und Ideen dieses Ansatzes auch sind, so inkonsistent sind die UCA-Dokumente. Die europäischen Anforderungen können mit den derzeit vorliegenden UCA-Entwürfen nicht ausreichend abgedeckt werden.

Ein Problem des IEC Entwurfes ist die gewünschte Abwärtskompatibilität zu vorhandenen IEC-Standards (z.B. IEC 60870-5-103). Ein weiteres Problem ist, dass die gewünschte Interoperabilität zwischen Geräten verschiedener Hersteller allein mit Hilfe des Daten- und Dienstmodells nicht erreicht werden kann. Abhilfe wird eine genormte Beschreibung schaffen, mit der die Adressierung der Daten des Datenmodells für ein Gerät festgelegt werden kann. Als Beschreibungssprache wird XML eingesetzt werden. Die Semantik der Beschreibung wird Teil der IEC-Norm sein.

Als Ergebnis des Forschungsprojektes kann festgehalten werden, dass die europäischen Anforderungen an die Stationsleittechnik in Schaltanlagen mit dem jetzt vorliegenden IEC-Normentwurf abgedeckt werden können. Dazu hat das Forschungsprojekt unmittelbar beigetragen, da dessen Ergebnisse fortlaufend in die internationalen Normungsgremien eingebracht wurden. Auf diese Weise sind eine Fülle von Änderungen und Verbesserungen vorgenommen worden und Fehler in den Normentwürfen beseitigt worden. Die enge Zusammenarbeit von Energieversorgungsunternehmen, Herstellern und Mitgliedern der zuständigen IEC-Gremien im Rahmen dieses Projektes war wesentliche Voraussetzung für die Überführung der Ergebnisse in die Normenarbeit und in die zukünftige praktische Anwendung. Damit hat das Projekt dazu beigetragen, dass mit IEC 61850 ein zukunftsweisender internationaler Standard entsteht, der auch für den deutschen Markt in vollem Umfang einsetzbar sein wird. Nutznießer der Projektergebnisse sind die Netzbetreiber als Anwender von Stationsleitsystemen sowie Hersteller von Stationsleitsystemen oder Komponenten der Stationsleittechnik sowie Hersteller von Schaltanlagen und Schaltgeräten, da auch die Feldelektrotechnik

entsprechend IEC 61850 ausgeführt werden wird und damit alle Geräte über entsprechende Kommunikationsfähigkeit verfügen müssen.

Das Ziel des Vorhabens wurde in vollem Umfang erreicht.

2 Einführung

2.1 Sekundärtechnik

Flexible, modulare technische Lösungen in der Energieversorgung sind wichtige Faktoren für das dauerhafte Bestehen im Wettbewerb des Energiemarktes. Einen bedeutenden Beitrag dazu leisten die Sekundärsysteme zum Schutz und zum Steuern von Schaltanlagen der Energieverteilung. Die technischen Anforderungen an diese Systeme steigen ständig. Dies betrifft vor allem die Ansprüche an die Kommunikationsfähigkeiten des Systems. Heute ist die Kommunikation integraler Bestandteil moderner Sekundärsysteme, ihre Struktur orientiert sich am Datenaufkommen, an der maschinellen Verarbeitung und nicht zuletzt am Bediener und Benutzer einer Anlage.

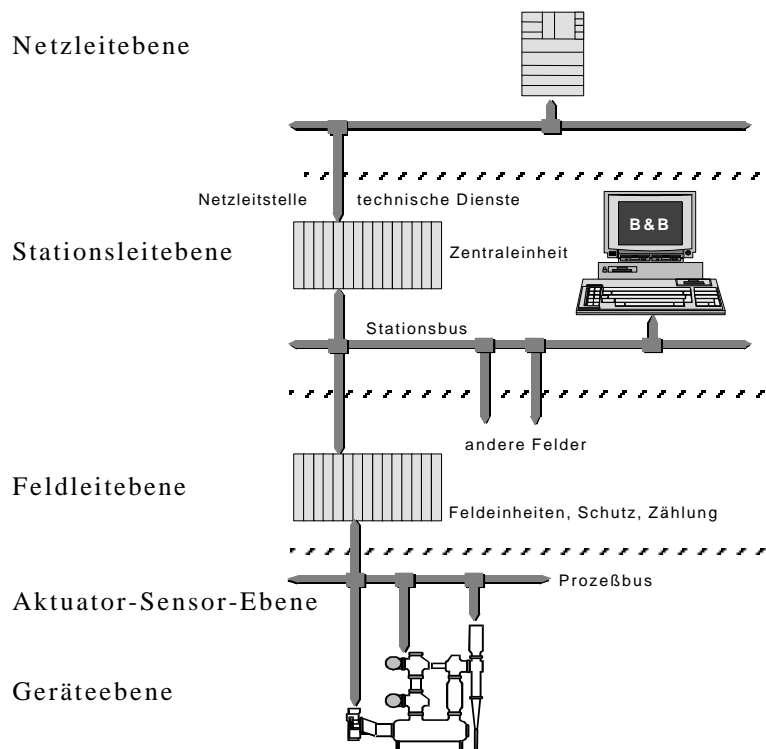


Bild 1 Mögliche Kommunikationsstruktur für die Sekundärtechnik

Es hat sich als zweckmäßig erwiesen, das Sekundärsystem entsprechend Bild 1 hierarchisch in die Ebenen

- Netzleitebene,
- Stationsleitebene,

- Feldleitebene,
- Aktor-Sensor-Ebene,
- Geräteebene

zu gliedern [1]. Die Geräte einer Ebene kommunizieren über ein geeignetes Kommunikationssystem mit den Geräten der nächst höheren und mit denen der nächst niedrigeren Ebene in einem Informationsverbund.

Geräteebene

Die unterste Ebene bilden die primärtechnischen Geräte selbst, z.B. Leistungsschalter, Sammelschienen, Transformatoren, etc. Letztlich sind die Geräte die einzigen Datenquellen und -senken im gesamten System und stellen aus der Sicht des Sekundärsystems den Prozess dar.

Aktor-Sensor-Ebene

Um zu erfahren, welche Zustände, z.B. welche Spannungs- und Stromwerte, an welchen Stellen im Prozess vorliegen, müssen Sensoren verwendet werden. Aktoren dagegen sind Geräte, mit denen auf den Prozess eingewirkt werden kann. Die Aktor-Sensor-Ebene steht über die Prozesskommunikation mit der Feldleitebene in Verbindung. Die Informationen aus dem Prozess werden über eine entsprechende Parallelverdrahtung an die binären und analogen Eingangskarten des jeweiligen Feldleitgerätes weitergeleitet. Diese Prozessebene ist bisher noch nicht digitalisiert worden.

Feldleitebene

In der Feldleitebene befinden sich überwiegend Feldleit- und Schutzgeräte. Sie stehen über die Feldkommunikation mit den Geräten der Stationsleitebene in Kontakt [1].

Stationsleitebene

Heutige Stationsleitsysteme basieren auf Leitrechnersystemen, die in kleineren Stationen zentral in einem Stationsleitgerät zusammengefasst sind, während in größeren Anlagen ein aus einem Stationsleit- und mehreren Feldleitgeräten bestehendes, über Lichtwellenleiter verbundenes Rechnernetzwerk aufgebaut wird. Die Datenübertragungsprotokolle der Verbindungen und teilweise sogar die Funktionsverteilung zwischen Feld- und Stationsleitebene sind herstellerspezifisch, so dass die leittechnischen Komponenten einer Station aus einer Hand kommen müssen.

Netzleitebene

Die Anbindung der Stationsleittechnik an die Netzleittechnik erfolgt über einen Protokollumsetzer, der die Informationen aus dem Stationsleitsystem in das jeweils verwendete Fernwirkprotokoll übersetzt.

Der heutige Stand der Technik innerhalb des Sekundärsystems ist die Integration von Schutz und Stationsleittechnik in ein System mit verteilter Intelligenz [2]. Der numerisch arbeitende Schutz kommuniziert über ein serielles Kommunikationssystem mit den Leitgeräten der Feld- und Stationsleitebene [3]. Auch die Verbindung zur Netzleitstelle, die Fernwirkchnittstelle, ist als serielle Schnittstelle ausgeführt und integraler Bestandteil der Stationsleitebene. Alle

Kommunikationsverbindungen im System sind seriell, lediglich die Anbindung des Prozesses an das Feldgerät geschieht noch mit der Parallelverdrahtung (Bild 2).

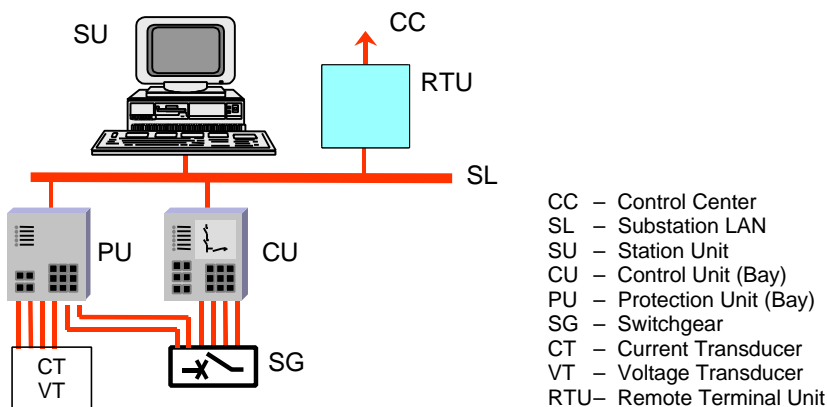


Bild 2 Kommunikationsstruktur „Heute“ (schematisch)

Die Tendenz in der Anlagenkommunikation geht zu einer Vernetzung des gesamten Informationsaustauschs in der Station [4]. Die Kommunikation wird auf allen Ebenen über Bussysteme erfolgen. Zur Anbindung der Schaltanlage wird ein Prozessbus verwendet und zur Anbindung der Feldleitgeräte der Stationsbus. Die schnelle Kommunikation ist sowohl zwischen Feldgeräten als auch zwischen einem Feld- und dem Stationsleitgerät möglich. Durch die Informationsvernetzung wird eine Parallelverdrahtung vermieden, was den Aufwand bei der Verkabelung reduziert.

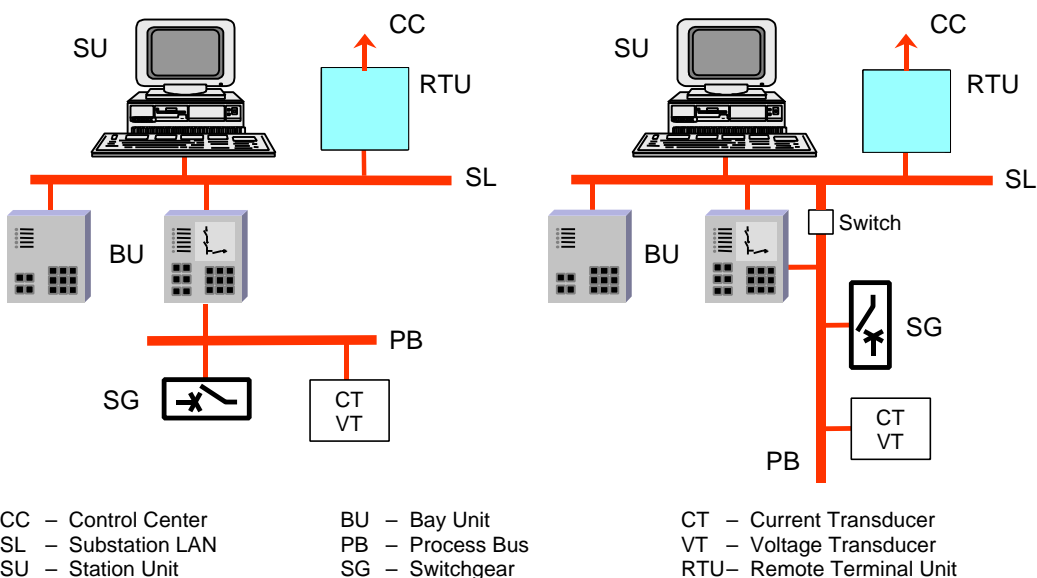


Bild 3 Kommunikationsstruktur „Morgen“ (schematisch)

Um die vielfältigen Aufgaben der Kommunikation zwischen den Geräten des Sekundärsystems erfüllen zu können, bedient man sich eines seriellen Kommunikationssystems. In diesem Fall wird auf das OSI-Referenzmodell [5, 6] für offene Kommunikation zurückgegriffen. Das OSI-Referenzmodell beschreibt die Architektur der Kommunikation anhand von sieben aufeinander aufbauenden Schichten. Die Kommunikation

zwischen den Geräten eines Sekundärsystems fordert Standards, mit denen zum Beispiel die Dienste und deren Verwendung, Baudraten und Regeln zur Codierung der in den Datenpaketen enthaltenen Informationen festgelegt werden. Für jede Schicht sollen entsprechende Funktionen und Eigenschaften definiert und als internationale Protokoll-Standards veröffentlicht werden [7].

Der heutige Markt ist durch herstellereigenspezifische, hardwareorientierte Lösungen geprägt. Es existiert daher eine Vielzahl von Protokollen für die Kommunikation, was im allgemeinen dazu führt, dass Geräte verschiedener Hersteller und selbst Geräte verschiedener Generationen eines Herstellers nicht oder nur mit unverhältnismäßig hohem Aufwand miteinander kommunizieren können. Mit zunehmendem Einsatz moderner Informationssysteme, steigendem Datenaufkommen und durch die immer kürzer werdenden Innovationszyklen von Hard- und Software ist mit einem weiteren Zuwachs an inkompatiblen Protokollen zu rechnen. Die Spezifikation dieser Protokolle ist in der Regel sehr aufwendig und eine Implementierung führt meist zu hohen Kosten. Eine herstellereigenspezifische Lösung hat für den Hersteller den Vorteil, dass er sich mit speziellen Protokollen optimal an seine spezifischen Gegebenheiten anpassen kann. Sie hat dagegen den Nachteil, dass die Kommunikation mit Systemen anderer Hersteller, d.h. die geforderte Interoperabilität, nur mit aufwendigen Protokollumsetzern möglich ist. Dies geschieht oft unter Verlust von Funktionalität und verschlechtertem Zeitverhalten. Daher erfordert die gewünschte Interoperabilität eine klare Definition und Offenheit der Kommunikation über standardisierte Schnittstellen [8]. Für Hersteller und Anwender bringt die Reduktion der Vielfalt bei gleichzeitig relativ kleinem Markt erhebliche Vorteile. Das gilt sowohl für die Konzeption einer neuen Station als auch für das Nachrüsten oder Erweitern von bestehenden Anlagen, da Geräte verschiedener Hersteller in einem Gesamtsystem eingesetzt werden können. Ein wesentlicher Vorteil besteht bezüglich der Lebensdauer der sekundärtechnischen Ausrüstung. Sofern allseits akzeptierte und genormte Schnittstellen existieren, kann eine Anlage auch zu einem späteren Zeitpunkt noch ergänzt oder erweitert werden, ohne dass die gesamte Sekundärtechnik wegen Inkompatibilität zu den dann am Markt verfügbaren Geräten ausgewechselt werden muss.

2.2 Normungsaktivitäten

2.2.1 IEC

Um die geforderte offene Kommunikation der digitalen Technik zu erreichen, haben sich Arbeitsgruppen unter Beteiligung von Energieversorgungsunternehmen und Herstellern gefunden, die das Ziel haben, die Datenmodelle und die Kommunikation von Sekundärsystemen herstellerübergreifend zu normen. Maßgebend für die weitere Entwicklung in der Stationsleittechnik ist die Normungsaktivität des IEC im Technical Committee 57 (TC 57) und innerhalb des TC 57 die Working Groups (WG) 10, 11 und 12. Die WGs erstellen ein Datenmodell für die Daten einer Station und definieren die notwendige Kommunikation um die Daten austauschen zu können. Das TC 57 arbeitet seit November 1995 an der Erstellung einer allgemeinen Normenreihe für den Datenaustausch über einen Stations- und Prozessbus (Bild 3). Eine sekundärtechnische Anlage, im Sprachgebrauch des IEC „Stationsautomatisierungssystem“ genannt, setzt sich danach aus unabhängigen interoperablen intelligenten Einheiten (IED) zusammen, wobei jede Einheit spezifische

Funktionen gegenüber einer oder mehreren Komponenten des Umfeldes ausübt. Die intelligenten Einheiten können dabei Schutzgeräte, Feldleitgeräte, Geräte der Bedienerperipherie, Zentralgeräte für stationsübergreifende Automatisierung, Fernwirkinterfaces, Regler u.a. sein. Der Anspruch an die Standardisierung ist, dass alle Einzelkomponenten des Stationsautomatisierungssystems herstellerunabhängig miteinander kommunizieren können. Die Arbeiten sollen als Norm IEC 61850 "Communication Networks and Systems in Substations" veröffentlicht werden [9].

2.2.2 IEEE

Parallel zu den Normungsaktivitäten des TC 57 läuft in den USA seit 1992 ein Forschungsprojekt im Auftrag des Electric Power Research Institute (EPRI) mit der Bezeichnung Utility Communication Architecture (UCA.2) [10].

Ziel des Projektes UCA ist eine standardisierte Vernetzung aller Komponenten der Leittechnik, wie Netzleitstellen, Kraftwerke, Stationen, Energy Management Systeme, Schaltanlagen, Schnittstelle zum Kunden usw. Als Voraussetzung dafür soll ein einheitlicher Industriestandard für die durchgängige Kommunikation in der gesamten Energieversorgung bezüglich des Schutzes, der integrierten Steuerung, der Datenerfassung und der Möglichkeit, Feldleit- bzw. Stationsleitgeräte verschiedener Hersteller innerhalb einer Station einsetzen zu können, zur Normung bei IEC eingebracht werden. Das Modell, das die Daten der Felder inklusive Schutz und Steuerung beschreibt, ist in dem Datenmodell GOMSFE (Generic Object Models for Substation & Feeder Equipment) beschrieben [10]. Das Modell, das die notwendigen Dienste beschreibt, die benötigt werden um Meldungen, Befehle, Daten etc. zwischen der Primärtechnik und der Sekundärtechnik auszutauschen, ist im Dienstmodell CASM (Common Application Service Models) niedergelegt [10]. Beide Dokumente sind Teile von UCA.2.

Im Juni 1999 wurde ein technischer Report „Utility Communications (UCA™)“ von IEEE veröffentlicht. Inhalt des Reports ist die Definition von Daten und Kommunikationsdiensten für die Kommunikation innerhalb von Stationen und steht damit in Konkurrenz zum Entwurf IEC 61850.

EPRI und IEC einigten sich Anfang 1998, an einem gemeinsamen IEC- Standard zu arbeiten, wobei die bisherigen Ergebnisse von UCA in die bisherige Normungsaktivität der IEC eingebracht werden. Die Zusammenarbeit gestaltet sich jedoch schwierig, da der IEC- und der UCA-Entwurf auf unterschiedlichen Ansätzen basieren. Der IEC-Entwurf verwendet eine abstrakte, funktionen-orientierte Datendefinition, der UCA-Entwurf dagegen eine reale geräte-orientierte Datendefinition. Der IEC-Ansatz ist damit unabhängig von der Kommunikationstechnologie und unabhängig von möglichen neuen Datentypen.

2.3 Ziel der Arbeit

Die Standardentwürfe von IEC und EPRI versprechen eine durchgängige und offene Kommunikation in Stationen. Sie basieren jeweils auf einem Daten- und Dienstmodell. Die Datenmodelle beschreiben die Daten der Felder inklusive Schutz und Steuerung. Die Dienstmodelle beschreiben die notwendigen Dienste, die benötigt werden, um die Daten (Meldungen, Befehle, etc.) zwischen der Primärtechnik und der Sekundärtechnik sowie zwischen den Geräten der Sekundärtechnik auszutauschen.

Ziel des Forschungsvorhabens war es, die Standardentwürfe zu vergleichen und auf Machbarkeit, Einsetzbarkeit und Funktionalität zu überprüfen. Dazu sollten die Entwürfe in die Praxis umgesetzt und mit Hilfe eines Testaufbaus untersucht werden. Die Ergebnisse dieses Forschungsprojektes sollten in die internationalen Normungsgremien einfließen und damit die deutschen Interessen unterstützen. So sollte in Zusammenarbeit mit Energieversorgungsunternehmen, Mitgliedern der zuständigen IEC-Gremien und den Herstellern ein auch für den deutschen Markt zukunftsweisender internationaler Standard entstehen.

Mit Hilfe der Ergebnisse des Forschungsvorhabens sollten bereits im frühem Stadium der Standardisierung die folgenden Anforderungen verfolgt und durchgesetzt werden:

- Fehlerfreiheit der Modelle
- Unbeschränkte Erweiterbarkeit der Modelle
- Vollständigkeit der Modelle bezogen auf die heute vorhandenen Kommunikationsfunktionalitäten
- Durchgängigkeit vom Prozess zur Leitstelle
- Trennung der Applikation (Datenmodell) von der Kommunikation (Dienstmodell) und von der verwendeten Kommunikationstechnologie
- Rücksichtnahme auf die Anforderungen der Energieversorgungsunternehmen, die in den Empfehlungen für die digitale Stationsleittechnik in der Publikation der VDEW [11, 12, 13, 14] niedergelegt sind.
- Einsetzbarkeit von Feld- und Stationsgeräten verschiedener Hersteller innerhalb einer Station.

Das Projekt wurde in vier voneinander abhängige Teilaufgaben strukturiert. Es erstreckt sich über die Definition einer Referenzschaltanlage über erste PC-Simulationsaufbauten zu realitätsnahen Testaufbauten mit realen Geräten. Den Abschluss bildete eine umfassende Testphase.

Der Fokus der Untersuchungen lag auf der Kommunikation über einen Stationsbus, d.h. eine Kommunikation über einen Prozessbus wurde nicht in jeder Teilaufgabe berücksichtigt.

Das Forschungsvorhaben Nr. 11892N wurde aus Haushaltsmitteln des Bundesministeriums für Wirtschaft und Technik (BMWi) über die Arbeitsgemeinschaft industrieller Forschungsvereinigungen "Otto von Guericke" e.V. (AiF) gefördert.

3 Funktionsspezifikation

Die Entwürfe von IEC und UCA beschreiben Daten- und Dienstmodelle für die Kommunikation innerhalb von Stationen. Um die Modelle überprüfen zu können musste eine Schaltanlage als Referenzanlage für das gesamte Projekt definiert werden.

Um von Anfang an einen möglichst nahen Bezug zur Realität zu erhalten, wurde eine reale Mittelspannungsstation, bestehend aus drei Feldern und einem Stationsleitgerät, als Referenzstation verwendet (Bild 4). Die Mittelspannungsebene wurde ausgewählt, weil der Kostendruck dort besonders groß ist. Der erhöhte Kostendruck ergibt sich aus folgenden Gründen:

- Die Anzahl der Mittelspannungs-Stationen ist wesentlich größer als die der Hochspannungs-Stationen.
- Geringere Anforderungen an die Leittechnik und damit an die Kommunikation als in den Hochspannungsstationen.
- In der Mittelspannungsebene ist die Integration der Primär- mit der Sekundärtechnik am weitesten fortgeschritten.
- Schlüsselfertige Bauweise, damit eine hohe Integration.

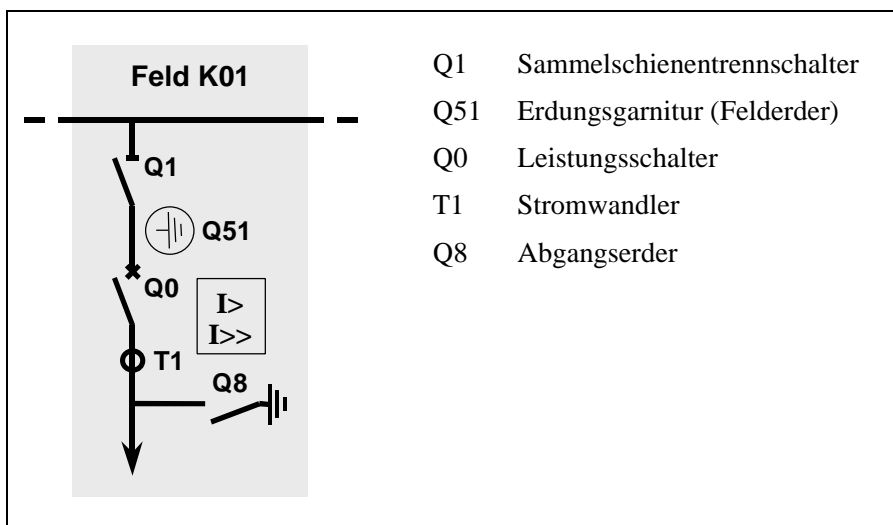


Bild 4 Referenzfeld

Der in dieser Station zu untersuchende Prozess besteht aus zwei luftisolierten, geschotteten Mittelspannungsfeldern, gemäß dem Entwurf "Digitale Stationsleittechnik – Empfehlungen" der VDEW (Vereinigung Deutscher Elektrizitätswerke e.V.). Nur der Leistungsschalter ist motorgetrieben und kann somit ferngesteuert werden. Vor dem Leistungsschalter befinden sich Kugelbolzen zum Anschließen einer Erdungsgarnitur (Felderder Q51). Der von Hand betätigte Sammelschienen trennschalter und Abgangserder ist nur in Melderichtung an die Stationsleittechnik angebunden. Dem Leistungsschalter ist zunächst nur ein ungerichteter, zweistufiger UMZ-Schutz zugeordnet. In weiteren Arbeiten könnte die Schutzeinrichtung um einen Distanzschutz, eine AWE und eine Erdschlusserfassung erweitert werden.

Um die Datenmodelle von IEC und UCA anwenden zu können, mussten für das Referenzfeld bestimmt werden, welche Daten ausgetauscht werden sollen. Dazu wurde eine Funktionsspezifikation erstellt in dem die Anforderungen an die Referenzanlage beschrieben wurden. Die Anforderungen sind im Anhang A: Anforderungen dokumentiert. Die Referenzstation besteht aus drei identischen Feldern gemäß Bild 4. Jedes Feld soll von jeweils einem kombinierten Feldleit- und Schutzgerät (Kombigerät) gesteuert werden. Aus dieser Beschreibung ergibt sich der Konfigurator der Sekundärtechnik gemäß Bild 5.

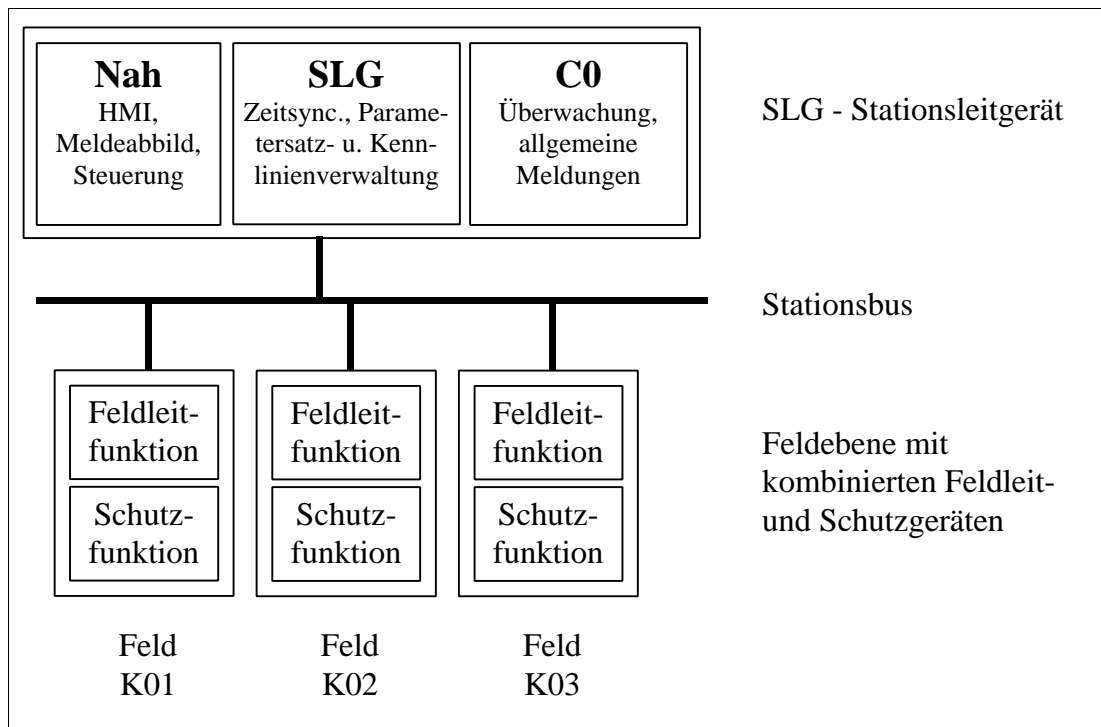


Bild 5 Konfigurator

4 Projektierungsliste

Basierend auf den Anforderungen und der Referenzschaltanlage wurden die notwendigen Daten, Meldungen, Befehle und Messwerte bestimmt. In der Projektierungsliste sind diese Informationen bzw. Aktionen aufgelistet (Anhang B: Projektierungsliste). D.h., nicht die Funktionalität des Schutzes oder eines Schalters wurde beschrieben, sondern die benötigten Aktionen, um die geforderte Funktionalität zu erfüllen.

Als Beispiel soll die Umsetzung der Anforderungen aus

- Abschnitt 16.1.2: Systemanlauf (Seite 57) und
- Abschnitt 16.1.8: Informationssperren (Seite 58)

dienen. Als System soll hierbei die Versorgungsspannung des Schutzgerätes und des Leistungsschalters betrachtet werden. Beide werden durch den Gleichstromkreis 1 versorgt. Im Folgenden sind Anforderungen an das System aufgeführt.

Anforderung an das System:

- Der Ausfall (Fehler „Kommend“) und der Wiedereintritt (Fehler „Gehend“) der Hilfsenergie (220V Gleichspannung) des Sekundärstromkreises für den Leistungsschalter (LS) und den Schutz soll im Kombigerät gemeldet werden.
- Die Überprüfung der Hilfsenergie soll über den Kontakt E14 des Kombigerätes verwirklicht werden.
- Der Vorfall (Ausfall der Hilfsenergie) soll in die Ereignisliste (E) eingetragen werden.
- Der Vorfall soll in die Warnmeldeliste als Warnung (W) eingetragen werden.
- Es soll ein Alarm abgesetzt werden.
- Der Vorfall soll im Meldebereich der Anzeige des Displays des Kombigerätes dargestellt werden (A).
- Es soll für diese Funktion eine Infosperre (INFSPE) für die Nahsteuerung (NAH) gesetzt werden können.

Entsprechend dieser Anforderung ergibt sich der erste Eintrag in der Projektierungsliste im Anhang B: Projektierungsliste.

Tabelle 1 Ausschnitt aus der Projektierungsliste

Informationen		E	V	Kontakt	Bemerkung
1. Systeminformationen					
Sekundärstromkreise					
220V GS 1 FEH	KOM / GEH	W	A	E14	Gleichstromkreis 1 (GS): Schutz, LS
INFSPE NAH					Informationssperre Nahsteuerung

Das E (Ereignis) in der Tabellenüberschrift steht für die Ereignisliste und das V (Verarbeitung) für die Warnmeldeliste.

Ein A (Anzeige) in der Spalte V steht für die Forderung, einen Vorfall zusätzlich zu einem Listeneintrag auch im Anzeigebereich (Display) eines Feldgerätes anzuzeigen.

Ein W (Warnung) in der Spalte E steht für die Forderung, dass ein Vorfall als Warnung in die Ereignisliste und in die Warnmeldeliste eingetragen werden soll und zusätzlich die Alarmhupe aktiviert werden soll.

Die Projektierungsliste dient als Grundlage für die folgenden Teilaufgaben.

5 Modellierung entsprechend IEC 61850

5.1 Funktionsanalyse

Damit ein Bezug zur Realität gegeben ist, wurden die Anforderungen bzw. die Projektierungsliste an den kombinierten Feldleit- und Schutzgeräten von drei Herstellern gespiegelt. Innerhalb dieser Funktionsanalyse der Kombigeräte der Hersteller wurden die Funktionen herausgefiltert, die alle Geräte zur Verfügung stellen. Mit Rücksicht auf die Teilaufgabe „Pilotsysteme mit Prototypen“ wurde zunächst nur die Schnittmenge dieser Funktionen für die folgenden Teilaufgaben weiterverwendet.

Der Vergleich der Funktionalität der Geräte, also die Funktionsanalyse, ist nicht schriftlicher Bestandteil dieses Berichtes, da die von den Herstellern zur Verfügung gestellten Unterlagen vertraulich behandelt werden müssen. Daher wird aus dieser Arbeit nicht ersichtlich werden, welche der geforderten Funktionen von welchem Hersteller bzw. welchem Kombigerät nicht im Leistungsumfang enthalten sind. Ziel dieser Arbeit war auch nicht der Vergleich der Geräte, sondern vielmehr der Vergleich der beiden Normungsansätze von IEC und UCA.

5.2 Aufgabenstellung

Die aus der Funktionsanalyse (Abschn. 5.1) ermittelte Schnittmenge der Kommunikationsfunktionen und Objekte aus der Projektierungsliste werden mit Hilfe des Datenmodells entsprechend IEC 61850 modelliert. Zum Verständnis der Modellierung wird im folgenden Abschnitt am Beispiel der Referenzschaltanlage (Bild 4, Bild 5) das Daten- und Dienstmodell von IEC 61850 erläutert.

5.3 Das IEC 61850 Datenmodell

Die zukünftige Normenreihe IEC 61850 wird als gemeinsame Norm für alle Kommunikationsaufgaben innerhalb der Station ausgelegt. Der Bereich erstreckt sich von der Stationsebene über die Feldebene bis in den Prozess hinein, d.h. künftige Entwicklungen hinsichtlich digitaler Schnittstellen zum Prozess werden bereits berücksichtigt. Der Ansatz von IEC 61850 resultiert aus diesen Erfahrungen der letzten 20 Jahre. Um die Offenheit eines neuen Stationsautomatisierungssystems (SAS) zu gewährleisten, werden konsequent die Funktionen (Applikationen) von der Kommunikation getrennt. Damit ergibt sich die Möglichkeit, für ein SAS ein oder auch mehrere Kommunikationsprofile unter Verwendung von Standardbussen festzuschreiben, ohne Funktionen oder Funktionsverteilungen neu definieren zu müssen.

Bild 6 zeigt schematisch diesen Grundgedanken. Getrennt von der Applikation wird die Basiskommunikation (ACSI – Abstract Communication Service Interface) als anwendungsunabhängiger gemeinsamer Kern für darauf aufsetzende konkrete Kommunikationsverfahren definiert. Im ACSI sind die Datenmodelle und Kommunikationsdienste abstrakt und strukturell festgelegt.

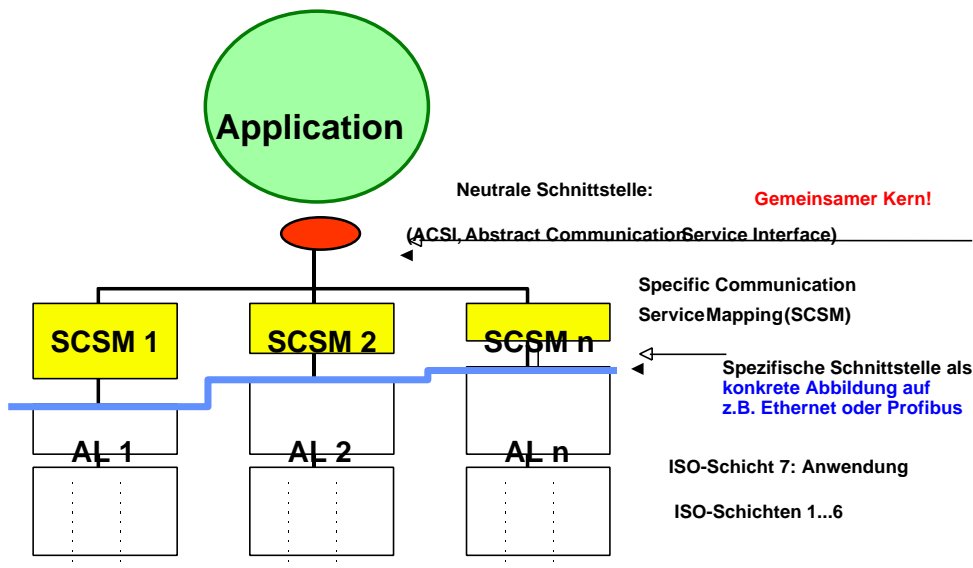


Bild 6 Gemeinsamer Kern und spezifische Abbildung auf Standardkommunikationsverfahren

Bei der spezifischen Abbildung auf ein konkretes Kommunikationsverfahren ist zwar streng nach den Regeln des ACSI zu verfahren, allerdings sind die Codierung sowie die Auswahl als optional geltender Strukturelemente und Dienste des ACSI frei definierbar. Dadurch soll vor allem Offenheit für unterschiedliche Marktanforderungen sowie für die künftige Technologieentwicklung erhalten werden. Ebenfalls wird auf die Festlegung von Hardwarestrukturen oder Gerätestandards verzichtet. Für notwendige physikalische Gerätedaten werden andere Standards referenziert.

Dieser offene Ansatz der IEC 61850 erfordert bei allen Vorteilen für Hersteller und Anwender jedoch ein ausgeprägtes Systembewusstsein und konsequente Objektmodellierungen. Sonst wird die Zukunftssicherung der Investitionen einerseits und die uneingeschränkte Innovationsfreiheit für Hard- und Softwarekomponenten andererseits nicht gelingen. Schließlich ist IEC 61850 keine exakte Bauvorschrift, die „plug and play“ garantiert, sondern eine Anwendungsvorschrift bei deren Umsetzung „plug and play“ zu erreichen ist.

Die Hierarchie des Datenmodells besteht aus fünf Ebenen (Bild 7). Zur Darstellung der Hierarchie wurde ein Klassendiagramm gemäß der "Unified Modelling Language" (UML) verwendet. Ein Dreieck gibt an, dass eine Klasse eine Spezialisierung einer anderen Klasse ist. Die Raute gibt an, dass sich eine Klasse aus einer bestimmten Anzahl von anderen Klassen zusammensetzt. Die Pünktchen sind keine UML-Symbole, sondern geben an, dass sich das Diagramm an der Stelle eigentlich fortsetzt.

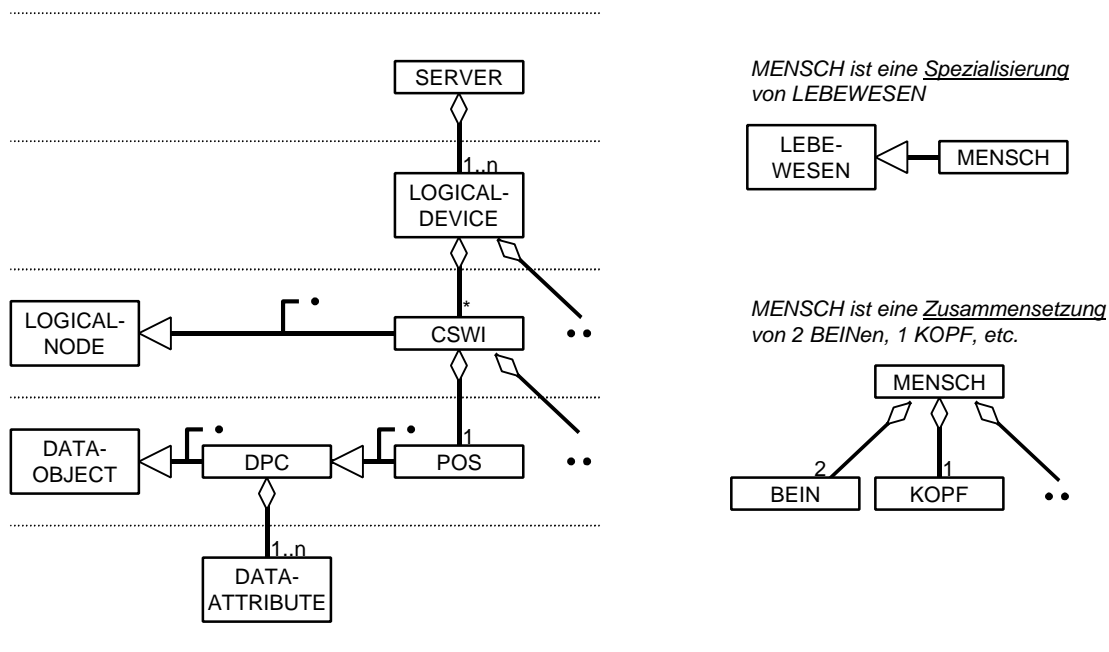


Bild 7 Hierarchie des Datenmodells von IEC 61850

Tabelle 2 Teilfunktionen und Datenobjekte für Steuerung, Überwachung und Automatische Wiedereinschaltung

Teilfunktion Information	Logical Node Data Object	Common Data Class
Leistungsschaltersteuerung	CSWI1 (Switch Controller)	
Position	Pos (Position)	DPC
Trennersteuerung	CSWI2 (Switch Controller)	
Position	Pos (Position)	DPC
Erdersteuerung	CSWI3 (Switch Controller)	
Position	Pos (Position)	DPC
Messwertverarbeitung	MMXU (Measurand Unit)	
Strom IL1	APhSARMS	MV
Binär Ein-/Ausgänge	GGIO (General Input Putput)	
GS1 fehlt	PwrSupAlm1 (Power supply alarm)	SPS
GS2 fehlt	PwrSupAlm2 (Power supply alarm)	SPS
GS Mot.Spg. fehlt	PwrSupAlm3 (Power supply alarm)	SPS
Autom. Wiedereinschaltung	RREC (Autorecloser)	
Funktion aktiv	EnaFct (Logical Node on - not off)	SPC
Funktion nicht bereit	ARNR (Autorecloser not ready)	SPS
AWE ausgeführt	ARSuc (AR cycle successful)	SPS/T

In der dritten Hierarchiestufe (Logical Node) werden die Aufgaben der Kombigerätes (siehe Abschnitt 3) in Teilfunktionen aufgeteilt. Pro angeschlossenem Primärschaltgerät wird zunächst eine Teilfunktion definiert, außerdem für die Messwertverarbeitung und für die Automatische Wiedereinschaltung (Tabelle 2) sowie für den Schutz (Tabelle 3). In den Tabellen sind in der linken Spalte die Teilfunktionen angegeben, die die Sekundärfunktionen des Mittelspannungsfeldes abdecken.

Tabelle 3 Teilfunktionen und Datenobjekte für Schutzaufgaben

<i>Teilfunktion Information</i>	<i>Logical Node Data Object</i>	<i>Common Data Class</i>
Überstromzeitschutz	PTOC (<i>Time Overcurrent</i>)	
Anregung L1	AIA (<i>Alarm Phase A</i>)	SPS
Anregung L2	AIB (<i>Alarm Phase B</i>)	SPS
Anregung L3	AIC (<i>Alarm Phase C</i>)	SPS
Anregung N	AIN (<i>Alarm Neutral</i>)	SPS
Auslösung Stufe 1	TrPTOC (<i>Trip I ></i>)	SPS/T
Auslösung Stufe 2	TrPIOC (<i>Trip I >></i>)	SPS/T
Erdschlussrichtungserfassung	PDEF (<i>Directional Earthfault</i>)	
Erdschluss vorwärts	FoEF (<i>Foreward earthfault</i>)	SPS
Erdschluss rückwärts	ReEF (<i>Reverse earthfault</i>)	SPS
Schutz allgemein	PBRO (<i>Basic Relay Object</i>)	
General-Anregung	GAI (<i>General alarm</i>)	SPS
General-Auslösung	GTr (<i>General Trip</i>)	SPS/T

Tabelle 4 Teilfunktion für übergeordnete und gerätespezifische Informationen

<i>Teilfunktion Information</i>	<i>Logical Node Data Object</i>	<i>Common Data Class</i>
Gerät allgemein	LLN0 (<i>Logical Node Zero</i>)	
Selbstüberwachung - Warnung	SupAI (<i>Supervision alarm</i>)	SPS
Selbstüberwachung - Störung	DisDS (<i>Logical Node disturbance</i>)	SPS
Steuerung Ort / Fern	LocDS (<i>Local operation - not remote</i>)	SPS
Informationssperre	IEBI (<i>Information exchange blocking</i>)	SPC
LED Quittierung	LED (<i>LED reset</i>)	SPC
Parametrierung läuft	PaSet (<i>Parameter setting running</i>)	SPS
Generalabfrage läuft	GI (<i>General Interrogation running</i>)	SPS
aktiver Parametersatz (Kennlinienumschaltung)	Ch (<i>activate characteristic</i>)	ISC

IEC 61850-7-4 enthält eine Liste von etwa 100 Teilfunktionen, die fast alle erdenklichen Aufgaben einer Stationsleittechnik abdecken sollen. Die Teilfunktionen werden dort Logical Nodes (LN) genannt. Dieser Begriff wurde gewählt, weil man sich die Teilfunktionen als Knotenpunkte in einem Informationsnetzwerk vorstellen kann, zwischen denen Informationen ausgetauscht werden. Auf der rechten Seite in den Tabellen sind die Abkürzungen und Bezeichnungen der entsprechenden Logical Nodes angegeben, wie sie im Teil 7-4 definiert sind.

In der vierten Hierarchieebene des Datenmodells werden nun die Objekte der Prozessinformationen aufgeführt, die den jeweiligen Teilfunktionen zugeordnet sind. Die drei Tabellen oben enthalten alle Objekte der Prozessinformationen, die zum Betrieb der Referenzschaltanlage gemäß den Anforderungen notwendig sind. In IEC 61850 werden diese Informationen Data Objects (Datenobjekte) genannt. Zu jedem Logical Node sind im Teil 7-4 die zwingend erforderlichen und möglichen Data Objects aufgelistet. Auch sie haben – wie die Logical Nodes – alle eine definierte Abkürzung. Die Liste der Logical Nodes mit zugehörigen Data Objects im Teil 7-4 stellt gewissermaßen einen Baukasten dar, aus dem die gewünschten Funktionen einer Stationsleittechnik zusammensetzen werden können. Es ist ein Grundprinzip von IEC 61850, dass dieser Baukasten völlig unabhängig von den verwendeten leittechnischen Geräten und der verwendeten Kommunikationstechnologie (Ethernet, Profibus, etc.) ist.

Tabelle 5 Attribute der Common Data Class "Controllable Double Point" hier verwendet für die Attribute eines Leistungsschalters

<i>Detailinformation</i> Wertebereich	<i>Data Attribute</i> D.A. Content
Sollzustand ein / aus	ctlVal (control value) BOOLEAN { off (FALSE), on (TRUE) }
Istzustand Zwischenstellung / aus / ein	stVal (status value) ENUMERATED { intermediate (0), off (1), on (2), bad-state (3) }
Qualität ungültig / nachgeführt / test / ...	q (quality) ...
Zeitstempel Datum + Uhrzeit	t (timestamp) ...
Befehlsherkunft fern / nah / ort / ...	orCat (originator category) ...
	<i>weitere ...</i>
	Controllable Double Point - DPC

Es gibt auch Informationen, die den Zustand eines leittechnischen Gerätes beschreiben. Die Teilfunktion "Gerät allgemein" in Tabelle 4 stellt daher eine gewisse Ausnahme dar, da hier die Informationen zugeordnet sind, die den Zustand des Kombigerätes beschreiben. Würde anstelle des Kombigerätes ein separates Schutzgerät und Feldleitgerät eingesetzt werden, so würde sich an der Datenstruktur in Tabelle 2 und Tabelle 3 nichts ändern (geräteunabhängig). Die Informationen in Tabelle 4 müssten jedoch diesen Geräten angepasst werden. Diese übergeordneten Teilfunktionen werden dem Logical Node Zero zugeordnet.

Mit Hilfe der logischen Knoten und der Datenobjekte wird die Adressierung zu den Prozessinformationen hergestellt. Die Daten des Prozesses werden als Attribute einer Datenklasse im Teil 7-3 hinterlegt. Im Teil 7-3 sind etwa 20 solcher Common Data Classes definiert, in dem für jede Klasse eine sinnvolle Auswahl an Attributen zusammengefasst ist. Jedem DataObject wird im Teil 7-4 eine Datenklasse zugewiesen, in der die Attribute enthalten sind, die für dieses DataObject notwendig sind. Um z.B. den Status eines Leistungsschalters zu erhalten, muss das DataObject „Pos“ die Datenklassen „Controllable Double Point (DPC)“ zugewiesen bekommen. Das DataObject erbt damit alle Attribute der Datenklasse DPC. Diese Vorgehensweise entspricht dem objektorientierten Ansatz [15].

In Tabelle 5 sind die notwendigen Attribute aufgelistet, die eine Schalterstellung beschreiben. Die Attribute heißen in IEC 61850 DataAttributes (Datenattribute). Für die DataAttributes sind im Teil 7-3 Abkürzungen definiert. Da es sich hier um die unterste Hierarchieebene handelt, ist für jedes Attribut ein definierter Wertebereich (Data Attribute Content) angegeben. In der Datentechnik werden diese Wertebereiche bekanntermaßen durch Datentypen wie z.B. INTEGER oder BOOLEAN ausgedrückt.

Die in Tabelle 5 definierte Klasse nennt sich Controllable Double Point (was etwa soviel bedeutet wie "schaltbare 2-Bit Information") und wird für das Data Object verwendet, das die Informationen der Schalterstellung enthält. Die rechte Spalte in den Tabellen 2 bis 4 zeigt, zu welcher Common Data Class das jeweilige DataObject gehört. Die Bedeutung der Abkürzungen kann im Teil 7-3 nachgelesen werden.

Tabelle 6 Attribute der Common Data Class "Single Point Status"

Detailinformation Wertebereich	Data Attribute D.A. Content
Istzustand kommen / gehen	stVal (status value) BOOLEAN { off (FALSE), on (TRUE) }
Qualität	q (quality) ...
Zeitstempel	t (timestamp) ...
Single Point Status - SPS	

Die meisten DataObjects, aus denen Einzelmeldungen gebildet werden, sind von der Klasse Single Point Status ("1-Bit Zustand"). Diese Common Data Class ist in Tabelle 6 dargestellt. Auch Informationen, die nur einen Zustand annehmen können (Wischer-Signale – z.B. Schutz-Aus) werden mit dieser Klasse gebildet. Solche DataObjects sind im Teil 7-4 durch ein "T" wie transient gekennzeichnet.

Es besteht die Möglichkeit eine Anzahl von DataObjects zu einem DataSet zusammenzufassen. Somit besteht eine effiziente Methode, alle DataObjects eines DataSets anzusprechen, ohne die Adresse jedes einzelnen DataObjects zu übertragen, sondern lediglich die Adresse zur Identifikation des DataSets. Die Verwendung von DataSets entstammt dem

amerikanischen Normungsansatz UCA.2. Bei der Übertragung gemäß UCA.2 wird zur Adressierung eines Objektes eine ASCII-Zeichenkette übertragen, die einige Dutzend Bytes lang sein kann. Um die Datenmenge bei der Übertragung vieler Objekte zu reduzieren, besteht daher die Möglichkeit die Werte der DataObjects als Paket und nur mit einer Adresse, die das DataSet identifiziert, zu übertragen. Damit der Empfänger einer solchen Nachricht mit den Werten etwas anfangen kann, muss er natürlich wissen, welche DataObjects in welcher Reihenfolge zu einem DataSet gehören.

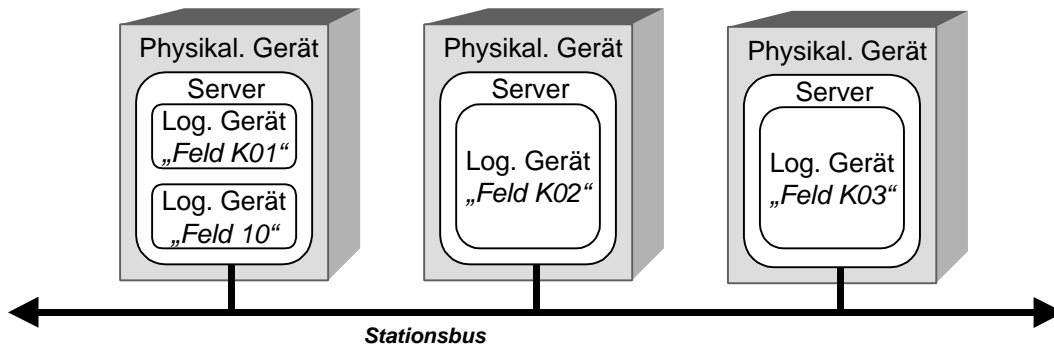


Bild 8 Anwendung von "Server" und "Logical Device"

Die wesentlichen drei Hierarchieebenen des Modells wurden nun vorgestellt. Das vollständige Modell in IEC 61850 beginnt in der obersten Ebene mit dem Objekt Server. Ein Server umfasst die gesamte Kommunikationssoftware eines leittechnischen Gerätes, das an den Bus angeschlossen ist. Ein Server wiederum kann ein oder mehrere Logical Devices (Logische Geräte) umfassen. Bild 8 zeigt dazu ein Beispiel. Drei Kombigeräte sind an einen Bus angeschlossen und sollen die Referenzschaltanlage steuern. Das mittlere Gerät steuert das Feld K02, das rechte Gerät Feld K03. Das linke Kombigerät hingegen steuert Feld K01 und zusätzlich ein Erdungs- und Messfeld K10. Dazu lässt sich das "physikalische" Gerät im Datenmodell in zwei "logische" Geräte aufteilen. Das ermöglicht z.B. eine Informationssperre für Feld K01 zu aktivieren, während die Informationen des Feldes K10 weiterhin übertragen werden können. Ein Logical Device setzt sich zusammen aus beliebig vielen aller in Teil 7-4 aufgelisteten Logical Nodes.

Die Zugriffsverfahren auf die Daten bzw. die Vergabe von Befehlen, das Anfordern von Meldungen usw. werden als Services (Dienste) bezeichnet. Entsprechend dem in IEC 61850 verwendeten Client-Server-Modell werden die Dienste in bestätigte und unbestätigte Dienste unterschieden.

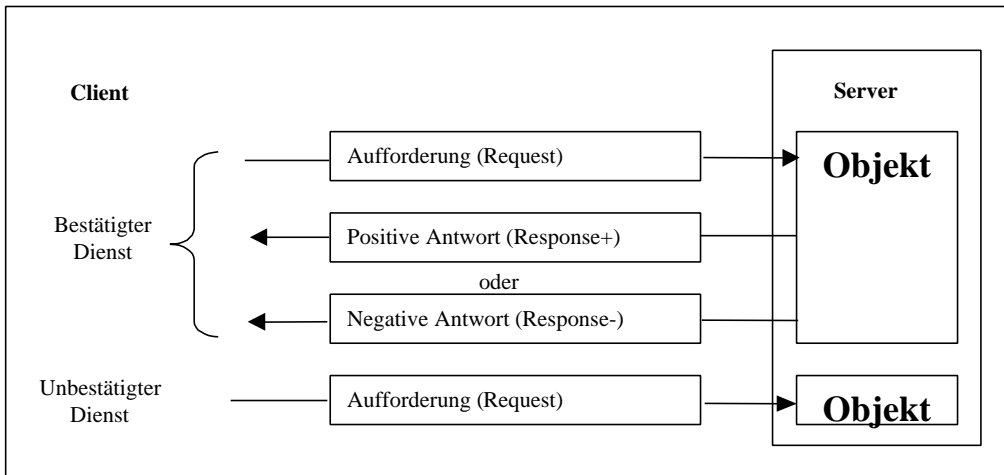


Bild 9 Darstellung bestätigter und unbestätigter Dienste

Bei einem bestätigten Dienst gibt der Server nach einer Aufforderung durch den Client eine positive oder negative Antwort, je nachdem ob er der Aufforderung nachkommen konnte oder nicht. Mit einem bestätigten Dienst kann z.B. die Ausführung eines Befehls festgestellt werden. Bei einem unbestätigten Dienst wird keine Bestätigung für die Ausführung des Befehls übermittelt (Bild 9). IEC 61850 beinhaltet ca. 50 Dienste, gruppiert in 13 Modellen, die in IEC 61850-7-2 definiert sind. Das Report Model beschreibt Dienste zur Übermittlung eines Reports, z.B. den Status eines Schalters, vom Server zum Client. Das Control Model beschreibt Dienste um den Prozess zu steuern. Die Dienste enthalten als Parameter in der Regel die Adresse der Daten, die übertragen werden sollen.

5.4 Die Modellierung

Im Kapitel 18 ist die Modellierung der Referenzschaltanlage vollzogen. Im Folgenden wird ein Beispiel für die Modellierung eines Leistungsschalters (LS) vorgestellt. Es soll der Status des LS abgefragt und anschließend soll der LS eingeschaltet werden.

Die Modellierung unterteilt sich in drei Schritte:

- Adressierung aller Betriebsmittel und Funktionsblöcke
- Zuordnen der erforderlichen Attribute
- Auswahl der Kommunikationsdienste

Adressierung, Attribute:

Entsprechend dem Datenmodell von IEC 61850 Teil 7 müssen zunächst das Schaltfeld und die zugehörigen Komponenten des Schaltfeldes der Hierarchie des Datenmodells zugeordnet werden. Der Begriff des Servers, der Teil des Datenmodells ist, wird hier der Einfachheit wegen weggelassen. Ein Logical Device (LD) soll den Schutz und die Steuerung des gesamten Feldes C02 übernehmen. Der Name des LDs wird auf C02 festgelegt. Um den LS ansprechen zu können stellt die Norm den Logical Node CSWI (Control Switch) zur Verfügung. CSWI ist eine Klasse. Um die Klasse verwenden zu können, muss eine Instanz der Klasse gebildet werden: "Q0CSWI1". Für die Erfassung des Status sowie zur Steuerung

des LS stellt die Norm das Data Object "Pos" (Position) zur Verfügung. Pos ist eine Unterklasse der Klasse Common Data Class "DPC" (Double Point Control). Von dieser Klasse erbt Pos die Attribute, die den Status sowie einige weitere Informationen (Qualitätsbit, Zeitstempel, etc.) bereitstellt. Aus dieser Zuordnung ergibt sich die vollständige Adressierung des LS "Q0": C02/Q0CSWI1.Pos. Die Information, welchen Schaltzustand der LS hat, kann dem Attribut "stVal" entnommen werden. Das Attribut "q" gibt Auskunft über die Qualität des Wertes "stval" und das Attribut "t" beinhaltet den Zeitstempel, d.h. die Zeit an dem die Information abgefragt wurde. Der Zusammenhang wird in Bild 10 und Bild 11 dargestellt.

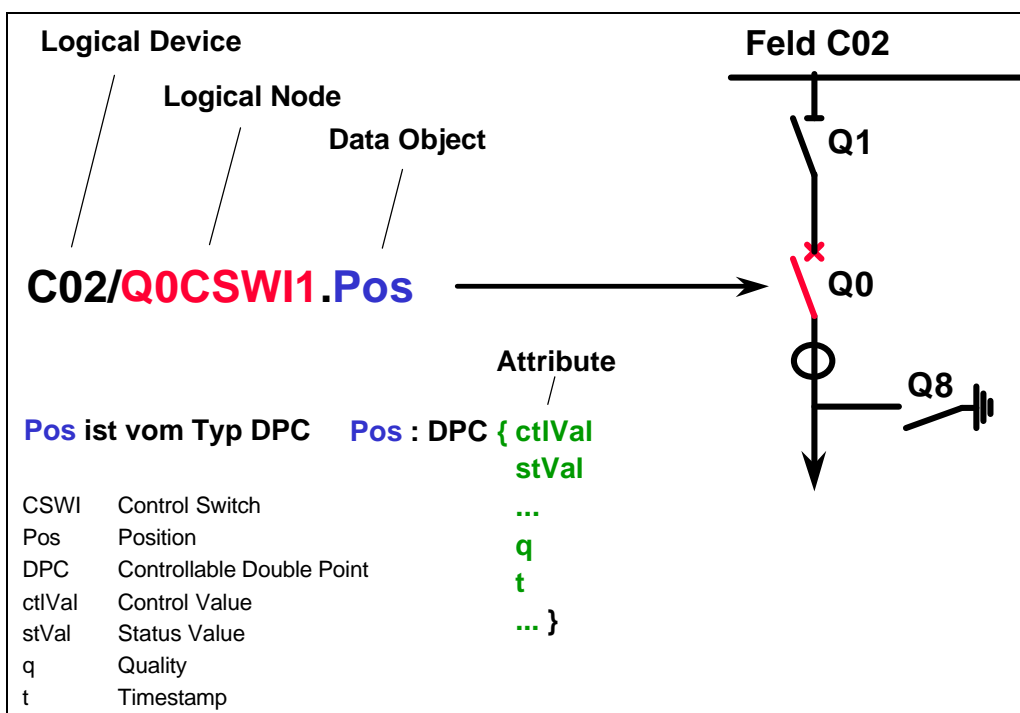


Bild 10 Adressierung

In Bild 11 werden die Datenobjekte mit ausgewählten Attributen beim Abfragen des Status und beim Einschalten des Leistungsschalters dargestellt:

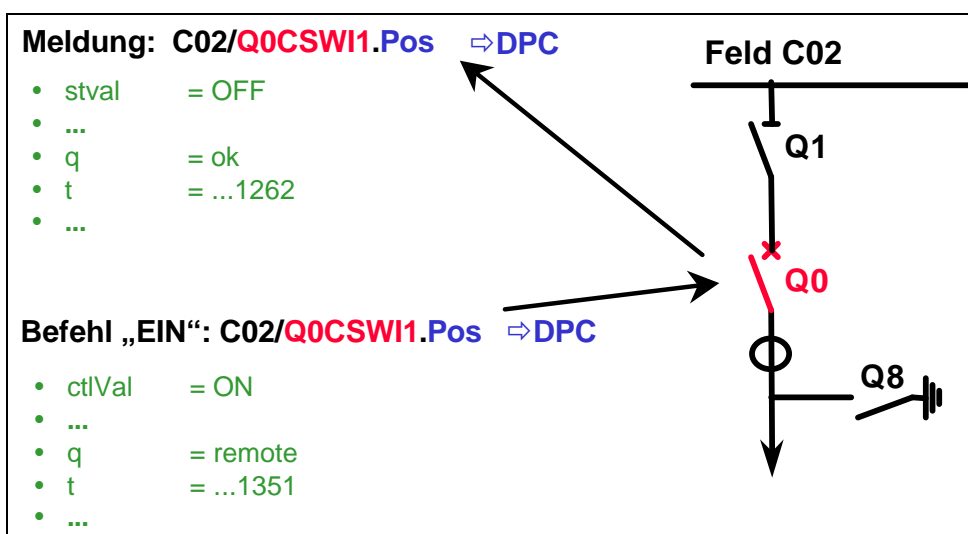


Bild 11 Beispiel Meldung und Befehl

Kommunikationsdienste:

Um den Leistungsschalter einzuschalten, muss das Stationsleitgerät mit Hilfe von Kommunikationsdiensten Befehle über den Stationsbus an das entsprechende Feldgerät absenden. Hierbei wird im Stationsleitgerät der Wert des Attributes "ctlVal" auf "on" gesetzt, die Zeit an dem der Befehl ausgeführt wird in das Attribut "t" geschrieben, das Qualitätsbit "q" und einige weitere Attribute werden gesetzt. Das Qualitätsbit erhält die Information "Befehl erfolgte über die Fernsteuerung". Nach einer vorgeschriebenen Sequenz (Folge von Befehlen und Meldungen, siehe Bild 12) wird der LS (Device) mit Hilfe der Dienste und Attribute über das Feldgerät "C02" eingeschaltet. Zu Beginn fragt das Stationsleitgerät (Client) das Feldgerät (Server) mit dem Dienst "SelectWithValue Request" ob der LS eingeschaltet (on) werden darf. Nach entsprechender Prüfung (Check Interlocking) gibt der Server mit dem Dienst "SelectWithValue Response +" die positive Antwort mit den entsprechenden Attributen zurück. Im nächsten Schritt schickt der Client einen Schaltbefehl mit "Operate Request (on)" an den Server. Der Befehl wird nochmals auf Ausführbarkeit geprüft und der entsprechende Ausgang im Feldgerät wird aktiviert. Die Aktivierung des Antriebes des LS kann über einen Prozessbus oder auf konventioneller Art erfolgen. Der Client erhält die Antwort: „Schaltbefehl wird ausgeführt“ ("Operate Response +"). Die Zwischenstellung während der Laufzeit des Schaltgerätes wird innerhalb eines definierten Zeitfensters nicht gemeldet (Störstellungsunterdrückung). Sobald das Feldgerät die Schalterstellung EIN erkennt, wird diese mit dem Dienst "Report Request (on)" an das Stationsleitgerät zurückgemeldet und der Ausgang abgesteuert (Deactivate Output). Nach erfolgreichem Schalten erhält der Client über den Report Dienst "CmdTerm Request +" die Meldung: „Schaltvorgang erfolgreich abgeschlossen“.

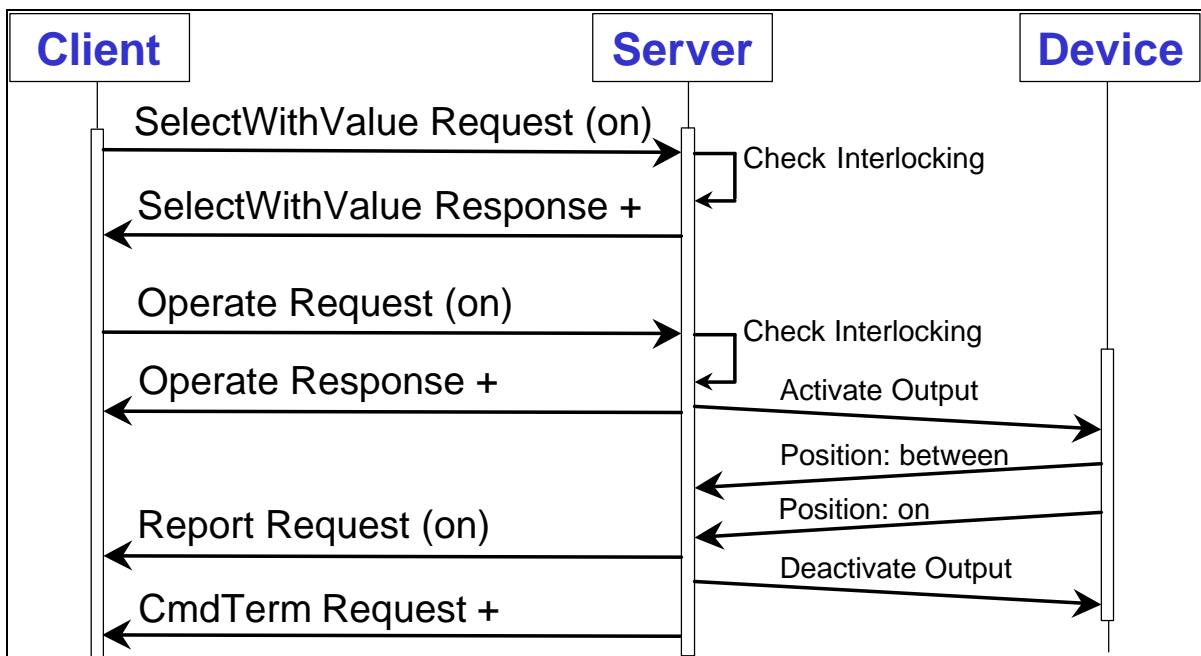


Bild 12 Kommunikationsdienste

6 Modellierung entsprechend GOMSFE (UCA.2)

6.1 Aufgabenstellung

Die aus der Funktionsanalyse (Abschn. 5.1) ermittelte Schnittmenge der Kommunikationsfunktionen und -objekte aus der Projektierungsliste werden mit Hilfe des GOMSFE Datenmodells modelliert. Dazu soll dieses Datenmodell zunächst näher erläutert werden.

6.2 Das GOMSFE (UCA) Datenmodell

6.2.1 Eigenschaften

UCA besteht aus drei Dokumenten die im Bild 13 dargestellt sind. Im Rahmen dieses Forschungsprojektes werden nur die Dokumente GOMSFE und CASM berücksichtigt, da diese das Daten- und Dienstmodell für die Kommunikation innerhalb von Stationen beinhalten. GOMSFE (Generic Object Model for Substation & Feeder Equipment) beschreibt die Informationsobjekte der Sekundär- und Primärtechnik, deren Daten innerhalb der Leittechnik ausgetauscht werden sollen. Das Datenmodell GOMSFE ist geräteorientiert. Die Geräteorientierung liegt in der Tatsache begründet, dass die Daten aller vorhandenen Geräte der amerikanischen Hersteller jeweils im Datenmodell aufgenommen wurden und nicht, wie beim abstraktem IEC Datenmodell, die Daten der Funktionen (Schutz, Steuerung, etc.). Das bedeutet, dass ein und dasselbe Datenobjekt, z.B. das Datenobjekt für den Messwert „Spannung“, in verschiedenen Ausprägungen auftritt.

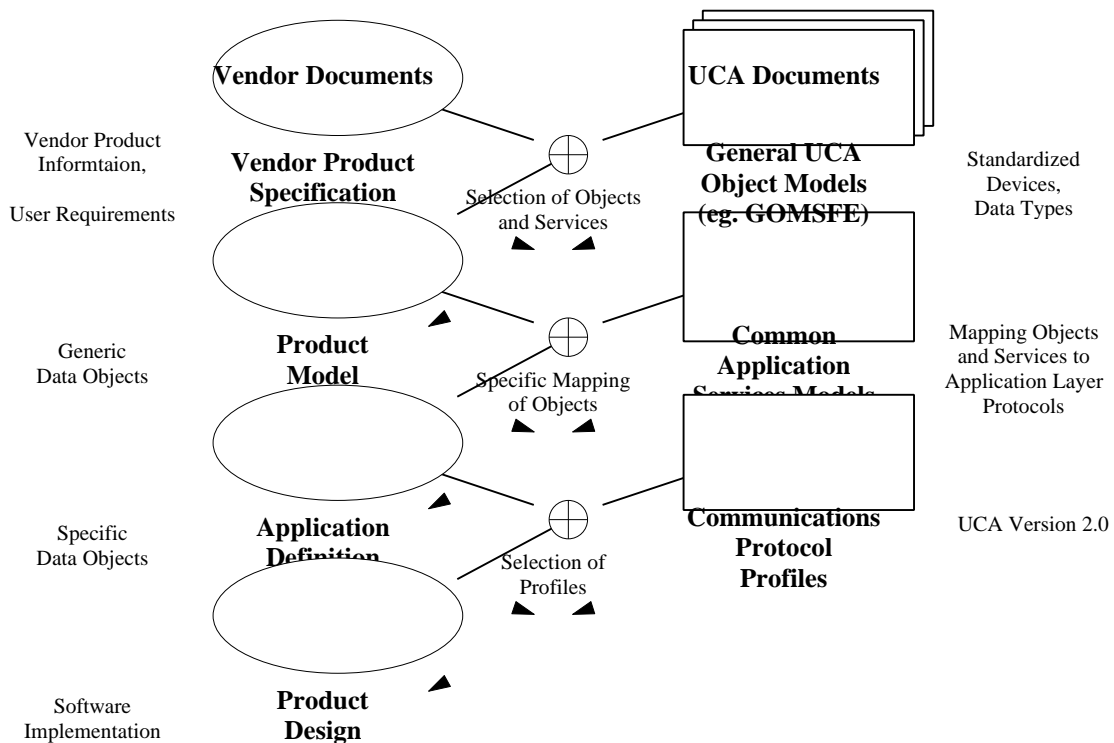


Bild 13 Beziehungen zwischen den UCA Dokumenten

GOMSFE ist prinzipiell unabhängig vom benutzten Kommunikationsstack (ISO/OSI-Schichten eins bis sieben). Da das Datenmodell jedoch auf MMS (Manufacturing Message Specification) als Anwendungsschichtprotokoll abgebildet wird, ist Datenmodell „GOMSFE“ auf MMS angepasst bzw. optimiert worden. D.h., wenn in Zukunft ein anderer Kommunikationsstack bzw. ein anderes Anwendungsschichtprotokoll eingesetzt werden soll, kann die Abbildung von GOMSFE auf das neue Protokoll zu erheblichem Aufwand führen. Im Besonderen dann, wenn sich MMS und das neu eingesetzte Protokoll erheblich unterscheiden.

In dem Dokument CASM werden

- die Kommunikationsdienste definiert, die für einen Austausch der Daten notwendig sind,
- diese definierten Dienste auf die vom Anwendungsschichtprotokoll MMS zur Verfügung gestellten Dienste abgebildet und
- das GOMSFE-Datenmodell auf das vom Anwendungsschichtprotokoll MMS zur Verfügung gestellte Datenmodell abgebildet.

Die Dienste, die in CASM beschrieben sind, sind ebenfalls auf das Anwendungsschichtprotokoll MMS angepasst bzw. optimiert worden. Somit gelten die gleichen Nachteile wie für GOMSFE (s.o.).

Das Datenmodell GOMSFE folgt modernen Methoden der objektorientierten Beschreibung mit den Eigenschaften

- Identifikation: Namen statt nichts aussagende Zahlenfolgen,
- Kapselung: Verbergen der Implementierung,
- Klassen: Typisierung und Aggregieren von Objekten,
- Objekte: Instanzen von Klassen,
- Vererbung: Definition von Objekten auf Basis bereits vorhandener Objekte,
- Hierarchie: Hierarchie der Objekte und Namen.

Jedes Objekt hat Verbindungen mit der Außenwelt und kommuniziert mit anderen Objekten nach bestimmten Regeln, die im GOMSFE Access Model (Datenzugriff) und im CASM Service Model (Dienstmodell) festgelegt sind. Objekte können aggregiert sein und somit ein ganzes Gerät beinhalten. Objekte bestehen aus Daten die funktionell unterteilt sind. Alle Objekte werden prinzipiell unabhängig vom Kommunikationsprotokoll beschrieben. Die Identifikation der Objekte erfolgt nicht punktorientiert (Punkt zu Punkt Verbindung), wie in der Fernwirktechnik üblich, sondern geräteorientiert mit bedeutungsvollen Namen. Dabei haben die Namen eine hierarchische Struktur.

6.2.2 Die Hierarchie

Die Hierarchie besteht aus fünf Ebenen. Die oberste Ebene, im Bild 14 mit Server benannt, stellt die Kommunikationssoftware dar, sie ist die Schnittstelle zur Außenwelt.

Ein physikalisches Gerät, allgemein ein reales Gerät, kann aus mehreren Servern bestehen. Der Server eines physikalischen Gerätes kann mit Hilfe einer netzweit eindeutigen Netzwerkadresse `server@<address>` identifiziert werden. Ein physikalisches Gerät ist ein reelles Gerät, z.B. ein Kombigerät, Feldleitgerät oder ein Schutzgerät und kann aus mehreren logischen Geräten bestehen.

Die zweite Ebene beinhaltet das logische Gerät (LogicalDevice), im Bild 14 beispielhaft R1234 genannt. Unter einem logischen Gerät ist eine Funktion zu verstehen, die aus mehreren Gerätemodellen bestehen kann. Ein Kombigerät besteht z.B. aus den logischen Geräten Feldleitfunktion und Schutzfunktion (AMZ, UMZ, oder Distanz etc.).

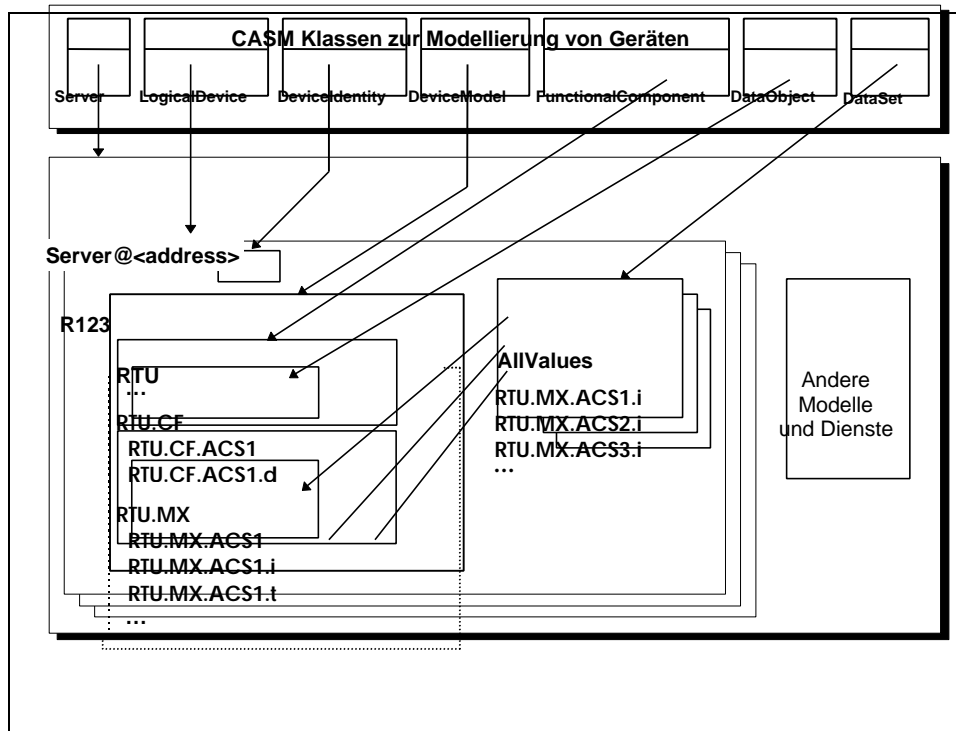


Bild 14 CASM-Klassen und ihre Beziehungen zu den Objekten

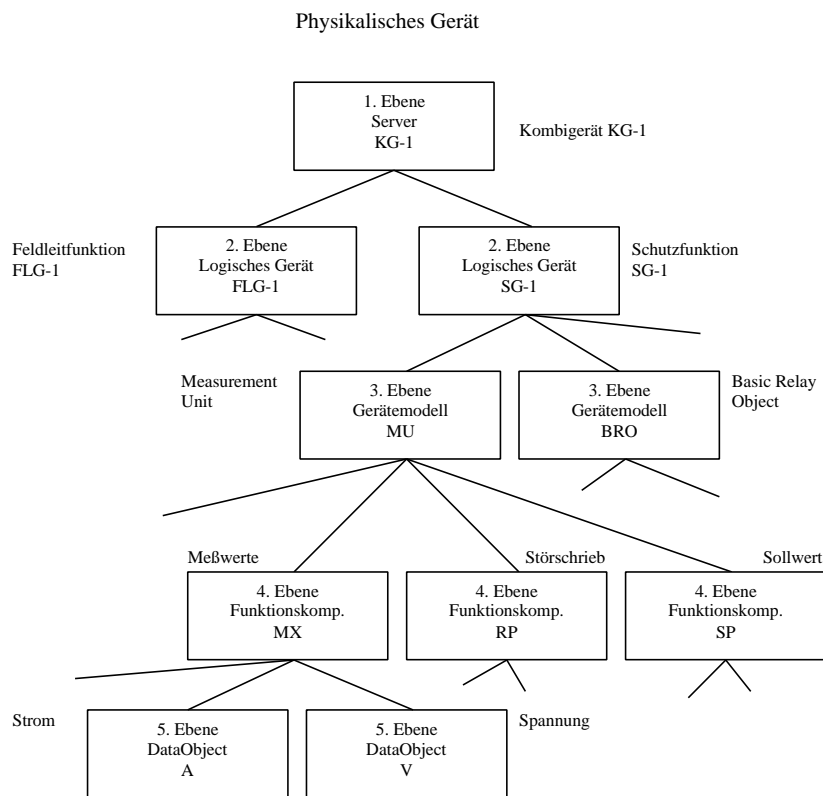


Bild 15 Beispiel einer Hierarchie anhand eines physikalischen Gerätes

Ein logisches Gerät besteht aus einem oder mehreren Gerätemodellen (DeviceModel), im Bild 14 beispielhaft RTU (Remote Terminal Unit) benannt. Für jedes logische Gerät kann eine Geräteidentifikation (DeviceIdentity) hinterlegt werden. Gerätemodelle sind zum Beispiel die Schalter (Leistungsschalter, Trenner,...), Messeinheiten, Längsregelung, Schutz, RTU etc.

Die Gerätemodelle bestehen wiederum aus einer oder mehreren Funktionskomponenten (FunctionalComponent). Funktionskomponenten sind zum Beispiel Status ST (eines Schalters), Sollwerte SP, Messwerte MX, Konfiguration CF, Report RP. An den Funktionskomponenten schließen sich letztendlich die eigentlichen Informationen (DataObject) an. Instanzen von CASM-Klassen heißen Objekte und repräsentieren den Kommunikationsteil eines Gerätes. Anhand eines Kombigerätes, das das Referenzfeld (Bild 4) ansteuern soll, verdeutlicht Bild 15 diese Hierarchie noch einmal.

Die Informationen (DataObject) können zu Datensätzen (DataSet) zusammengefasst werden. Das ist zum Beispiel dann sinnvoll, wenn notwendigen Messwerte für eine Funktion (z.B. Schutz) gelesen werden sollen. Mit Hilfe des DataObjects kann dann auf eine Gruppe von Werten mit nur einem Lesevorgang zugegriffen werden, anstatt jeden Wert einzeln einzulesen. Für die Informationen sind in CASM 24 standardisierte Datentypen hinterlegt. Datentypen sind Definitionen von Informationstypen der Informationen. Unter Informationstypen fallen Ganzzahlen und Festkommazahlen mit und ohne Vorzeichen, Zeichenfolgen, Booleanfolgen etc.

Der Informationszugang erfolgt über eine Adressierung, die sich aus der Hierarchie des Datenmodells zusammensetzt und mit dem logischen Gerät beginnt. Die Adresse

SG-1/MU.MX.A.PhsBi

greift auf die Information „Messwert (MX) eines Stromes (A) in Leiter B (PhsB) als Ganzzahl (i) der Schutzfunktion SG-1 mit dem Gerätemodell MU (Measurement Unit)“ zu. A ist dabei ein DataSet der die Messwerte der folgenden Tabelle enthält.

Tabelle 7 Beispiel eines DataSets mit der Bezeichnung A

Wert in Leiter A: PhsAi	Wert zwischen Leiter A und B: PhsABi	Zeitstempel: t
Wert in Leiter B: PhsBi	Wert zwischen Leiter B und C: PhsBCi	Qualifier: q
Wert in Leiter C: PhsCi	Wert zwischen Leiter A und C: PhsACi	

D.h. die Adresse SG-1/MU.MX.A würde alle Daten des DataSets A entsprechend Tabelle 7 liefern.

Der Zeitstempel gibt die Aktualität der Messwerte an und der Qualifier den Vertrauensbereich der Werte innerhalb des DataSets, d.h. ob z.B. die Werte gültig oder ungültig sind.

Die bedeutungsvolle Benennung der Informationen verringert den Parametrierungsaufwand und erlaubt das hierarchische Lesen von Informationen in Gruppen, entsprechend der Namenshierarchie. Mit SG-1/MU.MX werden z.B. alle Informationen der Funktionskomponente MX identifiziert. Neben dem Namen bei der Eingabe des Datenmodells ist auch das Generieren eines Indexes für die Implementierung möglich.

6.3 Die Modellierung

Für die Modellierung des Referenzfeldes (Bild 4) wurde der Stand vom Mai 1998 verwendet. Im Anhang D: Modellierung GOMSFE ist die Modellierung vollzogen.

Im Folgenden ein Beispiel (Bild 16) für die Modellierung eines Leistungsschalters Q0. Q0 soll geschaltet werden können und der Status der Schalterstellung übermittelt werden können. Der Vorgang bei der Modellierung entspricht dem von IEC 61850. Unter Verwendung von UCA.2 müssen zwei Variablen angelegt werden, eine für die Befehlsrichtung und eine für die Melderichtung. Unter Verwendung von IEC 61850 braucht nur eine Variable angelegt werden. Im DataObject „Pos“ befinden sich die Einträge für die Befehlsrichtung „ctlval“ und der Melderichtung „stval“ (siehe auch Bild 10). In Beiden Fällen wurden die Ebenen Server und Logical Device der Einfachheit halber weggelassen.

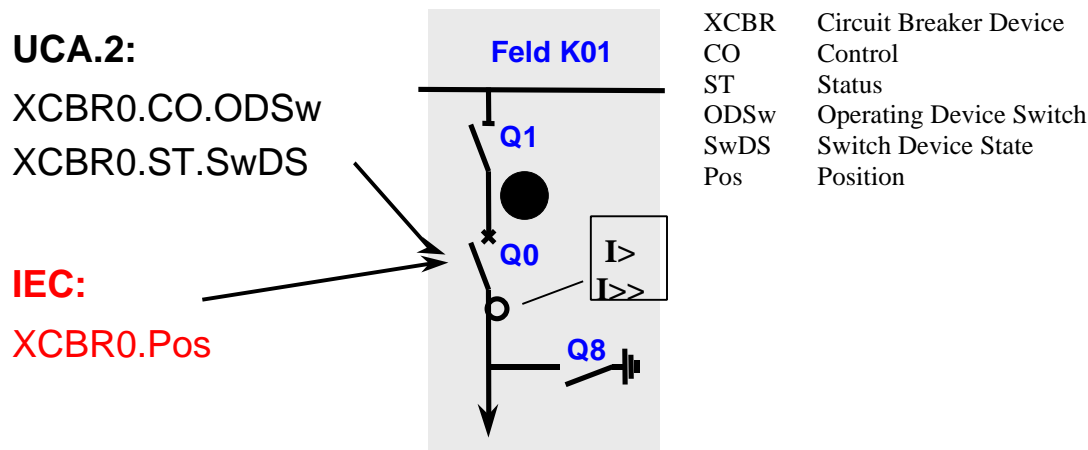


Bild 16 Modellierung eines Leistungsschalters Q0

7 Vergleich und Ergebnisse der Modellierung

Im Rahmen dieses Forschungsprojektes wurden die Entwürfe von IEC und UCA nicht nur in Bezug auf die Anforderungen gemäß Kapitel 16 überprüft. Vielmehr wurde jeweils das gesamte Daten- und Dienstmodell überprüft. Fehler, Unzulänglichkeiten und Erweiterungswünsche sind als Kommentare direkt an die zuständigen Arbeitsgruppen bei IEC weitergegeben worden.

7.1 Vergleich

Die IEC Entwurfspapiere

- IEC 61850-7-3 Common Data Classes
- IEC 61850-7-4 Compatible Logical Node and Data Classes

entsprechen dem UCA Entwurfspapier GOMSFE (Bild 17). Im Gegensatz zu GOMSFE verwendet IEC den funktionsorientierten Ansatz. Das Datenmodell ist, wie auch GOMSFE, unabhängig vom benutzten Anwendungs- und Transportprotokoll.

Die IEC Entwurfspapiere

- IEC 61850-7-2 Abstract Communication Service Interface (ACSI)
- IEC 61850-8-1 SCSM Mapping to MMS

entsprechen dem UCA Entwurfspapier CASM (Bild 17).

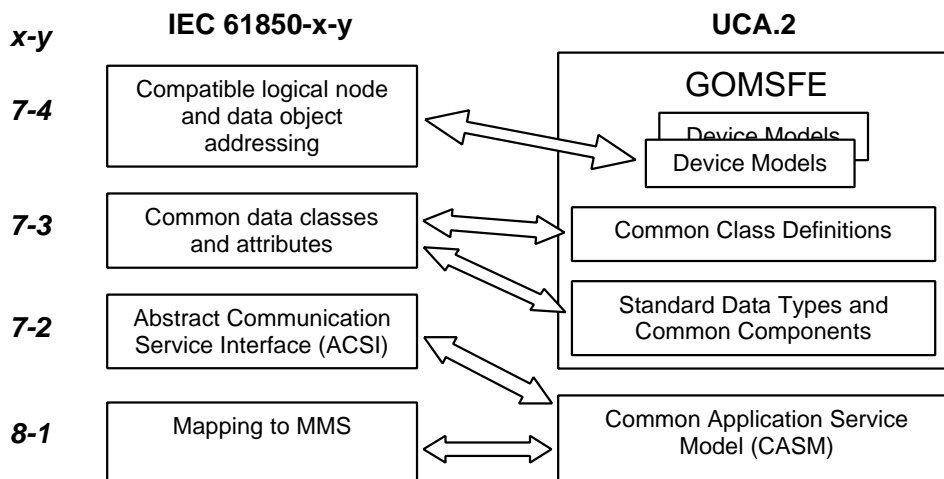


Bild 17 Gegenüberstellung der UCA- und IEC-61850- Dokumente

Im Gegensatz zu CASM sind die Kommunikationsdienste unabhängig vom Anwendungs- und auch Transportprotokoll. Die Abbildung der Anwendung (z.B. Schutz) auf die Anwendungsschicht wird in den Entwürfen 61850-8-X beschrieben. Der Teil IEC 61850-8-1 befasst sich mit der Abbildung auf MMS. Die Modellierung folgt modernen Methoden der objektorientierten Beschreibung entsprechend GOMSFE.

Auf den ersten Blick erscheinen die Hierarchien der Datenmodelle nach IEC und UCA sehr ähnlich, d.h. die grundsätzliche Struktur der Hierarchien ist identisch wie auch die ersten beiden Ebenen Server und LogicalDevice. In der dritten Ebene verfolgt der UCA Entwurf den geräteorientierten Ansatz mit dem Gerätemodell (DeviceModel), der IEC Entwurf dagegen den funktionsorientierten Ansatz mit dem „Logischen Knoten“ (LogicalNode). Unter funktionsorientiert wird nicht die Funktionalität bzw. der Funktionsumfang oder auch der Leistungsumfang z.B. eines Sammelschienenschutzes verstanden, denn dieser soll natürlich nicht genormt werden.

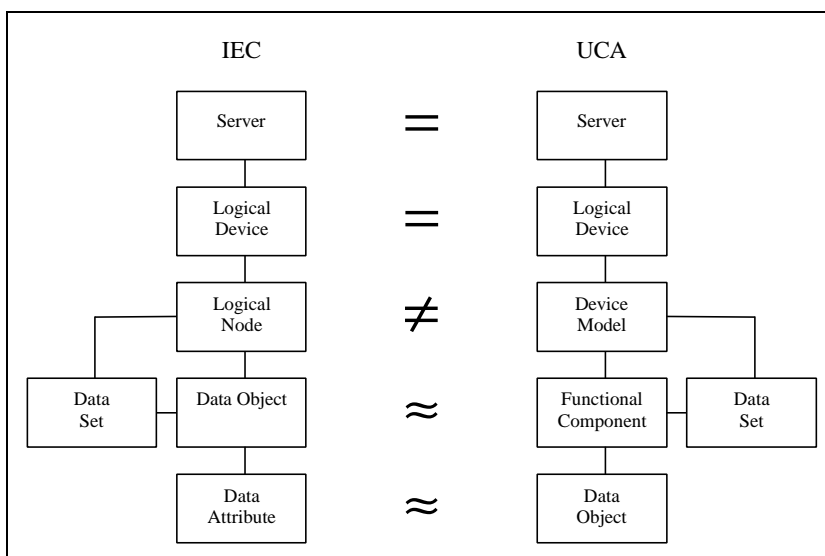


Bild 18 Vergleich der Hierarchien der Datenmodelle nach IEC und UCA

Der UCA Entwurf betrachtet das ganze vorhandene Gerät und ermittelt welche Funktionen bzw. Funktionskomponenten (FunctionalComponents) für dieses Gerät benötigt werden und ordnet diese Funktionen genau diesem Gerät zu. Daher kann eine Funktionskomponente im UCA Entwurf mehrmals vorkommen aber unterschiedliche Inhalte aufweisen. Der IEC Entwurf definiert Funktionen (LogicalNode) und setzt aus diesen die Geräte zusammen, daher hat ein bestimmter logischer Knoten immer den gleichen Inhalt, unabhängig unterhalb welchem logischen Gerät er verwendet wird.

Unterhalb der logischen Knoten wie auch der Funktionskomponenten sind die Datenobjekte (DataObjects) angeordnet, die die eigentlichen Informationen wie z.B. die Messwerte enthalten. Aufgrund der verschiedenen Betrachtungsweisen (geräte- bzw funktionsorientiert) und der unterschiedlichen Anforderungen an die Stationsleittechnik der Europäer und Amerikaner, sind Umfang und Bedeutung der Datenobjekte nicht vollständig identisch.

Der Unterschied in den Modellierungen soll mit Hilfe der folgenden zwei Bilder (Bild 19 und Bild 20) noch einmal deutlich gemacht werden. Dazu wird die Funktion „Automatische Wiedereinschaltung (AWE)“ modelliert. Die Abkürzungen werden nicht erläutert, da dies den Rahmen dieser Ausarbeitung sprengen würde (siehe 9 und 10). Die Bilder sollen die oben beschriebenen Sachverhalte plastisch darstellen.

Die Funktion AWE (Recl) wird mit Hilfe von logischen Knoten (LLN0, MMUX, RRec, GGIO, XCBR) modelliert (Bild 19).

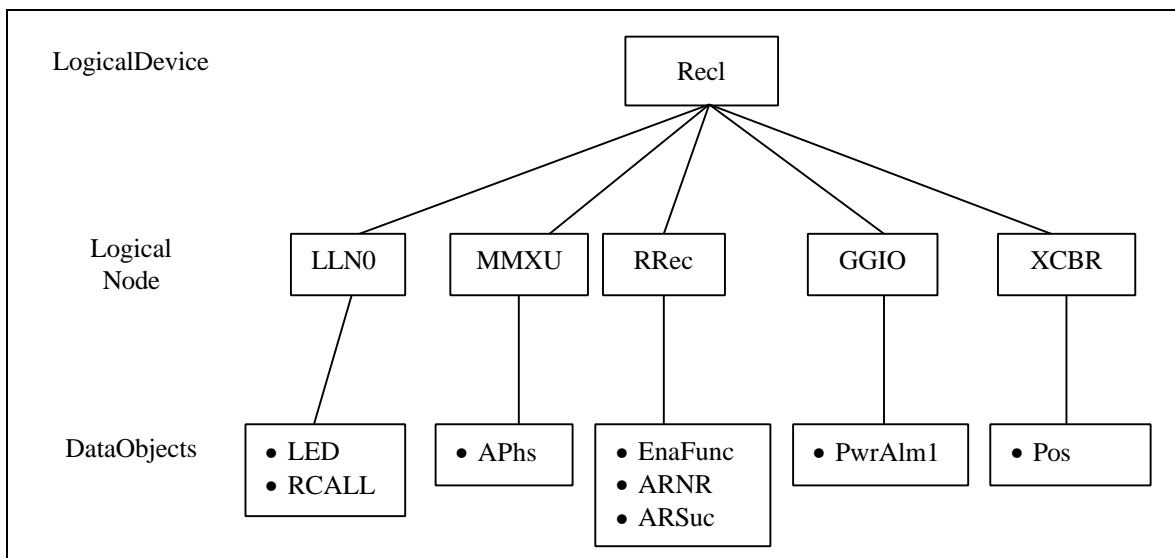


Bild 19 Modellierung einer AWE nach IEC Datenmodell

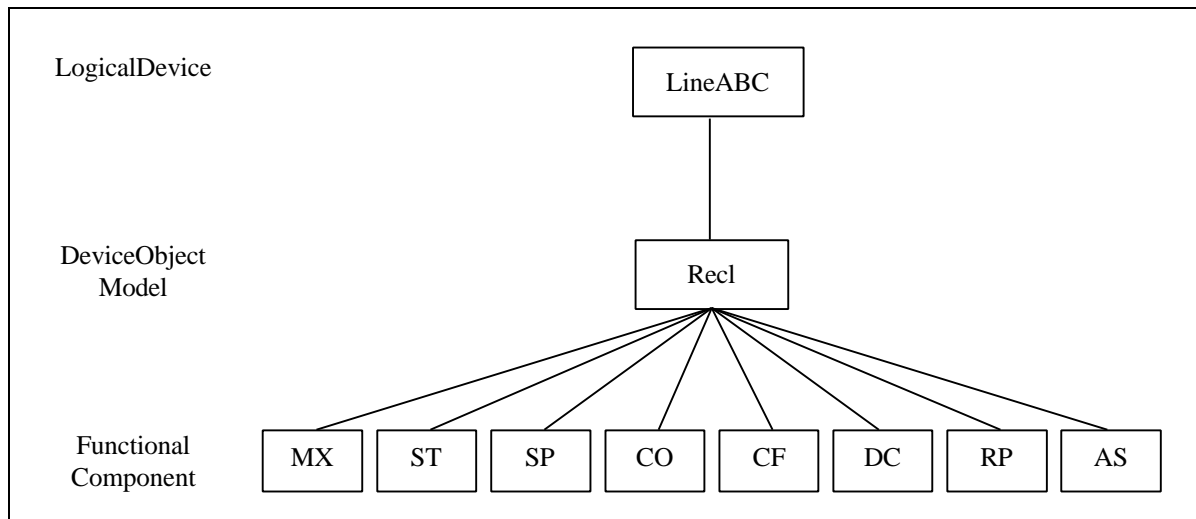


Bild 20 Modellierung einer AWE nach UCA Datenmodell

Das Gerät „Recloser“ wird mit Hilfe von Funktionskomponenten (MX, ST, SP, ...) modelliert, die an die Anforderungen des Gerätes angepasst werden (Bild 20). In Amerika ist eine AWE Funktionalität ein reales Gerät, d.h. ein separates Gerät. In Deutschland ist eine AWE eine reine Funktion, die sich der vorhandenen Mittel (Schutzgerät, Leistungsschalter) bedient, wobei die Funktion in einem Schutzgerät integriert ist. Entsprechend der europäischen Philosophie würde eine AWE im Normalfall nicht als Logisches Gerät modelliert, sondern wäre z.B. Teil eines Logischen Gerätes „UMZ-Schutz“.

IEC 61850 unterscheidet sich von UCA.2 nicht nur in der Namensgebung, sondern auch in der Struktur. In UCA.2 befindet sich z.B. zwischen der dritten (LogicalNode) und der vierten (DataObject) Hierarchieebene eine Ebene – genannt "Functional Components", die die DataObjects gruppiert in Befehls-Objekte, Status-Objekte, Messwert-Objekte usw. Diese Unterschiede zwischen IEC-Objektklassenmodell und UCA.2- Objektklassenmodell werden oft – besonders von den Amerikanern – nicht bedacht. Sie erschweren natürlich die konsistente Einbindung der UCA.2-Modelle in IEC erheblich. Seitens der UCA.2 Vertreter wird das IEC-Objektklassenmodell regelmäßig falsch dargestellt. Besonders der amerikanischen Öffentlichkeit wird das UCA.2-Modell fälschlicherweise als das IEC-Modell verkauft.

7.2 Ergebnisse

Die Funktionsanalyse der kombinierten Feldgeräte hat gezeigt, dass die Anforderungen der EVU, die in den Empfehlungen der VDEW für die digitale Stationsleittechnik festgehalten sind, zum großen Teil abgedeckt werden. Auf der anderen Seite werden aber Funktionalitäten zur Verfügung gestellt, die an den Anforderungen vorbeigehen. Der Vergleich amerikanischer und europäischer Anforderungen an die Stationsleittechnik hat deutliche Unterschiede aufgezeigt. Das liegt zum einem an dem unterschiedlichen Stand der Entwicklung in der digitalen Stationsleittechnik, zum anderen an den unterschiedlichen Topologien der Versorgungsnetze. In Europa kann auf eine über 15 jährige Erfahrung mit der digitalen Technik aufgesetzt werden, für die USA stellt diese Technik einen kompletten Neuanfang dar.

Dementsprechend sind die unterschiedlichen Ansätze für die Objektmodelle und Kommunikationsdienste des UCA Standard Entwurfes und des IEC Normentwurfes nur mit großem Aufwand zu harmonisieren.

Durch den Neuanfang in Bezug auf die digitale Stationsleittechnik der Amerikaner, brauchten sie, im Gegensatz zu Europa, auf keine vorhanden Techniken Rücksicht nehmen. Mit einem pragmatischen Ansatz wurde im UCA Entwurf auf vorhandene standardisierte Bausteine (MMS, Ethernet etc.) zurückgegriffen. Jeder Teil des Entwurfes greift auf neuste Techniken zurück, wie z.B. die objektorientierte Modellierung, moderne Ethernet Technologien oder CORBA (Common Object Request Broker Architecture). Ein Problem des amerikanischen Ansatzes ist die geräteorientierte Modellierung, da jedes neue funktionserweiterte Gerät, das in Zukunft auf dem Markt angeboten wird, neu modelliert und in dem Standard aufgenommen werden muss. Ein weiteres Problem ist die fehlende Erfahrung der Amerikaner auf dem Gebiet der digitalen Stationsleittechnik und dass der UCA Entwurf auf rein theoretischer Basis entwickelt wurde. So gut die angewandten Techniken und Ideen auch sind, so inkonsistent sind die Papiere der Standardisierungsentwürfe. Grund ist der fehlende praktische Bezug, d.h. zum einen die fehlende Erfahrung und zum anderen die damals fehlenden praktischen Testprojekte.

Ein Problem des IEC Entwurfes ist die gewünschte Abwärtskompatibilität zu vorhandenen IEC Standards wie z.B. IEC 60870-5-103 (Anbindung der Schutzgeräte an einen Stationsbus) oder -101 (Fernwirkprotokolle). Ein anderes Problem ist das Bestreben der Hersteller ihre proprietären Bussysteme für den Stations- wie auch den Prozessbus in den Standard einbringen zu können. Denn je mehr verschiedene Bussysteme in dem Standard aufgenommen werden, um so weniger ist die gewünschte Interoperabilität realisierbar. Weiterhin wurde in Europa die Weiterentwicklung der Standards für die digitale Stationsleittechnik versäumt und erst durch die massiven Bestrebungen der Amerikaner, einen eigenen nationalen IEEE/ANSI Standard zu definieren, zu neuem Leben erweckt.

Der UCA.2 Entwurf ist noch weit entfernt davon einen stabilen und realisierbaren Standard darstellen zu können, und durch die Harmonisierung von UCA.2 und IEC 61850 wird der Zeitpunkt einer vollendeten Normenreihe IEC 61850 weiter in die Zukunft verschoben. Dementsprechend sind die Modellierungen des Referenzfeldes, die in den Anhängen vorgestellt sind, zum Teil unvollständig und zum Teil auch inkonsistent. Die Erfahrungen, die im Zuge dieser Modellierungen gemacht wurden sind komplett in die zuständigen Normungsgremien eingegangen und in die IEC Entwurfspapiere aufgenommen worden. Es wurden ca. 300 Kommentare erfolgreich eingebracht.

Einige Mängel in den Entwürfen wurden aufgedeckt und entsprechende Verbesserungen vorgeschlagen. Besonders mit dem amerikanischen Ansatz UCA.2 ließen sich einige der für den deutschen Markt üblichen Informationssequenzen nicht modellieren. Als Beispiel sei die Steuerung eines Schaltgerätes genannt (siehe auch Bild 12). Im Falle, dass der Schalter sich nach einem Befehl nicht bewegt, wird in gebräuchlichen Leitsystemen vom überwachenden Feldleitgerät eine entsprechende Fehlermeldung gesendet. Hierauf wird der Schaltwärter bei der UCA-Variante allerdings vergeblich warten.

Mitte 1999 hatte die Beschreibung der Datenobjekte und Dienste von IEC 61850 einen Status erreicht, nach dem sich der o.g. Beispielprozess vollständig und eindeutig modellieren ließ. Ein Problem von IEC 61850 ist die Gewährleistung der Interoperabilität zwischen Geräten verschiedener Hersteller. Die gewünschte Interoperabilität wird durch die folgenden Tatsachen beschränkt:

- Es gibt verschiedene Möglichkeiten mit IEC 61850 ein und dieselbe Funktion zu modellieren.
- Es gibt viele verschiedene Möglichkeiten die Informationen innerhalb einer Station zu strukturieren.
- Es gib verschiedene Vorstellungen davon, was in einer Schaltanlage über einen Bus kommuniziert werden soll.

Tabelle 8 Alternative Strukturierung der Daten für den Überstromzeitschutz

<i>Teilfunktion Information</i>	<i>Logical Node Data Object</i>	<i>Common Data Class</i>
Überstromzeitschutz Stufe 1 (I >>)	PTOC1 (<i>Time Overcurrent</i>)	
Anregung	activated	SIG
Auslösung	operated	SIG
Überstromzeitschutz Stufe 2 (I >)	PTOC2 (<i>Time Overcurrent</i>)	
Anregung	activated	SIG
Auslösung	operated	SIG

Um dies zu verdeutlichen, wird die Datenstruktur des Überstromzeitschutzes näher betrachtet. In Tabelle 3 ist er modelliert durch einen Logical Node "Überstromzeitschutz" (PTOC). Die sechs zugehörigen Data Objects für Anrege- und Auslösemeldungen sind alle von der Klasse Single Point Status (SPS) – die in Tabelle 6 beschrieben ist. Tabelle 8 und Tabelle 9 zeigen eine andere Möglichkeit, wie genau die gleichen Informationen strukturiert werden könnten. Hier wurde für beide Stufen des UMZ-Schutzes je ein separater Logical Node gewählt. Jedem Logical Node sind nur noch zwei Data Objects zugeordnet – eines für Anregung und eines für Auslösung. Ein solches Data Object enthält jedoch nicht nur die Informationen "kommen" oder "gehen", sondern gleichzeitig auch Information über die beteiligten Leiter. Denn das Data Attribute "Group Value" in Tabelle 9 ist vom Typ BIT STRING und hält für jeden Leiter ein separates Bit zur Informationsspeicherung bereit.

Tabelle 9 Detailinformationen der Common Data Class "Status Indication Group"

Detailinformation Wertebereich	Data Attribute D.A. Content
Beteiligte Phasen L1 / L2 / L3 / N	grpVal (group value) BIT STRING { phase-A (1), phase-B (2), phase-C (3), neutral (4), ... }
Qualität	q (quality) ...
Zeitstempel	t (timestamp) ...
Status Indication Group - SIG	

Dieses Beispiel zeigt, dass die selben Informationen in zwei völlig unterschiedliche Datenstrukturen gegliedert sein können. Prinzipiell könnte die Normenreihe IEC 61850 vorschreiben, welche Variante angewendet werden darf. Aber leider konnten sich die Mitglieder in den zuständigen Arbeitsgruppen bei IEC nicht einigen.. Der Grund dafür liegt darin, dass jeder der Mitglieder der Arbeitsgruppen eine bestimmte Vorstellung für die Umsetzung der Datenstruktur in eine bestimmte Kommunikationssoftware hat. Bezüglich der Datenstrukturen sind bekanntermaßen zur Zeit viele unterschiedliche Lösungen auf dem Markt. Die Darstellung in den Tabellen in Abschnitt 5.3 ist stark an den Kommunikationsnorm IEC 60870-5-103 angelehnt, der die Kommunikation von Schutzgeräten beschreibt. Da diese Norm besonders in Europa weit verbreitet ist, wundert es nicht, dass diese Darstellung besonders von europäischen Herstellern bevorzugt wird. Eine Darstellung der Detailinformationen in Bit-Strings kann das notwendige Datenaufkommen auf dem Bus reduzieren. Bei Verwendung eines Kommunikationsstacks mit MMS als Anwendungsschichtprotokoll würde eine Übertragung von einzelnen Daten des Typs BOOLEAN zu einem größerem Datenverkehr führen, als wenn die einzelnen Informationen in einem Datum vom Typ BIT STRING zusammengefasst sind. Die Darstellung in Tabelle 8 und Tabelle 9 zeigt den aktuellen Normungsvorschlag der Amerikaner für IEC 61850. Wie bereits erwähnt, favorisieren die Amerikaner MMS als Anwendungsschichtprotokoll und daher ist ihr Daten- und Dienstmodell dementsprechend auf MMS optimiert. Natürlich gibt es auch das eine oder andere technische Argument für die eine oder andere Variante, aber Marktinteressen spielen hier eine stärkere Rolle. Abhilfe wird eine genormte Beschreibung schaffen, mit der die Adressierung der Daten des Datenmodells für ein Gerät festgelegt werden kann. Als Beschreibungssprache wird XML eingesetzt werden. Die Semantik der Beschreibung wird im Teil 6 der IEC-Norm aufgeführt sein (IEC 61850-6).

Diese Teilaufgabe ist entsprechend dem AiF-Antrag vollständig und erfolgreich bearbeitet worden.

8 Simulation

8.1 Aufgabe

Aufbauend auf den Modellierungen aus der ersten Teilaufgabe und Ethernet als Stationsbus sollte die Kommunikation und damit die Tauglichkeit von Ethernet als Stationsbus untersucht werden. Dazu wurden zunächst zwei PCs über ein Ethernet Netzwerk verbunden, wobei ein PC, in Bild 21 ist es der PC 1, die Nahsteuerung simuliert (Stationsleitgerät) und die anderen PCs die Feldgeräte simulieren. Jedes Feldgerät soll die oben beschriebenen Schutz- und Steuerungsfunktionen enthalten (Kombigerät).

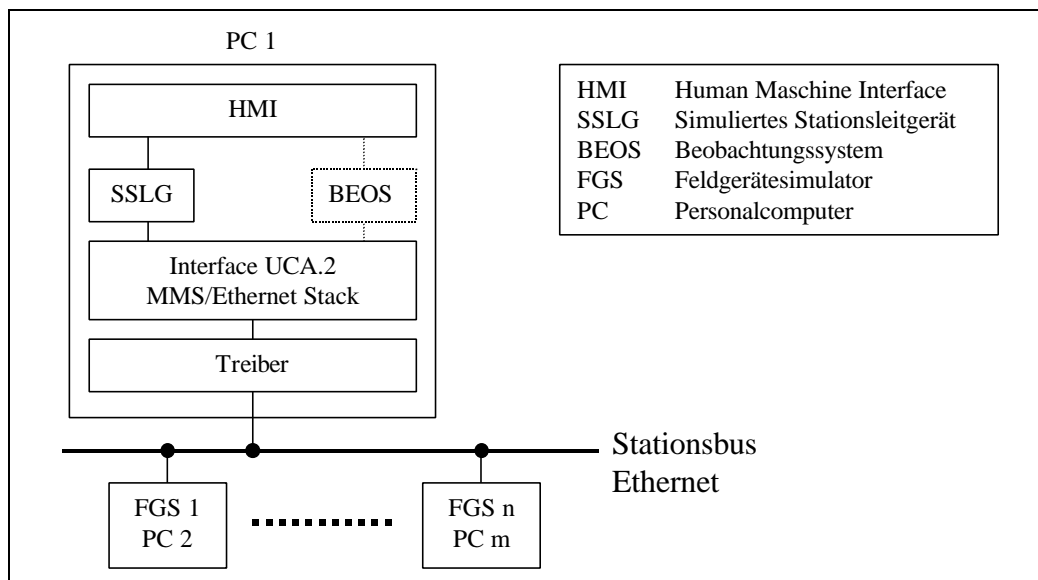


Bild 21 Simulationsaufbau

Für die Simulation des Stationsleitgerätes und des Feldgerätes wurden jeweils verschiedene Softwarepakete verwendet und weiterentwickelt die auf den Standardentwürfen von UCA.2 aufsetzen. Mit den IEC-Standardentwürfen war dies nicht möglich, da bisher keine entsprechende Software erhältlich ist. Das spielte für die Untersuchung der Tauglichkeit von Ethernet jedoch keine Rolle. Für den Client (Stationsleitgerät) wurde die Software AX4MMS sowie der MMS-ObjectExplorer von SISCO verwendet. Für die Server (Feldgeräte) die Software MMSd von Tamarack. Für die Untersuchung wurden die üblichen Kommunikationsdienste verwendet. Kommunikationsdienste sind zum Beispiel Verbindungsdienste, Variablenzugriffsdienste, Gerätesteuerungsdienste, unbestätigter Datentransfer, Filetransfer, etc.

8.2 Ergebnis

Grundsätzlich ist Ethernet als Stationsbus in allen Stationen aller Spannungsebenen tauglich. Auch die Anforderung, dass ein Stationsbus deterministisch sein soll, also ein bestimmtes Zeitverhalten hat, kann durch den Einsatz bestimmter Ethernet-Komponenten (sog. Switches)

erreicht werden. Zeit- sowie Performance- Probleme sind nicht zu erwarten, da die Ethernet-Technologie ausreichende Bandbreiten zur Verfügung stellt. Die Frage des wirtschaftlichen Einsatzes dieser Technologie wurde nicht untersucht. Jedoch bieten schon heute immer mehr Hersteller entsprechende Komponenten für die Stationsleittechnik an, so daß in absehbarer Zeit der Einsatz der Ethernet-Technologie vielleicht sogar zeitgleich mit dem Erscheinen des IEC 61850-Standards wirtschaftlich werden kann.

Ein weiteres Ergebnis ist, dass die Austauschbarkeit der Geräte der Sekundärtechnik nicht automatisch durch den zugekauften bzw. verwendeten Kommunikationsstack sichergestellt wird. Um die gewünschte Austauschbarkeit zu erreichen sind ergänzende Festlegungen in der Norm erforderlich. Darunter fallen Angaben über das Zeitverhalten und Angaben über eine einheitliche Adressierung der Informationsobjekte.

Diese Teilaufgabe ist entsprechend dem AiF-Antrag vollständig und erfolgreich bearbeitet worden.

9 Pilotsysteme mit Prototypen

9.1 Der Aufbau

Für diese Teilaufgabe, der Hauptaufgabe des Projektes, hatten sich die Hersteller ABB, Siemens und ALSTOM bereit erklärt, kombinierte Feldleit- und Schützgeräte auf Basis der zukünftigen IEC 61850 Norm zu als Prototypen zu entwickeln und für das Projekt zur Verfügung zu stellen.

Die Aufgabe teilt sich in zwei Bereiche, in denen bei der FGH zwei Systeme aufgestellt und untersucht wurden (Bild 22). Im ersten System wurde Profibus/FMS als Stationsbus verwendet und die Kombigeräte aller drei Hersteller sowie ein Stationsleitgerät von Siemens daran angeschlossen. Im zweiten System wurde Ethernet/MMS als Stationsbus verwendet und die Kombigeräte aller drei Hersteller sowie ein Stationsleitgerät von ABB daran angeschlossen. Das Ziel dieser Aufgabe waren pro System die Interoperabilitätstests zwischen den Kombi- und Stationsleitgeräten der Hersteller.

Für das Projekt kamen folgende markteingeführte kombinierte Feldleit- und Schutzgeräte aus neuen Gerätegenerationen zum Einsatz:

- REF542 von ABB
- PS982 von ALSTOM
- SIPROTEC[®] 4 7SJ63 von Siemens.

Neu entwickelt für das Projekt wurden von den Herstellern jeweils Prototypen für die Kommunikationsanschlungen für Profibus/FMS und Ethernet/MMS.

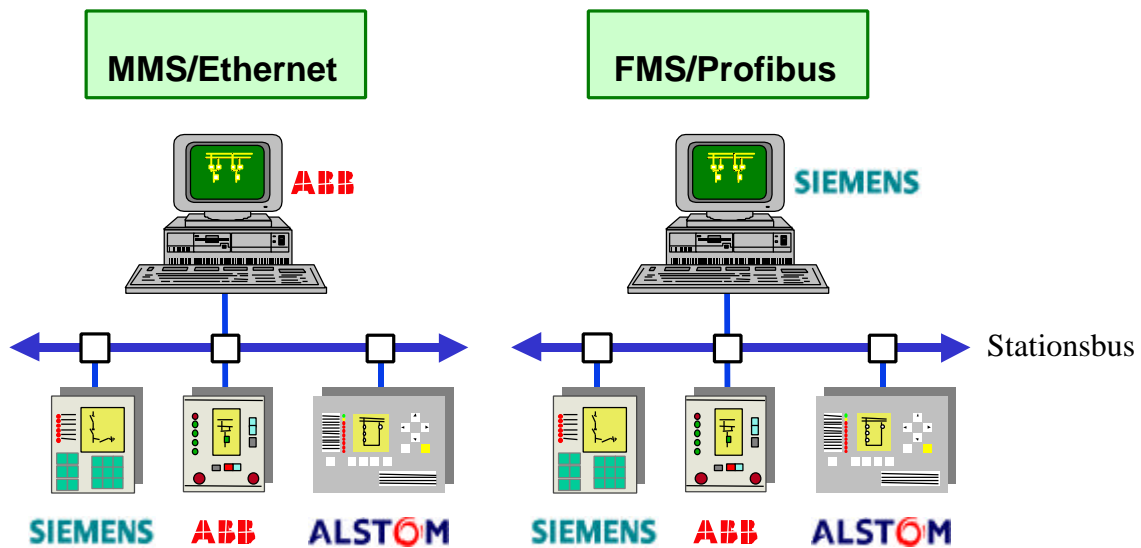


Bild 22 Pilotssysteme

9.2 FMS/Profibus

Beim Profibus-System kam als Stationsleitgerät SICAM[®] SC von Siemens zum Einsatz. SICAM[®] SC ist die zentrale Komponente des auf dem Industriestandard SIMATIC[®] basierenden modularen Automatisierungssystem für Schaltanlagen SICAM SAS. SICAM[®] WinCC, die Komponente für das Bedienen und Beobachten auf Stationsebene, ist ein vollgrafisches System, das auf Standard-PCs unter Windows 95 bzw. NT lauffähig ist.

Alle Geräte kommunizieren mit der Stationsleitebene über Profibus/FMS, dessen Applikationsschicht entsprechend IEC 61850-7 ausgelegt ist. Damit gelang ein Zusammenlegen der Anforderungen der Stationsautomatisierungstechnik mit den Kommunikationsnetzen der Industrieautomatisierung.

Ein Kommunikationsnetz stellt einer Anwendung in einem Gerät eine Anzahl von Diensten für die Übertragung von Informationen zu einem anderen Gerät zur Verfügung. Dieses Kommunikationsnetz kann für die Übertragung von (beinahe) beliebigen Informationen verwendet werden [16]. Mit anderen Worten: Die an der Kommunikation beteiligten Geräte müssen gegenseitig wissen, welche Informationen ausgetauscht werden, in welchen Datenformaten und -modellen und mit welchen Objektadressen dies geschieht und welche Dienste dabei verwendet werden. Diese Informationen ergeben zusammengefasst das Kommunikationsprofil. Für Profibus lag es als Vorschlag für den Teil IEC 61850-8 vor ist jedoch abgelehnt worden. D.h. Profibus/FMS wird nicht Teil der zukünftigen Normenreihe IEC 61850 sein.

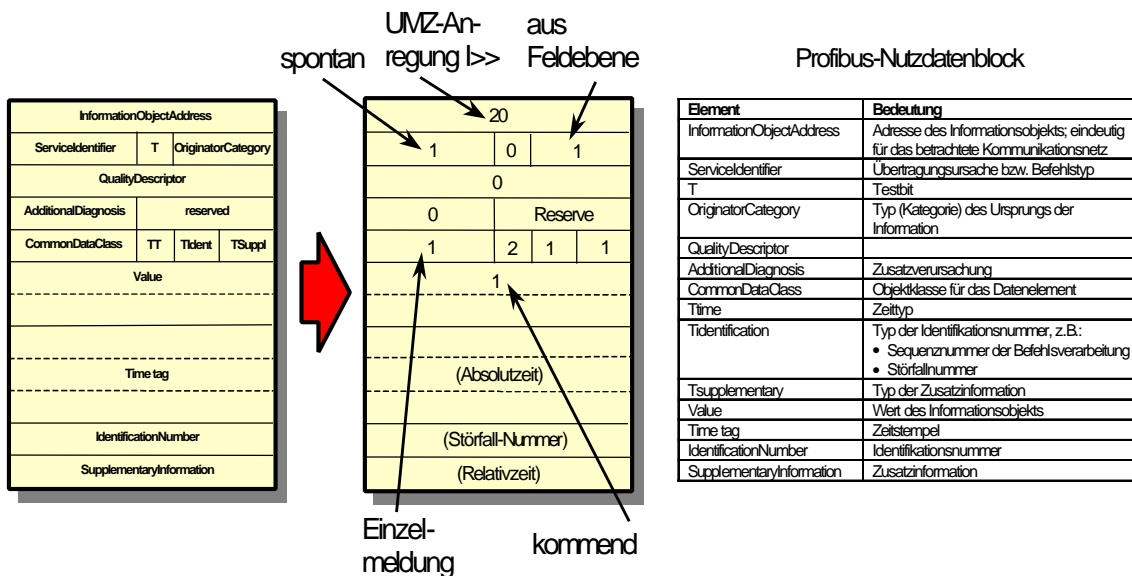


Bild 23 Bearbeitung einer spontanen Meldung bei Profibus

Als Beispiel wird betrachtet, wie die Teilinformationen des IEC-Objektmodells in Nachrichten verpackt werden. Bei Profibus wurde für die Übertragung von Messwerten, Meldungen und Befehlen ein gemeinsamer Nutzdatenblock definiert. Er enthält die in Bild 23 gezeigten Elemente.

Allgemein besteht eine Information in der Schaltanlagenleittechnik nicht nur aus ihrem Wert, sondern sie besitzt auch noch Zusatzinformationen, die alle für die Übertragung berücksichtigt werden müssen. Der Nutzdatenblock beim Profibus besitzt, unabhängig davon, ob eine Meldung, ein Messwert oder ein Befehl übertragen wird, die gleiche Länge, auch wenn für einzelne Telegramme ggf. nicht alle Elemente benötigt werden. Die Elemente des Nutzdatenblocks werden in Form von Nummern codiert. Dies hat den Vorteil, dass gleiche Informationen im Telegramm immer an der gleichen Stelle stehen. Damit sind Analysen des Telegrammstroms, z.B. zu Inbetriebsetzungs- oder Servicezwecken, auf einfache Art und Weise möglich.

Meldungen werden bei Profibus spontan übertragen. Ein Beispiel für die Meldung "UMZ-Anregung I>> kommend" ist in Bild 23 gezeigt. Aus den Werten der Felder im Profibus-Telegramm ist ersichtlich, dass es sich um eine Einzelmeldung handelt, die aus der Feldebene kommt und den Wert "kommend" hat.

Die bisherigen Erfahrungen mit der Realisierung der Profibus-Schnittstelle haben gezeigt, dass die Abbildung der IEC 61850 auf Profibus auf einfache Art und Weise möglich ist. Als tragfähig erwies sich der in IEC 61850 gewählte Ansatz, die Definition der Dateninhalte dicht an den schon existierenden Normen IEC 60870-5-101 und -103 zu orientieren. Damit kann man die Verbindung zwischen Kommunikationsfunktion und Anwendungsprogramm im Kombigerät ohne aufwendige Gatewayfunktionalität herstellen. Hierbei spielt ferner eine Rolle, dass auch die funktionellen Abläufe an der Profibus-Kommunikationsschnittstelle ähnlich zu den beiden zitierten Normen gewählt wurden. Wenn darüber hinaus die

Kommunikationsanschaltung auch physisch auf einer eigenen Karte realisiert ist, kann Kommunikation und Applikation voneinander getrennt, unabhängig voneinander gestaltet und damit die Flexibilität erhöht werden.

Aufgrund der Performance von Profibus/FMS kann es in kleineren und mittelgroßen Schaltanlagen bei EVU und Industrie eingesetzt werden.

9.3 MMS/Ethernet

Beim Ethernet-System kam als Stationsleitgerät μ SCADA von ABB zum Einsatz. μ SCADA ist ein vollgrafisches Bedien- und Beobachtungssystem für Schaltanlagen, das auf Standard-PCs unter Windows NT lauffähig ist. Alle Geräte (Bild 22) kommunizieren mit der Stationsebene über Ethernet und MMS (Manufacturing Message Spezifikation) als übergeordnete Applikationsschicht. Die Basis für eine Interoperabilität der Geräte bildet die im Standardentwurf IEC 61850 Teil 8-1 definierte Abbildung der Objekte und Dienste auf Ethernet/MMS. Ethernet ist die Bezeichnung für eine weit verbreitete, standardisierte Kommunikationsinfrastruktur mit verschiedenen Kommunikationsmedien und Übertragungsraten.

Es wurde TCP/IP als Transportprotokoll verwendet, da die breite Akzeptanz und Verbreitung es möglich macht, auf Standardkomponenten zuzugreifen. Auch MMS ist eine kommerziell verfügbare Standardimplementierung, die für die Anforderungen dieses Projektes entsprechend den IEC 61850-Abbildungsvorschriften angepasst wurde. MMS ist international genormt und berücksichtigt alle kommunikationstechnischen Anforderungen verteilter Automatisierungssysteme.

Alle schaltanlagenpezifischen Objekte und Dienste, die im Gerät entsprechend den Festlegungen von IEC 61850 modelliert sind, werden auf allgemeingültige Basisdienste und Objekte von MMS abgebildet (Bild 24). Das MMS- und IEC 61850-Modell besitzen Gemeinsamkeiten. Beide Standards beschreiben eine konkrete Anwendung abstrakt, um gegenüber Erweiterungen, die durch den technologischen Fortschritt notwendig werden, offen zu sein. So werden z.B. Objekte und Attribute von IEC 61850 auf MMS-Variablen abgebildet. Eine zweistufige Schaltsequenz lässt sich somit als eine Folge von Schreibzugriffen auf hierarchisch strukturierte MMS-Objekte abbilden. Die neue Schalterstellung mit allen definierten Attributen (z.B. Zeitstempel, Qualitätsattribute, ...) und die Beendigung der Schaltsequenz werden als spontane Meldungen zur Stationsebene übertragen.

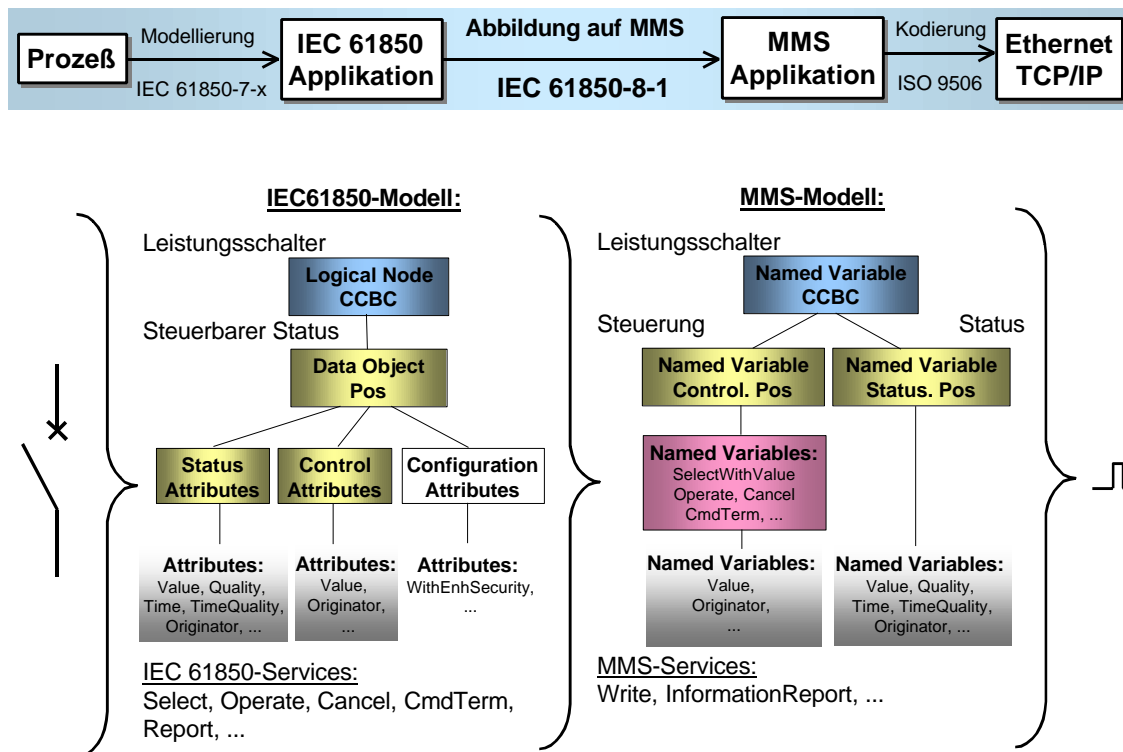


Bild 24 Abbildung des IEC 61850-Modells auf Basisobjekte und Dienste von MMS

Für die Umsetzung des IEC 61850-Modells wird nicht der gesamte Funktionsumfang von MMS benötigt. Nur die relevanten Abbildungen auf MMS-Objekte und Dienste werden in IEC 61850-8-1 zusammengefasst. Weil alle Adressen in der IEC 61850-konformen Namengebung als ASCII-Texte über den Bus übertragen werden, können Standardanalysewerkzeuge zum Beobachten des Datenflusses auf der MMS-Ebene eingesetzt werden. Alle Objektadressen sind systemweit ohne aufwendiges Engineering in ihrer Bedeutung erkennbar.

Die bisherigen Erfahrungen mit der Realisierung der IEC 61850-Ankopplung basierend auf Ethernet/MMS zeigten, dass ein normungsbegleitendes Projekt maßgeblich und positiv die Qualität des Standardentwurfs beeinflusst. Es zeigt sich, dass der IEC 61850-Ansatz - Entkopplung der Applikation von der Kommunikation - tragfähig und zukunftsorientiert ist.

Mit dem Einsatz von Ethernet/MMS können alle Möglichkeiten des neuen Standards IEC 61850 genutzt werden. Die Übertragung der Geräteselbstbeschreibung für das Online-Engineering eines Stationsgerätes ist nur eine der Optionen, die zur Verfügung stehen. Ein großer Vorteil von Ethernet/MMS ist die hohe Skalierbarkeit. Der Anwenderwunsch nach steigenden Übertragungsraten wurde beim Ethernet-Standard berücksichtigt (Fast- und Gigabit- Ethernet). Zunehmend können bewährte und kostengünstige Standardkomponenten aus dem Industriebereich und der Bürokommunikation eingesetzt werden. Ethernet mit seinen übergeordneten Protokollen verbindet die Firmennetze mit der Automatisierung und ermöglicht durch globale Vernetzungsmöglichkeiten den Aufbau von Ferndiagnose- und Fernwartungs- Systemen.

9.4 Ergebnisse

Aufgrund der Trennung der Applikation von der Kommunikation ist IEC 61850 unabhängig vom verwendeten Kommunikationsstack. Das konnte mit Hilfe der Pilotsysteme bewiesen werden. Für die gewünschte Interoperabilität jedoch sind ergänzende Festlegungen in der zukünftigen Normenreihe IEC 61850 notwendig. Das bezieht sich im besonderen auf das Zeitverhalten und notwendigen Regeln für eine identische Adressierung der auszutauschenden Informationen. Unter dem Zeitverhalten werden Zeitangaben verstanden, die definieren wie lange eine Information von der Quelle bis zur Senke benötigen darf oder wie oft und in welchem Intervall Informationen wiederholt werden müssen.

Diese Teilaufgabe ist entsprechend dem AiF-Antrag vollständig und erfolgreich bearbeitet worden.

10 Testphase

10.1 Anforderungen

Die Aufgabe von Tests ist die Überprüfung, ob ein Gerät oder System einer Spezifikation oder einer Norm entspricht. Allgemein ausgedrückt sollen mit Tests Fehler aufgedeckt werden. Test bezeichnet nicht nur die Aktivität des Testens, sondern auch die Planung, Durchführung und das Auswerten des Testens (Testmanagement). Die Notwendigkeit der Durchführung von Tests ergibt sich

- durch nicht ausreichend genau spezifizierte Normen,
- aus möglichen unterschiedlichen Interpretationen der Norm der verschiedenen Hersteller,
- durch die unterschiedliche Nutzung von optionalen Definitionen,
- aus möglichen Implementierungsfehlern.

Die Erfahrungen bei der Norm IEC 60870-5-103 zeigen, dass selten ein Gerät auf antrieb normkonform war. Daher ist eine Teststrategie notwendig, die die Testverfahren Protokoll-, Negativ- und Interoperabilitätstests beinhalten sollte. Es ist sinnvoll die Tests zu einem frühen Zeitpunkt, schon während der Produktentwicklung, durchführen zu lassen. Dadurch können bei festgestellter Nichtübereinstimmung mit der Norm die erforderlichen Maßnahmen und Auswirkungen optimiert werden. Die Ziele, bzw. die Erwartungen an Tests, sowohl von Herstellern als auch von Käufern, sind:

- Erhöhung der Sicherheit vor Überraschungen
- Reduzierung des Risikos der Nichteinhaltung von Terminen
- Reduzierung der Kosten
- Vergrößerung der Produktqualität

Spezielle Ziele der Hersteller sind:

- Reduzierung der Entwicklungs- und Projektkosten
- Erhöhung der Akzeptanz beim Kunden
- Erhöhung der Absatzchancen

Bezüglich der Komponenten eines Sekundärsystems beschränkt sich der Test auf die Überprüfung der Konformität zu einer entsprechenden Norm, die für die Kommunikation zwischen Komponenten der Sekundärtechnik und zwischen den Komponenten der Sekundär- und Primärtechnik erstellt worden ist. Als Beispiel dienen die Normen

- IEC 60870-5-103
- IEC 60870-5-101
- IEC 60870-5-104
- IEC 60870-6 TASE.2
- IEC 61850

Die Fehlerbehebung und -lokalisierung gehört begrifflich nicht zur Tätigkeit des Testens.

10.2 Testverfahren und Ergebnis

10.2.1 Konformitätstest

Bei der Durchführung von Konformitätstests wird dem getesteten System eine Konformität/Nichtkonformität, d.h. eine Übereinstimmung oder Nichtübereinstimmung, zu der zugrundeliegenden Norm (genauer einem Implementierungsprofil dieser Norm) bestätigt. Diese Konformitätsbescheinigung impliziert nicht, dass es mit einem anderen normkonformen System problemlos Daten austauschen kann. Gründe hierfür können sein:

- unterschiedliche Nutzung von optionalen Definitionen der Norm,
- nicht ausreichend genau spezifizierte Norm,
- die Möglichkeit, eine Norm unterschiedlich interpretieren zu können.

Die Art des Nachweises, dass ein Produkt normkonform ist, und die Inhalte eines Konformitätstests werden in einer Testspezifikation festgelegt. Für die zukünftige Normenreihe IEC 61850 wird dies im Teil 10 "Conformance Tests" festgelegt werden. Inhalt eines Konformitätstests (Bild 25) kann ein Funktionstest, ein Protokolltest, ein Negativtest, ein Interoperabilitätstest, eine Abnahmetest oder eine Kombination dieser Testmethoden sein.

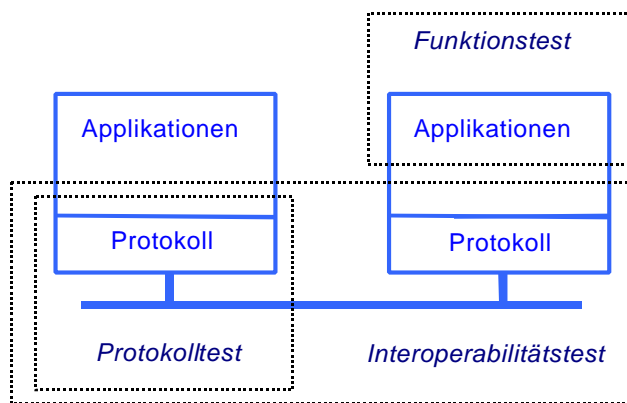


Bild 25 Beispiel eines Konformitätstests

10.2.2 Protokolltest

Ein Protokolltest gliedert sich in zwei Testaufgaben. Zum einen muss überprüft werden, ob die übertragenen Daten entsprechend einer Norm aufgebaut sind. Da die Daten mit Hilfe von Diensten übertragen werden, muss zum andern überprüft werden, ob die Daten mit den richtigen Diensten übertragen worden sind.

Eine sinnvolle Erweiterung des Protokolltests ist der Negativtest. Die Negativtests überprüfen, ob ein Gerät auch beim

- Empfang fehlerhaften Daten oder
- Empfang richtiger Daten die mit falschen Diensten übertragen worden sind

korrekt funktioniert. Dazu werden die Daten innerhalb einer PDU (Process Daten Einheit) manipuliert und versendet.

Um einen Protokoll- oder Negativtest ausführen zu können, muss ein simulierter Prozess die Geräte der Sekundärtechnik zur Kommunikation anregen. Mit Hilfe eines Analysers (siehe Abschnitt 10.5.1) kann dann das Protokoll überprüft werden.

Um die Konformität zur Norm IEC 60870-5-103 "Companion Standard for the Informative Interface of Protection Equipment" oder IEC 60870-5-101 "Companion Standard for Basic Telecontrol Tasks" nachzuweisen, ist ein Protokolltest ausreichend.

10.2.3 Interoperabilitätstest

Interoperabilität ist die Fähigkeit von zwei oder mehr Geräten desselben oder verschiedener Hersteller miteinander zu kommunizieren, Daten auszutauschen und die korrekte Interpretation der übertragenen Informationen. Offene Systeme zeichnen sich durch Interoperabilität aus.

Die Interoperabilitätstests sollen untersuchen, ob Geräte eines oder verschiedener Hersteller in einem System betrieben werden können. D.h., die Geräte sollen nicht nur Daten untereinander

austauschen können, sie sollen die Daten auch richtig interpretieren können. Der Test der Interoperabilität erweist sich weitaus schwieriger als der Protokoll- oder Negativtest. Er kann in drei Stufen gegliedert werden:

- Test zwischen zwei Geräten.
- Integrationstest: Ein Test zwischen mehreren Geräten, der sich auf das Zusammenwirken von Systemteilen konzentriert.
- Systemtest: Test, bei der das gesamte System gegen eine Spezifikation getestet wird.

Unter Interoperabilitätstest fällt insbesondere die Überprüfung des dynamischen Verhalten des einzelnen Gerätes in einem System und das dynamische Gesamtverhalten des Systems bei

- unterschiedlicher Beanspruchung der Geräte,
- unterschiedlicher Belastungen des Systems, d.h. der Belastung des Kommunikationsbusses.

Prinzipiell müssten immer paarweise alle Geräte eines Systems gegeneinander getestet werden, was in großen Systemen zu einer großen Anzahl

$$n = N \times (N - 1) / 2 \quad N = \text{Anzahl Geräte in einem System, } n = \text{Anzahl Tests}$$

durchzuführenden Tests führt. Bei z.B. 20 Geräten müssten 190 Tests gefahren werden. Diese Tatsache wiederum führt zu einem sehr hohen zeitlichen Aufwand bei beträchtlich ansteigenden Kosten was die Durchführbarkeit von Interoperabilitätstests ad absurdum führt. Weiterhin stellt sich die Frage, was zu tun ist, wenn zwei Geräte normkonforme Protokolle fahren aber nicht interoperabel zueinander sind. Im Besonderen dann, wenn die Norm oder die Spezifikation gegen die getestet wird keine Angaben über das Zeitverhalten der Kommunikation und des Datentransfers macht.

Beim gezielten Testen zwischen ausgewählten Anwendungen wird lediglich eine mögliche Zusammenarbeit zwischen den ausgewählten Geräten getestet. Durch solche Tests werden "Inseln von offenen Systemen" aufgebaut, die innerhalb der "Insel" problemlos arbeiten können. Werden der Insel weitere Geräte hinzugefügt, muss eine fehlerfreie Zusammenarbeit mit den anderen Geräten erst durch erneutes Testen geprüft werden.

Um überhaupt Interoperabilitätstest durchführen zu können, kann daher ein praktischer Ansatz gewählt werden. Anstatt die Geräte paarweise gegeneinander zu testen, wird ein Gerät als Referenzgerät definiert. Im Allgemeinen wird dies ein speziell für Testzwecke entwickeltes Testsystem sein. Die Theorie besagt, wenn alle Geräte gegen dasselbe Referenzgerät getestet wurden, sind die Geräte zueinander interoperabel. Allerdings ist diese Theorie noch zu beweisen.

Erfolgreich durchgeführte Interoperabilitätstests bestätigen zwar eine funktionierende Zusammenarbeit der Geräte, sie geben aber keine Gewähr über eine korrekt, entsprechend der Definition der Norm umgesetzte Schnittstelle. Voraussetzung für die Durchführung eines Interoperabilitätstest ist daher ein durchgeführter Protokolltest, der ein fehlerfreies Protokoll bestätigt hat. Erst wenn sicher ist, dass die Schnittstelle normkonform ist, ist es sinnvoll Interoperabilitätstests durchzuführen.

10.2.4 Abnahmetests

Die Abnahmetest setzen einen Test unter Beteiligung der Anwender des Systems voraus. Es ist ein Test gegen die expliziten Anforderungen des Auftraggebers, wie sie in einem Dokument (z.B. Testspezifikation) für beide Seiten verbindlich festgelegt sind und die impliziten Erwartungen des Auftraggebers, die dem allgemeinen Stand der Technik entsprechen. Es handelt sich im Allgemeinen um einen Systemtest aus Sicht des Auftraggebers mit folgenden Inhalten:

- Tests mit typischen Daten der Verarbeitungsaufgabe
- Tests mit Vollkapazitätsdaten
- Tests mit Überkapazitätsdaten
- Tests mit Fehlbedienungshandlungen

10.3 Testmanagement

10.3.1 Allgemeines

Zum Testmanagement gehört die Planung der Tests, d.h. wer testet was mit welchen Mitteln zu welchem Zeitpunkt, der Vergleich der Testergebnisse mit vorgegebenen Anforderungen (Testauswertung), Festlegung von Kriterien für das Ende des Testens (z. B. Überdeckungsgrade bezüglich einzelner Testkriterien) sowie die Entscheidung über das Ergebnis des gesamten Testens, d.h. Abbruch oder Fortsetzung.

Ein weiterer Bestandteil des Testmanagement ist die Teststrategie. Mit der Teststrategie wird die Auswahl und Bestimmung der Reihenfolge (des Zusammenspiels) von einzelnen Testverfahren (z.B. Protokolltest, Abnahmetest, etc.) und der Reihenfolge ihrer Anwendung auf Teilkomponenten des Gesamtsystems festgelegt.

10.3.2 Testszenarien

Grundlage der Tests sind die zu entwickelnden Testszenarien die beschreiben, wie ein Test ausgeführt werden muss, um die Konformität der Kommunikation zur Norm überprüfen zu können. Jedes Testverfahren beansprucht entsprechende Testszenarien und jedes Testszenario verlangt eine entsprechende Simulation des Prozesses, der Feldgeräte und der Stationsleitgeräte.

10.3.3 Testspezifikation

In der Testspezifikation wird beschrieben wie ein Test ausgeführt werden soll, d.h. was getestet werden soll. Sie beinhaltet somit die Testverfahren und Testszenarien die angewendet werden sollen, welche Ergebnisse vom Test erwartet werden und wie die Ergebnisse verwertet werden sollen. Die Spezifikation teilt sich in die Hauptbereiche

- abstrakte Spezifikation, die dem abstrakten Daten- und Dienstmodell der Normenreihe IEC 61850 entspricht, damit die Wiederverwertbarkeit der Spezifikation gewährleistet ist,
- Stack-orientierte Spezifikation, die abhängig vom verwendeten Kommunikationsstack ist,
- anwendungsfall-orientierte Spezifikation, die abhängig davon ist, welche optionalen Daten der Normenreihe IEC 61850 verwendet wurden.

10.4 Akzeptanz der Tests

10.4.1 Testprotokoll oder Zertifizierung

Das Wort Zertifizierung wird oft in einem falschen Zusammenhang verwendet. Die Prüfung und Bewertung eines Produkts mit dem die Erfüllung vorgegebener Anforderungen (z.B. Normenkonformität) nachgewiesen wird, kann in einem Testprotokoll oder in einem Testzertifikat bestätigt werden. Bei Erfüllung der gestellten Anforderungen kann ein Zertifikat ausschließlich von einer unabhängigen akkreditierten Prüfstelle vergeben werden. Ein Testprotokoll dagegen kann ein positives oder negatives Testergebnis bestätigen und muss nicht von einer unabhängigen akkreditierten Prüfstelle ausgestellt werden. Ein Testprotokoll kann demnach auch durch den Hersteller des Produkts ausgestellt werden. Prinzipiell ist es jedoch sinnvoll die Tests und somit auch das Testprotokoll von einer unabhängigen Instanz ausführen zu lassen. Zum einen ist dadurch Neutralität gewährleistet und zum anderen ist die Wahrscheinlichkeit hoch, dass eine andere Software als die des zu prüfenden Gerätes verwendet wird.

Bestandteile bzw. Inhalte eines Testprotokolls sind:

- die Testspezifikation und damit die Vereinbarung welches Produkt wie getestet werden soll,
- die normativen Referenzen,
- die Beschreibung des Testsystems (Komponenten, Konfiguration, Anforderungen),
- die Beschreibung der durchgeführten Tests mit den normativen Referenzen,
- die Resultate von jedem Test,
- die Wertung aller Tests mit negativem Ergebnis,
- die Empfehlungen aufgrund der Testergebnisse,
- die Namen der Autoren des Testprotokolls.

Derzeit existiert noch keine akkreditierte Prüfstelle die Zertifikate ausgeben könnte. Testdienstleister wie die FGH oder KEMA können daher bisher ausschließlich Testprotokolle ausstellen. Im Allgemeinen ist jedoch ein Testprotokoll völlig ausreichend.

10.4.2 Verantwortlichkeiten

Erfolgte für ein Gerät ein Konformitätstest bei einem unabhängigen Testdienstleister, ist im Testprotokoll detailliert beschrieben, welche Tests unter welchen Bedingungen mit welchen Resultaten durchgeführt wurden. Für die sorgfältige Durchführung und die Richtigkeit dieser Ergebnisse ist die Testeinrichtung verantwortlich.

Wurden alle Komponenten eines Systems auf diese Weise getestet, dürfte es keine Probleme grundsätzlicher Art geben. Sind aus Aufwands- und Kostengründen nicht alle im konkreten Prozessablauf auftretenden Fälle im Test enthalten, kann es durchaus zu Problemen bei einzelnen Funktionen kommen. In solchen Fällen, ist die Ursache für das Fehlverhalten zu bestimmen, z.B. mit Hilfe eines Protokollanalyzers. Es ist festzustellen, welches Produkt sich nicht nach dem Standard verhält. Der ermittelte Hersteller hat den Fehler zu beseitigen.

Für den Anwender erhöht sich die Wahrscheinlichkeit, dass es keine Überraschungen gibt, wenn von einer unabhängigen Testeinrichtung geprüfte Geräte eingesetzt werden.

Die Sicherheit kann erhöht werden, wenn zusätzlich ein Interoperabilitätstest, mit dem im System eingesetzten Komponenten, durchgeführt wurde. Eine 100%-ige Sicherheit wird jedoch nicht gegeben sein.

10.5 Testtools

Zur Durchführung der unterschiedlichen Testverfahren werden zwei grundsätzliche Arten von Testtools eingesetzt: Analyser und Simulatoren. Es handelt sich um Programme, die in einem handelsüblichen PC installiert werden. Der PC muss über die Schnittstellen verfügen, die für die zu testenden Protokolle erforderlich sind, z.B. COM oder Ethernet. Der PC mit dem Testtool wird in die Übertragungswege des zu testenden Systems integriert.

10.5.1 Analyser

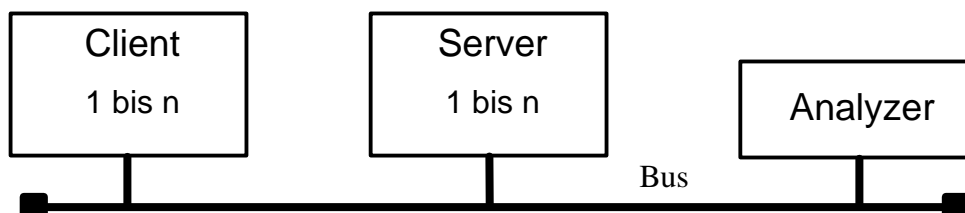


Bild 26 Einsatz eines Analysers

Ein Analyser wird in ein Netzwerk integriert, das aus mindestens zwei konfigurierten und laufenden Systemen (Client und Server) bestehen muss. Das Ziel ist, dass alle übertragenen Daten des Netzwerkes bzw. der Übertragungsstrecke „mitgehört“ und auf einem Monitor dargestellt werden können (Bild 26).

Merkmale von Analysern sind die komfortable Aufbereitung der Daten der einzelnen Protokollschichten einschließlich Klartextausgabe sowie die Erkennung und Kennzeichnung von auftretenden Protokollfehlern. Zur Auswertung stehen leistungsfähige Filterfunktionen zur Verfügung, und es besteht die Möglichkeit zur Erzeugung von Aufzeichnungsfiles.

Außer bei Interoperabilitäts- und Performancetests gibt es folgende Einsatzmöglichkeiten der Analyser:

- bei Werks- und Baustellen-Abnahmetests von komplexen Systemen
- bei der Systemintegration
- nach Software- und Konfigurationsänderungen
- bei fehlerhaften Verhalten.

10.5.2 Simulatoren

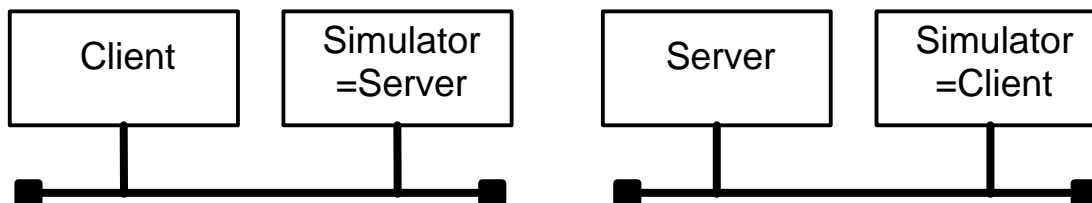


Bild 27 Einsatz eines Simulators

Mit einem Simulator wird je nach dem zu prüfenden System entweder ein Client oder ein Server nachgebildet (Bild 27). Jedes benötigte Telegramm kann im Simulator erzeugt und gesendet werden (auch mit Fehlern für den Negativtest). Die für den Test benötigten Telegrammfolgen können bei komfortablen Simulatoren einfach zusammengestellt werden und laufen teilweise automatisch ab.

Der Telegrammverkehr wird überwacht, Protokollfehler erkannt und gekennzeichnet. Die Daten werden zur Darstellung bearbeitet und z.B. in Klartextausgabe pro Protokollschicht ausgegeben. Komfortable Filtermöglichkeiten sowie die Möglichkeit der Erzeugung von Aufzeichnungsdateien gehören zum Funktionsumfang.

Neben dem Einsatz bei Konformitäts- und Negativtests kommen Simulatoren auf folgenden Gebieten zur Anwendung:

- Protokollentwicklung,
- Werks- und Baustellen-Abnahmetests.

10.6 Konformitätstest nach IEC 61850

Der Teil 10 der zukünftigen Normenreihe IEC 61850 normt einen Konformitätstest. Es wird der Prozess der Überprüfung beschrieben, ob sich ein Gerät/System nach IEC 61850 verhält. Gegenwärtig hat IEC 61850-10 den Status eines Arbeitspapiers (Working Draft, WD). Für Ende 2001 ist eine Veröffentlichung als Committee Draft (CD) geplant. Es sind überwiegend erst Lösungsansätze enthalten und nur einige Gliederungspunkte ausformuliert. Folgende Schwerpunkte sind Bestandteile der Konformitätstests:

- Konformitätstest-Prozeduren
- Qualitätssicherung
 - Qualitätsplan (Projektqualitäts- sowie Test- und Inspektionsplan)
 - Zeitplan
- Test
 - Gerätetests
 - Systemtests

Folgende Standardtestprozeduren werden z.B. empfohlen:

- Nachprüfung der Dokumentation und Kontrolle der Version des Gerätes (IEC 61850-4)
- Test des Geräte-Konfigurations-Files auf den standardisierten Syntax (IEC 61850-6)
- Test des Geräte-Konfigurations-Files auf das gerätebezogene Objektmodell (IEC 61850-7-3, IEC 61850-7-4)
- Test der Stackimplementation auf die verwendeten spezifischen Kommunikations-Service-Abbildungen (SCSM) in IEC 61850-8-x, IEC 61850-9-x
- Test der implementierten ACSI-Services auf ACSI-Definitionen (IEC 61850-7-2)
- Test der Basisfunktionen, wie Zeitsynchronisierung, Initialisierung usw. (IEC 61850-5)
- Test der gerätespezifischen Erweiterungen bezogen auf die Einhaltung der in IEC 61850 festgelegten Regeln.

Besonderer Wert wird auf die Qualitätssicherung und die Dokumentation der Konformitätstests gelegt. So ist für jedes Projekt ein Projekt-Qualitäts-Plan, der den Test- und Inspektions-Plan enthalten muss, zu erstellen. Die Dokumentation muss z.B. eine detaillierte Beschreibung und die Wertung der Resultate aller durchgeführten Tests enthalten.

Im Rahmen diese Forschungsvorhabens wurden die Konformitätstest auf den Protokolltest und den Interoperabilitätstest beschränkt. Die Tests wurden an allen sechs Feldgeräten und beiden Stationsleitgeräten der beiden Pilotsysteme "Profibus" und "Ethernet" durchgeführt.

Der grundsätzliche Aufbau der Pilotsysteme ist aus Bild 28 ersichtlich. Dieser besteht jeweils aus einem Stationsleitgerät, drei kombinierten Feldleit- und Schutzgeräten (Kombigeräte), einem Monitor und einer Prozesssimulation.

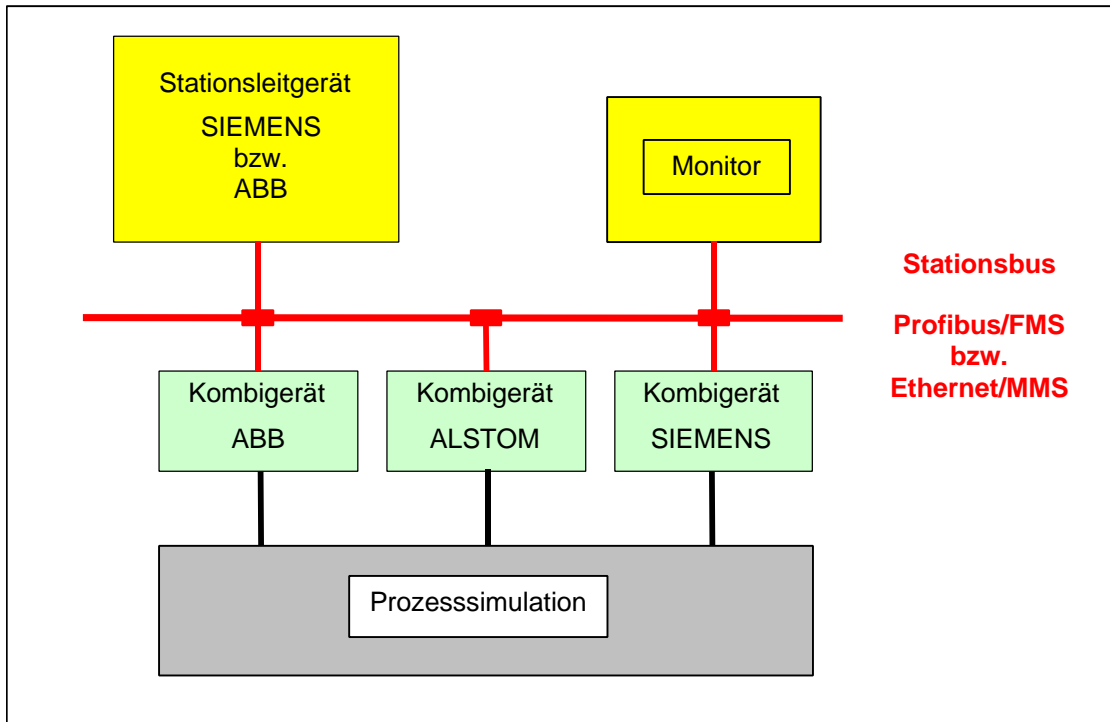


Bild 28 Aufbau der Pilotsysteme

Mit Hilfe des Monitors wird der Datenverkehr auf dem Stationsbus abgehört und analysiert. Die Prozesssimulation besteht aus einem Relais-Prüfgerät (OMICRON) und diversen Schaltern. Das Relaisprüfgerät stellt Spannungen und Ströme zur Verfügung und ist in der Lage Störfälle zu generieren. Mit den Schaltern werden binäre Signale generiert, die die Leistungsschalter und Trenner simulieren.

Die Konformität der Kommunikation zwischen Stationsleitgerät und Feldgerät soll folgendermaßen überprüft werden:

- Überprüfung des Verbindungsaufbaus und -abbaus (siehe Abschnitt 20.4.2).
- Überprüfung der Datenobjekte (siehe Abschnitt 20.4.3). Die Datenobjekte enthalten in ihren Attributen die Informationen, die übertragen werden. Es wird getestet, ob die Attribute die korrekten Werte enthalten und ob einem Datenobjekt die richtige Common Data Class zugeordnet worden ist.
- Überprüfung der Services (siehe Abschnitt 20.4.4). Die Informationen der Datenobjekte werden mit Services übertragen. Es wird überprüft, ob die Services das korrekte Format haben und die Attribute die korrekten Werte enthalten.
- Überprüfung der Common Data Class (siehe Abschnitt 20.4.7). Die Common Data Class werden Datenobjekten zugeordnet. Es wird getestet, ob die Common Data Class das korrekte Datenformat hat.
- Überprüfung der Befehlssequenzen (siehe Abschnitt 20.4.5). Um einen Befehl auszuführen, werden zwischen Client und Server verschiedene Services benutzt. Es wird überprüft, ob die Services die korrekte Reihenfolge einhalten und vollständig sind.
- Globale Überprüfung anhand eines Beispiels (siehe Abschnitt 20.8.7). Ein Leistungsschalter soll geschaltet werden.

Die allgemeinen Funktionstests die parallel zu den Teilaufgaben verlaufen, sollen daher durch folgende Untersuchungen bzw. Tests ergänzt werden:

- Durchgängigkeit der Datenmodellierung.
- Zukunftssicherheit der Datenmodellierung.
- Interoperabilität der Kombigeräte verschiedener Hersteller.
- Zeitdifferenzen zwischen den Geräten des Stationsbusses, bei aktivierter Zeitsynchronisierung. Diese Untersuchung wird für verschiedene Stationsbusse durchgeführt und dabei die Größe der Zeitdifferenzen ermittelt. Daraus lässt sich eine Aussage über die benötigte Genauigkeit der Zeitstempel machen.

Für die Tests ist eine Testspezifikation erstellt worden, in der alle durchzuführenden Tests mit den erwarteten Ergebnissen aufgeführt sind. Die Testspezifikation im Kapitel 20 beigefügt worden.

10.7 Testergebnisse

Vorraussetzung für ein erfolgreiches Zusammenarbeiten verschiedener Geräte innerhalb eines Systems sind im Vorfeld erfolgreich verlaufende Konformitätstests. Entsprechend dem Umfang der zukünftigen Normenreihe, stellen die Tests im Bereich der Energieversorgung eine noch nicht da gewesene Herausforderung dar. Ein erfolgreich absolvierter Protokolltest, wie z.B. bei der Norm IEC 60870-5-103, ist nicht mehr ausreichend sondern nur noch Grundvoraussetzung. Im Zuge des immer höher werdenden Automatisierungsgrades und dem damit erhöhten Datenaufkommens stellen Tests, ausgeführt durch unabhängige Prüfstellen eine der wichtigsten Bedingungen für die Zukunft dar. Das wichtigste Ergebnis des Test ist, dass diese Tests nicht mehr von Hand durchgeführt werden können. Der Umfang der Daten und die Kombinationsvielfalt für die Interoperabilitätstests bedingen eine Automatisierung der Testvorgänge. Im Fall der Überprüfung, ob die ausgetauschten Daten konform zur Norm sind ist dies sehr einfach zu bewerkstelligen. Im Fall einer Überprüfung der Interoperabilität sind weitere bzw. grundlegende Untersuchungen notwendig, um den Testaufwand in einem zeitlichen und wirtschaftlichen Rahmen zu halten. Im Kapitel 21 ist ein möglicher Ansatz für ein Testsystem vorgestellt.

Diese Teilaufgabe ist entsprechend dem AiF-Antrag vollständig und erfolgreich bearbeitet worden.

11 Der Nutzen

Die Betreiber von Schaltanlagen wollen bei der Anschaffung von Komponenten der digitalen Stationsleittechnik nicht mehr herstelleregebunden sein. Die Austauschbarkeit eines Gerätes gegen ein anderes ermöglicht dem Betreiber, sich in einem Gesamtsystem nicht auf einen Hersteller festzulegen, sondern einzelne Komponenten individuell zusammenstellen zu können. Für den jeweiligen Anwendungsfall können die am besten geeigneten Produkte unter technischen und wirtschaftlichen Gesichtspunkten ausgewählt werden. Hersteller dagegen müssen kein allumfassendes Stationsleitsystem anbieten um im Markt bestehen zu können, sondern können sich auf ausgewählte Komponenten der Sekundärtechnik konzentrieren. Somit haben auch kleine und mittlere Unternehmen die Möglichkeit in diesem Marktsektor z.B. mit Teillösungen zu bestehen, die bisherigen Positionen international zu verstärken bzw. ihre Produktpalette für diesen Marktsektor zu erweitern.

Genormte Schnittstellen tragen unter anderem zur Verlängerung der Lebensdauer der Sekundärsysteme bei, da auch zu einem späteren Zeitpunkt kompatible Geräte am Markt verfügbar sein werden, z.B. um Erweiterungen der Anlagen vornehmen zu können oder um neue Funktionen realisieren zu können.

Durch die neuen Standards und die damit einhergehende Informationsvernetzung in der Station, wird die Einbindung neuer Sensoren/Aktoren vereinfacht bzw. teilweise erst möglich gemacht. Diese Einbindung der gerätenahen Sensorik im Zusammenhang mit prozessnaher Intelligenz führt immer mehr von der zyklischen zur bedarfsorientierten Wartung, Instandhaltung und Erneuerung und ermöglicht somit geringere Betriebsaufwendungen und bei längeren Lebensdauern entsprechend geringere Instandhaltungskosten. Zu erwähnen sind auch die Schulungs- und Einarbeitungskosten für die immer komplexer werdenden Kommunikationssysteme, die bei entsprechender Normung deutlich gesenkt werden können.

Durch die beabsichtigte einheitliche Datenmodellierung und Kommunikation auf allen Ebenen der Automatisierung in Versorgungssystemen wird der Datenaustausch in Zukunft einheitlich und damit zu wesentlich geringeren Kosten möglich.

Schließlich erlaubt die digitale Kommunikation in Zukunft eine erweiterte Funktionalität, die der Versorgungsqualität der Kunden zugute kommen wird, da neue Störungsbehebungsstrategien kostengünstig realisiert werden können.

12 Ausblick

Die digitale Stationsleittechnik wird zukünftig in dem Maße verstärkt Einzug halten, wie auf der einen Seite zusätzlicher Funktionsaufwand und Informationsbedarf aufgrund von Netzoptimierungsaufgaben bzw. aus Anforderungen des deregulierten Marktes entstehen. Auf der anderen Seite wird der enorme Kostenverfall allgemein verfügbarer Lösungen aus dem Bereich der Büro- und Fertigungsautomatisierung den Einsatz dieser Lösungen unumgänglich machen. Die digitale Stationsleittechnik stellt die Plattform dar, bisher autonom betrachtete und realisierte Funktionsbereiche wie Automatisierung, Schutz und Zählung in ein Systemkonzept zu integrieren. Durch Einbeziehung der Sensorik rückt die Stationsleittechnik näher an den Prozess. Die so erzielbaren Rationalisierungspotentiale sind jedoch nur dann nutzbar, wenn sowohl die Betreiber als auch die Hersteller durch Normung, Standardisierung und Typisierung die langfristige Einsetzbarkeit der Systeme erreichen.

Die Anzahl der Funktionen, die in der digitalen Stationsleittechnik realisiert wurden, ist heute beachtlich. Hat sie doch im Laufe ihrer Entwicklung neben der Steuerung und Überwachung immer mehr Aufgaben übernommen – zum Beispiel die Fernwirkknotenfunktion, die Spannungsregelung oder die Synchronisierung. Aber all diese Funktionen kommunizieren im proprietären System herstellerspezifisch. Sie sind verschieden strukturiert und gestaltet. In dieser Situation fördert eine neue Norm für alle Kommunikationsaufgaben in der Station unweigerlich auch die künftige Vereinheitlichung der Funktionen, d.h. deren Erscheinungsbild an den Kommunikationsschnittstellen. Das wiederum wird zu vereinheitlichtem Engineering und somit zu Kostensenkungen für den Betreiber führen.

Andererseits wird durch die Norm die heute proprietäre Stationsleittechnik aufgebrochen. Im offenen System können prinzipiell alle Komponenten wie seinerzeit bei der konventionellen Technik von unterschiedlichen Herstellern stammen. Die Hersteller werden somit in die Lage versetzt, einzelne Geräte einfacher in vorhandene Systeme einzufügen. Aufwendige Entwicklungen sowie Pflege unterschiedlicher proprietärer Produktfamilien können künftig entfallen, was letztlich auch zu Kostenvorteilen führen wird. Die Steigerung der Wirtschaftlichkeit ist somit Triebkraft der Normung.

Vorteile der zukünftigen Normenreihe IEC 61850 sind u.a. die Unabhängigkeit von jeglicher Kommunikationsanschlüssen, die Trennung der Daten von den Diensten und der Kommunikation sowie das objektorientierte Datenmodell. Damit sind Wiederverwendbarkeit und Erweiterbarkeit gewährleistet. Mit diesen Eigenschaften kann der Normentwurf IEC 61850 nicht nur heutige Anforderungen abdecken, sondern wird auch für zukünftige Anforderungen gerüstet sein. Die Arbeiten von Pilotprojekten leisten einen wichtigen Beitrag für die Qualität der zukünftigen Normenreihe IEC 61850. Aus heutiger Sicht ist frühestens im Jahr 2003 mit einer Verabschiedung der Normenreihe zu rechnen.

13 Veröffentlichungen im Rahmen des Projektes

13.1 Veröffentlichungen mit Vortrag

- Schubert, H.; Shephard, B.; Janssen, M.: Standardised Communications in Substations, DPSP Conference, Amsterdam, 9. – 12. April 2001
- Dircks, P.: OCIS, der Weg zur Praxisreife, FGH/VDEW Fachtagung "Kommunikationsnormung in der Stationsleittechnik – Ergebnisse, Erfahrungen, Realisierungsansätze, Perspektiven", Eisenach, 14. bis 15. November 2000
- Schubert, H.; Dinges, R.: Stationsleittechnik, VBEW/FGH Seminar "Anforderungen und Möglichkeiten der Netzbetriebsführung im liberalisierten Markt", München, 9. Dezember 1999
- Schubert, H.: Project OCIS, Aims and first Results – An accompanying Project to the Standardisation of Communication Networks and Systems in Substations "IEC 61850", PSAC Conference Bled, Slovenian 6. – 8. October 1999
- Schubert, H.: Das Projekt OCIS (Open Communication in Substations), Workshop "Aktuelle Entwicklungen in der digitalen Schutz- und Stationsleittechnik", Sonnenenergie-Forum der VEW Energie AG, Dortmund, 23. bis 24. Februar 1999
- Schubert, H.: Pilotprojekt zur Erprobung herstellerneutraler Kommunikation innerhalb der Stationsleittechnik, VDEW-Infotage "Stationsleittechnik", Hannover, 17. September 1998

13.2 Veröffentlichungen ohne Vortrag

- Schubert, H.; Bauer, H.; Gebhardt, D.; Hoppe-Oehl, H., Oehring, M.; Walther, J-T: IEC 61850 - Kommunikation für die gesamte Stationsleittechnik, FGH/VDEW Fachtagung "Kommunikationsnormung in der Stationsleittechnik – Ergebnisse, Erfahrungen, Realisierungsansätze, Perspektiven", Eisenach, 14. bis 15. November 2000
- Schubert, H.; Bauer, H.; Gebhardt, D.; Hoppe-Oehl, H., Oehring, M.; Walther, J-T.: IEC 61850 – Einheitliche Kommunikation für die gesamte Stationsleittechnik, Elektrizitätswirtschaft 99 (Oktober 2000), Heft 21, Seite 28 - 32
- Schubert, H.; Haude, J.; A. Janz; Rudolph, T.; Schäffler, T.: A Pilot Project for Testing the Standard Drafts for Open Communication in Substations – First experiences with the future standard IEC 61850, CIGRE Conference Paris, France, 28. August till 02. September 2000
- Schubert, H.: Aktuelle Entwicklungen in der digitalen Stationsleittechnik - Erste Erfahrungen mit der zukünftigen Normenreihe IEC 61850, FGH-Geschäftsbericht, 1999
- Schubert, H.: Entwurf und Test von Kommunikationsschnittstellen für die digitale Leittechnik von Mittelspannungsanlagen - das Projekt OCIS, FGH-Geschäftsbericht 1999
- Schubert, H.; Haude, J.; Merke, E.; Rudolph, T.; Schäffler, T.: Normung in der Stationsleittechnik – Erfahrungen aus dem 1. Europäischen Pilotprojekt OCIS. Elektrizitätswirtschaft 98 (Oktober 1999), Heft 21, Seite 50 - 54

-
- Schubert, H.: FGH-Workshop – Aktuelle Entwicklungen in der digitalen Schutz- und Stationsleittechnik, FGH-Informationen, Januar 1999
 - Schubert, H.: Standardisierung in der digitalen Stationsleittechnik – Das Projekt OCIS, FGH-Geschäftsbericht 1998
 - Schubert, H.: Standardisierung in der digitalen Stationsleittechnik – Das Projekt OCIS, FGH-Informationen, März 1998

13.3 Vorträge

- Schubert, H.: Discussions about today's and future standards for communication in substations. Introduction of the project OCIS, CIGRE Conference Paris, France, Group 34, 28. August till 02. September 2000
- Schubert, H.: Substation Control – Project OCIS, IEC TC 57 Meeting "Power System Control and Associated Communication", Berlin, 8. – 11. November 1999
- Schubert, H.: IEC 61850 – UCA – Project OCIS, CIGRÉ SC 34 Colloquium Florence, Italy, 11.-15. October 1999
- Schubert, H.: Entwurf und Test von Kommunikationsschnittstellen für die digitale Leittechnik von Mittelspannungsanlagen – Das Projekt OCIS, FGH-Forschungsbeirat, 23. März 1999
- Schubert, H.: Aktuelle Entwicklungen in der Standardisierung der digitalen Stationsleittechnik – Die zukünftige Normenreihe IEC 61850 im Spiegel des Projektes OCIS, FGH-Mitgliederversammlung, 4. November 1998

13.4 Ausstellungen

- DPSP Conference Amsterdam, The Netherlands, 9. – 12. April 2001
- CIGRE Conference Paris, France 28. August till 02. September 2000
- FGH/VDEW Fachtagung "Kommunikationsnormung in der Stationsleittechnik – Ergebnisse, Erfahrungen, Realisierungsansätze, Perspektiven", Eisenach, 14. bis 15. November 2000
- IEC TC 57 Meeting "Power System Control and Associated Communication", Berlin, 8. bis 11. November 1999

14 Verwendete Abkürzungen

ACSI	Abstract Communication Service Interface
API	Application Program Interface
ASN.1	Abstract Syntax Notion One
AWE	Automatische Wiedereinschaltung
BER	Basic Encoding Rules
CASM	Common Application Service Models (Part of UCA.2)
CORBA	Common Object Request Broker Architecture
CSMA/CD	Carrier Sense Multiple Access with Collision Detection
DI	Device Identity
DUT	Device Under Test
EPA	Enhanced Performance Architecture
EPRI	Electric Power Research Institute
GM	General Motors
GOMSFE	Generic Object Model for Substation & Feeder Equipment (Part of UCA.2)
HDLC	High-level Data Link Control (Verbindungsschicht Protokoll)
HMI	Human Machine Interface
IEC	International Electrotechnical Commission
IED	Intelligent Electronic Device
IEEE	Institute of Electrical and Electronics Engineers
ISO	International Standardisation Organisation
LER	Local Encoding Rules
LLC	Logical Link Control
LN	Logical Node
LS	Leistungsschalter
MAC	Media Access Control
MAP	Manufacturing Automation Protocols
MFC	Microsoft Foundation Classes
MMS	Manufacturing Message Specification (ISO/IEC 9506)
OSI	Open Systems Interconnection
PDU	Protocol Data Unit
SAS	Substation Automation System
SQL	Structured Query Language (Datenbankabfragesprache)

TC	Technical Committee
UCA	Utility Communication Architecture
UML	Unified Modelling Language
UMZ	Unabhängiger Maximalstrom Zeitschutz
VDEW	Vereinigung Deutscher Elektrizitätswerke e.V.
WA	Windows-Applikation
WG	Working Group

15 Literatur

- [1] Kern, C.; Schäffler, Th.: Moderne Kommunikation in der Fernwirk- und Leittechnik. Elektrizitätswirtschaft, Zeitschrift des VDEW, Jg. 96 (1997), Heft 5, Seiten 182-186.
- [2] Buchholz, B. M.; Kolberg, B.; Schuster, N.: Schutz- und Stationsleittechnik sprechen die gleiche Sprache. Siemens EV-Report 2/94.
- [3] IEC TC57: IEC 870: Telecontrol, teleprotection and associated telecommunications for electric power systems. Part 5: Transmission protocols. Section 101: Companion standard for basic telecontrol tasks. Section 103: Companion standard for the informative interface of protection equipment.
- [4] DVG: Konzeption künftiger Informationssysteme für Lastverteiler/Schaltleitungen, Anforderungen an die neue Stationsleittechnik in einem Gesamtkonzept der Netzleittechnik. Deutsche Verbundgesellschaft e.V. Heidelberg, 1989.
- [5] März, W.: Prozeßdatenübertragung mit OSI-Protokollen - IEC 870-6 in Netzleitsystemen, Elektrizitätswirtschaft, Zeitschrift des VDEW, Jg. 93 (1994), Heft 24, Seiten 1539-1545.
- [6] ISO 7498-1: OSI-Reference Model
- [7] CIGRE WG 34.03: Communication requirements in terms of data flow within substations. Final report 96]12/01.
- [8] Becker, G.; Gärtner, W.; Kimpel Th.; Link, V.; März, W.; Schmitz, W.; Schwarz, K.: Offene Kommunikationsplattformen für die Leittechnik nach IEC 870-6 am Beispiel der Netzleittechnik. etz Report 28.
- [9] IEC 61850: Communication Networks and Systems in Substations
 - IEC 61850-1 Introduction and Overview
 - IEC 61850-2 Glossary
 - IEC 61850-3 General Requirements
 - IEC 61850-4 System and project management

- IEC 61850-5 Communication Requirements for Functions and Device Models
 - IEC 61850-6 Configuration description language for substation IEDs
 - IEC 61850-7-1 Principles and Models
 - IEC 61850-7-2 Abstract Communication Service Interface (ACSI)
 - IEC 61850-7-3 Common Data Classes
 - IEC 61850-7-4 Compatible Logical Node and Data Classes
 - IEC 61850-8-1 Specific Communication Service Mapping (SCSM), Mapping to MMS (ISO/IEC 9506 Part 1 and Part 2) over ISO 8802-3
 - IEC 61850-9-1 Specific Communication Service Mapping (SCSM) – Serial Unidirectional Multidrop Point to Point Link
 - IEC 61850-9-2 Specific Communication Service Mapping (SCSM) – ACSI mapping on an IEEE 802-3 based Process Bus
 - IEC 61850-10 Conformance Testing
- [10] IEEE-SA TR 1550, IEEE-SA Technical Report on Utility Communications (UCA™), Version 2.0
Common Application Service Models (CASM) and Mapping to MMS
Generic Object Models for Substation & Feeder Equipment (GOMSFE) Profiles
- [11] VDEW: Netzleitsysteme in EVU, Teil 6: Integrierte Leittechnik in Stationen. Vereinigung Deutscher Elektrizitätswerke-VDEW-e.V. Frankfurt (Main), 1987
- [12] VDEW/ZVEI-Empfehlung zur seriellen Schnittstelle der Schutzeinrichtungen in integrierten Stationsleitsystemen der Elektrizitätsversorgungsunternehmen. Vereinigung Deutscher Elektrizitätswerke-VDEW-e.V. Frankfurt (Main), 2. Ausgabe 1993.
- [13] VDEW Publikation: Digitale Stationsleittechnik - Empfehlungen. Vereinigung Deutscher Elektrizitätswerke-VDEW-e.V. Frankfurt (Main), 1. Ausgabe 1994.
- [14] VDEW Publikation: Digitale Stationsleittechnik - Empfehlungen, Ergänzungen zur Anwendung in Verteilnetzstationen. Vereinigung Deutscher Elektrizitätswerke-VDEW-e.V. Frankfurt (Main), 1. Ausgabe 1998, Entwurf.
- [15] Rumbough, James; Blaha, Michael; Premerlani, William; Eddy, Frederick; Lorenzen, William: Objektorientiertes Modellieren und Entwerfen Hanser Verlag, 1994.
Original: Object-Oriented Modeling and design, Prentice Hall 1991. ISBN: 0-13-630054-5
- [16] Tagungsband zum FGH-Workshop "Aktuelle Entwicklungen in der digitalen Schutz- und Stationsleittechnik" am 23. und 24.02.99 in Dortmund.

16 Anhang A: Anforderungen

Grundsätzlich sollen die Funktionen des Leit- und Schutzsystems den Empfehlungen der VDEW für Digitale Stationsleittechnik [13] sowie den Ergänzungen zur Anwendung in Verteilnetzstationen [14] entsprechen.

Im Folgenden sind besonders die Funktionen herausgezogen, die von der Kommunikation zwischen den Kombi-Geräten und der Bedien-/Zentral- Einheit abhängig sein können.

16.1 Systemeigenschaften

16.1.1 Identifikation

Jedes logische Gerät, also die virtuellen Darstellungen von Kombi-Gerät, Trenner, Erder und Leistungsschalter (LS) soll ein elektronisches Typenschild (Device Identity) erhalten, das von der Bedieneinheit aus gelesen werden kann.

16.1.2 Systemanlauf

Nach einem Ausfall der Hilfsenergie muss das Gesamtsystem bei wiederkehrender Spannung selbsttätig anlaufen (automatischer Restart, Zeitdauer für Systemwiederanlauf ≤ 5 min). Nach jedem Wiederanlauf ist eine automatische Generalabfrage mit Alt-/Neu- Vergleich durchzuführen und Änderungen sind allen Funktionsblöcken, die die neuen Informationen benötigen mitzuteilen. Jeder Wiederanlauf ist der Leitstelle zu melden.

16.1.3 Überprüfung der Funktionsblöcke

Um defekte Eingangsbaugruppen zu erkennen und die Kongruenz der Abbilder in den einzelnen Funktionsblöcken überprüfen zu können, sollte während des Betriebs eine "schleichende" Generalabfrage vorgesehen werden.

Von Ferne muss erkennbar sein, ob die Schutz- oder Steuerfunktion gestört ist.

Es sollte die Möglichkeit bestehen, die Zustände der Funktionsblöcke abzufragen und wie Prozessmeldungen zu verarbeiten.

Bei erkannten Fehlern müssen Systemfehlermeldungen erzeugt werden. Gegebenenfalls müssen Prozessinformationen in Abhängigkeit von Systemfehlern als ungültig bzw. nicht aktualisiert markiert werden.

16.1.4 Wiederanlauf von Funktionsblöcken

Beim Ausfall oder beim Wiedereinschalten eines Funktionsblockes darf in anderen Funktionsblöcken kein Fehlverhalten ausgelöst werden.

16.1.5 Datenerhaltung

Für das Datenmodell ergibt sich die Forderung, dass in jedem Funktionsblock, das für die eigenen Funktionen notwendige Teildatenmodell unverlierbar gespeichert wird. Auch bei Wiederanlauf eines Funktionsblockes nach einem Ausfall darf nicht auf ein zentrales Datenmodell zurückgegriffen werden.

Die vor Auftreten einer Störung an einer Feldeinheit vorhandenen Stellungsmeldungen müssen festgehalten werden und sind in Anzeigen und Archiven als nicht aktualisiert zu kennzeichnen.

16.1.6 Zeitverhalten

Verarbeitungszeiten und Übertragungszeiten müssen in Melderichtung und in Befehlsrichtung kleiner 1 sec sein. Für die Melderichtung wird diese Zeit gemessen vom Umschalten des Gerätekontaktes bis zur Anzeige in der Stationsbedienung. In Befehlsrichtung gilt die Zeitfestlegung vom Absetzen eines Befehls in der Stationsbedienung bis zum Schließen des Befehlsausgabekontaktes.

Messwerte müssen von der Quelle (Eingang Feldleitgerät) bis zur Senke innerhalb der Stationsleiteinrichtung innerhalb von 1 sec aktualisiert sein.

16.1.7 Zeitsynchronisierung

Dies ist die Versorgung aller Einrichtungen in der Station mit der absoluten Zeit und Überwachung der Zeitsynchronisierung bzw. der synchronen Zeit.

16.1.8 Informationssperren

Durch eine Informationssperre (Infosperre) wird ein physikalisches Gerät von der Kommunikation entkoppelt, d.h. es werden ab dem Setzen der Infosperre, keine Informationen wie Befehle und Meldungen mehr ausgetauscht. Zur Blockierung der Befehlsrichtung siehe auch 16.4.7.

Die Bedienung der Informationssperren sollte zentral (Nah-Steuerung) möglich sein, eine zusätzliche Möglichkeit die Infosperre vor Ort zu setzen ist nicht erforderlich. Gesetzte Informationssperren müssen angezeigt werden können und an die betroffenen Ebenen (Nah- und Fern-) gemeldet werden.

Alle Informationen, die durch die Infosperre unterdrückt werden, werden verworfen, d.h. sie werden auch nach Rücknahme der Infosperre nicht gesendet. Nach Entfernen der Infosperre müssen die Meldeabbilder z.B. durch Generalabfrage aktualisiert werden.

16.2 Messwerte

16.2.1 Umfang

Siehe Anhang A: Anforderungen und Anhang B: Projektierungsliste.

16.2.2 Messwertberuhigung

Die Messwertberuhigung gewährleistet, dass die Messwerte nur bei folgenden Kriterien übertragen werden:

- Der Wert ändert sich sprunghaft um einen individuell definierten Betrag.
- Das Integral der Änderung seit der letzten Übertragung ist größer als ein individuell definierter Betrag.
- Eine einstellbare Zykluszeit für die zyklische bzw. periodische Übertragung der Messwerte wird überschritten.

16.2.3 Umrechnung

Der Wert soll prozentual bezogen auf den Nennwert oder als Absolutwert anzeigbar sein.

16.2.4 Nullpunktunterdrückung

Falls ein Abzweig ausgeschaltet ist, so muss auch der angezeigte Strom Null betragen.

16.3 Meldungen

16.3.1 Umfang

Siehe Anhang A: Anforderungen und Anhang B: Projektierungsliste.

16.3.2 Zeitzuordnung

Dies sind Markierungen mit der Absolutzeit des Entstehens bei internen Informationen oder mit der Absolutzeit der Erfassung bei externen Informationen. Diese Zeit muss bezogen auf eine Stationsleiteinrichtung für alle Meldungen mit einer Genauigkeit von 10 ms und einer Auflösung von 1 ms erfasst und zusammen mit der Information zu jeder Senke übertragen werden.

16.3.3 Unterdrückung Flattermeldungen

Flattermeldungen sind Meldungen, die innerhalb sehr kurzer Zeit mehrfach kommen und gehen. Das Auftreten einer solchen Information muss zu Beginn einmal verarbeitet werden, die weiteren Kommen- und Gehen-Ereignisse sollen unterdrückt werden.

16.3.4 Unterdrückung kurzzeitiger Meldungen

Meldungen, bei denen Kommen- und Gehen-Ereignis innerhalb einer kurzen Zeitspanne liegen, sollten an geeigneter Stelle unterdrückt werden.

16.3.5 Störstellungsunterdrückung

Für eine definierbare Zeit müssen die Schalterstellungsanzeigen “weder EIN noch AUS” und “sowohl EIN als auch AUS” unterdrückt werden. Nach Ablauf dieser Zeit muss der Zustand “Störstellung” verarbeitet werden.

16.3.6 Sammelmeldungsbildung

Dies ist die Zusammenfassung einzelner Meldungen zu einer Sammelmeldung. Sammelmeldungen müssen wie Einzelmeldungen verarbeitbar sein. Jede Einzelmeldung muss für beliebig viele Sammelmeldungen nutzbar sein. Sammelmeldungen sind wie Einzelmeldungen mit einer Zeitinformation zu versehen. Jede neue Einzelmeldung sollte zu einer erneuten Übertragung abhängiger Sammelmeldungen führen.

16.3.7 Detailinformationen

Detailinformationen von Sammelmeldungen müssen zur Übertragung auf Anforderung bereitgehalten werden.

16.3.8 Schalterfallmeldung

Falls vom Prozess nicht angeboten, muss aus den Informationen “LS von EIN nach AUS” und der Information “keine Steuerung durch einen Bediener” über ein Zeitfenster die Information “Schalterfall” erzeugt werden (spezielle Form der Sammelmeldungsbildung).

16.3.9 Verwaltung von Meldeabbildern

Meldeabbilder sind zur Versorgung aller Komponenten einer Stationsleiteinrichtung an geeigneter Stelle zu führen. Über den Datenaustausch ist die Konsistenz und Synchronität aller Daten zu gewährleisten (spontaner und zyklischer Datenabgleich). Bei Störungen oder Ausfall im Datenaustausch der Leittechnik oder der Prozessdatenerfassung muss es möglich sein, die Informationen der Meldeabbilder zu speichern und mit der Zusatzkennung “nicht aktualisiert” zu versehen.

16.4 Steuerung

16.4.1 Umfang

Siehe Anhang A: Anforderungen und Anhang B: Projektierungsliste.

16.4.2 Rückmeldung

Jede Aktion des Bedieners muss eine Reaktion des Systems zur Folge haben. Dieses bestätigt die Bedienung oder weist sie zurück. Abweisungen müssen eine Begründung mit leicht verständlichen Fehlerhinweisen im Klartext enthalten. Der Start, das Beenden oder Abbrechen einer Bedienung muss besonders einfach und in allen Bedienstufen in gleicher Weise möglich sein.

16.4.3 Doppelbetätigungssperre

Dies ist die Durchführung einer Prüfung für ausgewählte Befehle zur Verhinderung einer zeitgleichen Ausgabe mehrerer Befehle.

16.4.4 Befehlslaufzeitüberwachung

Dies ist die Abstimmung der Befehlsausgabe nach Ausführung oder nach einer definierten Zeit bei nicht ordnungsgemäß ausgeführten Befehlen.

16.4.5 Feldverriegelung

Der Sammelschienentrennschalter soll nur schaltbar sein, wenn der Leistungsschalter AUS ist. Während der Betätigung des S1-Schlüsseltasters kann vor Ort unverriegelt geschaltet werden. In der Ereignisliste soll ein unverriegelter Schaltvorgang entsprechend gekennzeichnet sein, z.B. durch „LS BEFEHL ORTSTEU RM AUS - UNVERRIEGELT“.

16.4.6 Zulässigkeitsprüfung

Befehlsausgaben sind vor Ausführung auf Zulässigkeit zu überprüfen. Liegen bestimmte Ereignisse vor, ist die Durchführung zu blockieren und eine Meldung zu erzeugen.

16.4.7 Blockierung der Befehlsrichtung

An der Ortsteuerung soll die Befehlsrichtung von den übergeordneten Bedieneinheiten gesperrt werden können (Feld-Ort/Fern-Schalter). Eine Sperre muss an den Bedieneinheiten angezeigt werden können.

16.4.8 Befehlsherkunft

In der Stations-Zentraleinheit muss die Herkunft des Schaltbefehls erkennbar sein.

16.4.9 Nachführung

Eine Schalterstellung muss an der Nahbedienung von Hand eingegeben werden können (d.h. die Schalterstellung wird „nachgeführt“), dies ist z.B. erforderlich, wenn die Melderichtung gestört ist (z.B. Infosperre gesetzt) oder wenn keine Hilfsschalter zur Stellungserfassung

vorhanden sind, z.B. bei Erdungsgarnituren. Nachgeführte Stellungen müssen von gemeldeten Stellungen in der Anzeige und in der Ereignisliste unterschieden werden.

16.5 Protokollierung / Archivierung (optional)

16.5.1 Archivierung

Informationen (Messwerte, Zählwerte, Meldungen) müssen für eine spätere Auswertung vollständig unverlierbar archiviert werden. Die Archiv-Listen können aus den Protokollen (Ereignislisten) abgeleitet werden, ggf. ergänzt um dort nicht aufgeführte Informationen.

16.5.2 Protokollierung

Dies ist die Ausgabe von Betriebsereignissen und Zuständen spontan oder auf Anforderung gegebenenfalls nach bestimmten Sortierkriterien auf Sichtgerät oder Drucker.

Informationen in Protokollen und Listen (Ereignislisten) sollen übersichtlich und mit folgender Reihenfolge ausdrückbar sein:

- Datum und Uhrzeit
- Spannungsebene
- Abzweig
- Betriebsmittel
- Ereignis
- Zustand

16.6 Schutz

16.6.1 UMZ-Schutz

Auf den Leitungsschalter soll ein ungerichteter zwei-stufiger UMZ-Schutz wirken.

16.6.2 Informationsumfang

Der Informationsumfang, der an der Stationsbedieneinheit abrufbereit sein soll, umfasst:

- Schutzinformationen von Netzstörungen
- Störwerte analoger Größen mit einem Zeitraster von 1 bis 2 ms (siehe Störschriebe, 16.8)
- Zustandsmeldungen (z.B. Überstromschutz ein/aus)
- Informationen interner Störungen und Überwachungen des Gerätes.
- Einstellwerte

16.6.3 Rückwärtige Verriegelung (optional)

Um eine direkte Kommunikation zwischen den Kombigeräten zu ermöglichen, soll eine rückwärtige Verriegelung verwirklicht werden.

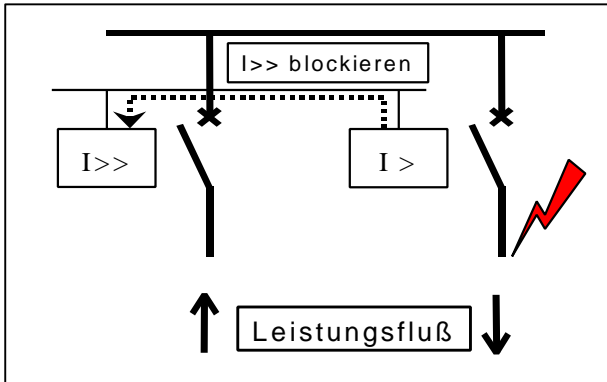


Bild 29 Rückwärtige Verriegelung

Stellt zum Beispiel das linke Schaltfeld im Bild 29 eine Einspeisung und das rechte einen Abgang dar, so soll bei einer Schutzanregung im linken geprüft werden, ob auch der Schutz im rechten Feld angeregt hat. Dies wird ihm durch die Meldung „I>> blockieren“ mitgeteilt. Ist dies der Fall, so soll die Auslösung in Schnellzeit (I>>) im linken Feld blockiert werden, um den Reserveschutz zu garantieren wird die zweite Zeitstufe (I>) nicht blockiert. Durch die Schutzparametrierung wird dieses Zusammenspiel der Schutzgeräte eingestellt. Die rückwärtige Verriegelung wird bisher über die parallel verdrahtete Meldung „Generalanregung“ sichergestellt. Liegen bei einer Störung mehrere Anregungen an, können diese als Sammelmeldung „Generalanregung“ übertragen werden. In diesem Fall liegt beim Empfänger nur die Meldung an, dass eine Anregung vorliegt, die Information in welchem Leiter fehlt.

Hier soll aber die Machbarkeit über die serielle Kommunikation getestet werden. Ein solches Ereignis soll auch in der Bedieneinheit entsprechend angezeigt werden.

16.6.4 Distanzschutz (optional)

Auf den Leistungsschalter soll ein Distanzschutz wirken. Bei Ausfall wird der UMZ-Schutz wirksam.

16.6.5 Erdschlusserfassung und Quittierung (optional)

Jedes Feld soll die Funktion eines Erdschlusswischerrelais erhalten. Die Verlagerungsspannung zur Erkennung des Erdschlusses erhält es aus dem Erdungs- und Messfeld. Die Richtung des Erdschlusses wird z.B. mit Hilfe des „Wattmetrischen Prinzips“, d.h. dem „Watt“-Reststrom der Petersenspule ermittelt.

Erkannte Erdschlüsse werden von zentraler Stelle an allen Feldgeräten gleichzeitig zurückgesetzt (gelöschtes Netz).

16.6.6 Automatische Wiedereinschaltung - AWE (optional)

Die automatische Wiedereinschaltung (AWE) kann von zentraler Stelle oder direkt am Feldleitgerät unwirksam gemacht werden. Dazu soll im Feld kein separater Schalter vorgesehen werden, um Kosten einzusparen.

16.7 Parametrierung (optional)

16.7.1 Off-line-Parametrierung

Off-line-Parametrierung (d.h. betrieblich entkoppelt) ist das Erzeugen eines geänderten Datenmodells über die Parametriereinrichtung. Das Einbringen des Datenmodells und die notwendigen Inbetriebsetzungsprüfungen erfordern vorübergehend eine Einschränkung des Netzbetriebes. Das Einbringen des Datenmodells soll von zentraler Stelle möglich sein.

16.7.2 On-line-Parametrierung

Dies ist der Direkteintrag von Parameterwerten in die Prozessrechnerlisten ohne Parametriereinrichtung bei uneingeschränktem Netzbetrieb. Wegen fehlender Prüfmöglichkeiten wird dies, außer für Betriebsparameter, nicht empfohlen (Schutzparameter sind keine Betriebsparameter.). Eingaben, die außerhalb der in den technischen Beschreibungen genannten Einstellbereiche liegen, müssen abgewiesen werden. Das Einbringen der Daten soll von zentraler Stelle möglich sein.

16.7.3 Fernaktivieren

Dies ist kein Parametriervorgang, es wird lediglich innerhalb eines Datenmodells von den angesprochenen Programmen auf andere Listenbereiche zugegriffen. Das Ansprechen der Programme ist wie eine Fernsteuerung zu sehen. Es soll z.B. zwischen verschiedenen Schutzparametergruppen von zentraler Stelle umgeschaltet werden können.

16.7.4 Lesen von Parametern / Erkennen von Parametersätzen

Bestimmte Parameter eines Kombi-Gerätes sollen über die Stationsleittechnik an zentraler Stelle gelesen werden können. Von zentraler Stelle soll der eingestellte Parametersatz erkennbar sein.

16.8 Störschriebe (optional)

16.8.1 Datenumfang

Es soll von mindestens drei Netzstörungen die binären Schutzinformationen gespeichert werden. Bei gefülltem Speicher und weiteren Vorgängen ist der jeweils älteste Vorgang zu löschen (first in first out). Es müssen die Störwerte analoger Größen von mindestens 3 s Länge gespeichert werden. Die Speicherung muss von der Generalanregung (Abschn. 16.6.3) veranlasst werden. Die einer Störung vorausgehenden 100 ms sind ebenfalls zu erfassen. Es wird der jeweils letzte Vorgang abgespeichert.

16.8.2 Zuordnung

Die Zuordnung von Schutzinformationen und Störwerten analoger Größen muss eindeutig sein.

16.8.3 Fernauswertung

Die Störprotokolle sollen an zentraler Stelle abrufbar und auswertbar sein.

17 Anhang B: Projektierungsliste

In der Projektierungsliste werden die notwendigen Aktionen (Meldungen, Befehle, etc.) aufgelistet damit die Anforderungen aus Kap. 16 erfüllt werden können. In der folgenden Tabelle sind die in der Projektierungsliste verwendete Abkürzungen aufgelistet.

ANR	Anregung	KOM / GEH	kommend/gehend
BF	Befehl	LED	Leuchtdiode
DIST	Distanz	MOT	Motor
ERDGARN	Erdungsgarnitur	Mx	Meldekontakt
ERDS	Erdschluss	RIC	Richtung
Ex	Binär- Ein/Ausgang	RM	Rückmeldung
FEH	Fehler	SPE	Sperre
F-O/F	Feld- Ort/Fern Schalter	SPG	Spannung
GS	Gleichstromkreis	STÖR	Störung
INFSPE NAH	Informationssperre Nahsteuerung	UEST	Überstrom
LS	Leistungsschalter		

Erklärung der Bedeutung der Einträge in der folgenden Tabelle.

- Unter Sekundärstromkreis wird die Versorgungsspannung sowie Steuerspannung der Sekundärtechnik verstanden. Die Information KOM/GEH bezieht sich in diesem Fall auf den Ausfall bzw. dem Wiedereintritt der Spannung.
- Unter Kontakt werden die binären Ein- und Ausgänge (Ex) sowie die Meldekontakte (Mx) verstanden.
- Das E (Ereignis) in der Tabellenüberschrift steht für einen Eintrag eines Vorfalles in die Ereignisliste
- Das V (Verarbeitung) in der Tabellenüberschrift steht für einen Eintrag in die Warnmeldeliste, in die Ereignisliste und dem aktivieren der Alarmhupe.
- Ein E (Ereignis) in der Spalte E steht für die Forderung, das ein Vorfall als Ereignis in die Ereignisliste eingetragen werden soll.
- Ein W (Warnung) in der Spalte E steht für die Forderung, das ein Vorfall als Warnung in die Ereignisliste, in die Warnmeldeliste eingetragen werden soll und zusätzlich die Alarmhupe aktiviert werden soll.
- Ein A (Anzeige) in der Spalte V steht für die Forderung, einen Vorfall zusätzlich zu einem Listeneintrag auch im Anzeigebereich (Display) eines Feldgerätes anzuzeigen.
- Ein Q (Quittieren) in der Spalte V steht für die gleichen Forderungen wie ein A, jedoch muss der Vorfall zusätzlich bestätigt (quittiert) werden.
- Grau hinterlegte Felder zeigen an, das ein Binäreingang erforderlich ist.
- Kursive Schrift zeigt an, das es sich um optionale Funktionalitäten handelt.

Informationen			E	V	Kontakt	Bemerkung
	Verknüpfung					
1. Systeminformationen						
Sekundärstromkreise						
	220V GS 1 FEH	KOM / GEH	W	A	E14	Gleichstromkreis 1 (GS): Schutz, LS
	INFSPE NAH					Informationssperre Nahsteuerung (INFSPE NAH)
	220V GS 2 FEH	KOM / GEH	W	A	E15	Gleichstromkreis 2 (GS): Trenner, Meldung, Steuerung
	INFSPE NAH					
	220V GS MOT-SPG FEH	KOM / GEH	W	A	E16	Stromkreis für Motorspannung (MOT-SPG)
	INFSPE NAH					
Infosperren						
Infosperren werden immer an der Nahsteuerung gesetzt.						
	INFSPE NAH	KOM / GEH	E			Infosperre zwischen Feld und Station für Meldungen und Befehle

Informationen			E	V	Kontakt	Bemerkung
Verknüpfung						
Meldungen der Funktionsblöcke allgemein						
KOMBIGERÄT NICHT BEREIT		KOM / GEH	W	A		
Diese Meldung war bisher parallel verdrahtet, eine Verknüpfung mit INFSPE NAH bestand daher nicht.						
Alle weiteren Warn- und Störmeldungen der Funktionsblöcke sind geräteabhängig.						
Die Auswahl, welche Meldungen in der Ereignisliste protokolliert werden sollen, geschieht daher nach Rücksprache mit den Herstellern.						
Für folgende Meldungen ist in den VDEW-Empfehlungen eine parallele Verdrahtung vorgesehen:						
GERÄT STÖRUNG					M1	
GERÄT WARNUNG					M2	
Feld- Ort/Fern Schalter (F-O/F)						z.B. Schlüsselschalter am Kombigerät
F-O/F		RM STÖR	E			Rückmeldung Störung (RM STÖR)

Informationen			E	V	Kontakt	Bemerkung
	Verknüpfung					
	INFSPE NAH					
F-O/F		RM ORT	E			
	INFSPE NAH					
F-O/F		RM FERN	E			
	INFSPE NAH					
Sonstige						
LED QUITTIERUNG		RESET	E			
	INFSPE NAH					
2. Meßwerte						
STROM IL1		A				Wandlerdaten: 400/5 A
	INFSPE NAH					Bildskala: 0-600 A
<i>FEHLERORT REAKTANZ</i>		<i>OHM</i>	<i>E</i>			
	<i>INFSPE NAH</i>					

Informationen			E	V	Kontakt	Bemerkung
	Verknüpfung					
	<i>FEHLERORT DISTANZ</i>	<i>KM</i>	<i>E</i>			
	<i>INFSPE NAH</i>					
3. Steuerung						
Leistungsschalter (LS) Q0						
	LS BEFEHL NAHSTEU	BF AUS	E			Befehl (BF): Nah -> Ort
	INFSPE NAH					
	F-O/F RM FERN					
	LS BF AUS					
	LS BEFEHL NAHSTEU	BF EIN	E			
	INFSPE NAH					
	F-O/F RM FERN					
	LS BF EIN					
	LS BEFEHL ORTSTEU	RM AUS	E			Ortssteuerung (ORTSTEU)

Informationen			E	V	Kontakt	Bemerkung
	Verknüpfung					
	INFSPE NAH					
	LS BF AUS					
LS BEFEHL ORTSTEU		RM EIN	E			
	INFSPE NAH					
	LS BF EIN					
LS		BF AUS			K1	Befehl: Ort -> LS
LS		BF EIN			K2	
LS		RM STÖR	E		E1+E2	Rückmeldung
	INFSPE NAH					
LS		RM AUS	E		E2	
	INFSPE NAH					
LS		RM EIN	E		E1	
	INFSPE NAH					
LS NACHFÜHRUNG		NF AUS	E			Nachführung (NF)

Informationen			E	V	Kontakt	Bemerkung
	Verknüpfung					
LS NACHFÜHRUNG		NF EIN	E			
LS FALL			E	Q	(M4)	
	INFSPE NAH					
	Für diese Meldung ist nach den VDEW-Empfehlungen eine parallele Verdrahtung vorgesehen.					
SS- Trenner (SS-T) Q1						
SS-T		RM STÖR	E		E3+E4	
	INFSPE NAH					
SS-T		RM AUS	E		E4	
	INFSPE NAH					
SS-T		RM EIN	E		E3	
	INFSPE NAH					
SS-T NACHFÜHRUNG		NF AUS	E			
SS-T NACHFÜHRUNG		NF EIN	E			

Informationen			E	V	Kontakt	Bemerkung
	Verknüpfung					
Abgangserder (AE) Q8						
AE		RM STÖR	E		E5+E6	
	INFSPE NAH					
AE		RM AUS	E		E6	
	INFSPE NAH					
AE		RM EIN	E		E5	
	INFSPE NAH					
AE NACHFÜHRUNG		NF AUS	E			
AE NACHFÜHRUNG		NF EIN	E			
Felderder 1						
NACHFÜHRUNG ERDGARN.		GESETZT	E			
NACHFÜHRUNG ERDGARN.		GELÖSCHT	E			

Informationen			E	V	Kontakt	Bemerkung
	Verknüpfung					
AWE- Funktion						
<p><i>Da die AWE- Funktion innerhalb des Kombigerätes blockiert wird, sind hier auch nicht die gesamten Informationen eines externen Schalters erforderlich. Im Gegensatz zum Setzen der Infosperre wird hier jedoch nach Befehl und Rückmeldung unterschieden, um die Befehlsherkunft (Ort / Nah) zu protokollieren.</i></p>						
AWE-FUNKT BEFEHL NAHSTEU		BF AUS	E			
	INFSPE NAH					
	F-O/F RM FERN					
AWE-FUNKT BEFEHL NAHSTEU		BF EIN	E			
	INFSPE NAH					
	F-O/F RM FERN					
AWE-FUNKT		RM AUS	E			
	INFSPE NAH					
AWE-FUNKT		RM EIN	E			
	INFSPE NAH					

Informationen			E	V	Kontakt	Bemerkung
	Verknüpfung					
4. Schutz						
UMZ-Schutz						
UEST ANR N		KOM / GEH	E			Überstromanregung (UEST ANR) Neutralleiter (N)
	INFSPE NAH					
UEST ANR G		KOM / GEH	W	A	(M3)	Generalanregung (G)
	INFSPE NAH					
	Für diese Meldung ist nach den VDEW Empfehlungen eine parallele Verdrahtung vorgesehen.					
UEST ANR L1		KOM / GEH	E			
	INFSPE NAH					
UEST ANR L2		KOM / GEH	E			
	INFSPE NAH					
UEST ANR L3		KOM / GEH	E			

Informationen			E	V	Kontakt	Bemerkung
	Verknüpfung					
	INFSPE NAH					
UEST AUS			W			Schutzauslösung (AUS)
	INFSPE NAH					
Distanzschutz						
<i>DIST ANR N</i>		<i>KOM / GEH</i>	<i>E</i>			<i>Distanz Anregung (DIST ANR)</i>
	<i>INFSPE NAH</i>					
<i>DIST ANR G</i>		<i>KOM / GEH</i>	<i>W</i>	<i>A</i>	<i>(M3)</i>	<i>Generalanregung</i>
	<i>INFSPE NAH</i>					
	<i>Für diese Meldung ist nach den VDEW-Empfehlungen eine parallele Verdrahtung vorgesehen.</i>					
<i>DIST ANR L1</i>		<i>KOM / GEH</i>	<i>E</i>			
	<i>INFSPE NAH</i>					
<i>DIST ANR L2</i>		<i>KOM / GEH</i>	<i>E</i>			
	<i>INFSPE NAH</i>					

Informationen			E	V	Kontakt	Bemerkung
	Verknüpfung					
<i>DIST ANR L3</i>		<i>KOM / GEH</i>	<i>E</i>			
	<i>INFSPE NAH</i>					
<i>DIST AUS</i>			<i>W</i>	<i>A</i>		<i>Schutzauslösung</i>
	<i>INFSPE NAH</i>					
Sonstige Schutzinformationen						
<i>ERDS RIC RÜCKWÄRTS</i>			<i>W</i>	<i>A</i>		<i>Erdschlußrichtung (ERDS)</i>
	<i>INFSPE NAH</i>					
<i>ERDS RIC VORWÄRTS</i>			<i>W</i>	<i>A</i>		
	<i>INFSPE NAH</i>					
<i>WDL-SPG FEH SCHUTZ</i>		<i>KOM / GEH</i>	<i>W</i>	<i>A</i>	<i>E12</i>	<i>bei Dist-Schutz zwingend, Wandler (WDL)</i>
	<i>INFSPE NAH</i>					
<i>UEST NOTBETR</i>		<i>KOM / GEH</i>	<i>E</i>			<i>heißt heute DIST UMZ NOTBETR (Notbetrieb)</i>
	<i>INFSPE NAH</i>					<i>UEST wird aktiv, wenn DIST gestört</i>

Informationen			E	V	Kontakt	Bemerkung
	Verknüpfung					
	<i>UEST NOTBETR ANR G</i>	<i>KOM / GEH</i>	<i>W</i>			<i>Überstrom-Notbetrieb Generalanregung</i>
	<i>INFSPE NAH</i>					
	<i>Automatische Wiedereinschaltung</i>					
	<i>AWE SPE</i>	<i>KOM / GEH</i>	<i>W</i>	<i>A</i>		<i>AWE im Feld nicht möglich, Sperre (SPE)</i>
	<i>INFSPE NAH</i>					
	<i>AWE</i>		<i>E</i>			<i>AWE war erfolgreich</i>
	<i>INFSPE NAH</i>					
	5. Zentrale Funktionen					
	<i>E-WISCH BEFEHL NAHSTEU</i>	<i>RESET</i>	<i>E</i>			
	<i>INFSPE NAH</i>					

18 Anhang C: Modellierung IEC 61850

18.1 Abstraktes Datenmodell nach IEC 61850

LN Object	Data Object Data Set Rpt Control	CDC	Data Attr.	Value	AllInfo	GI	Explanation	Relationship to OCIS
Common Data Class Conformance - identical in all OCIS-Servers:								
		SPS					Single Point Status	
			stVal				Status Value	
			q				Quality	
			t				Timestamp	
		MV					Measured Value	
			mVal				Measured value	
			q				Quality	
			t				Timestamp	
		SPC					Controllable Single Point	
			ctlVal				Control value	für Befehlsrichtung
			stVal				Status value	für Melderichtung
			q				Quality	
			t				Timestamp	
			orCat				Originator category	

LN Object	Data Object Data Set Rpt Control	CDC	Data Attr.	Value	AllInfo	GI	Explanation	Relationship to OCIS
Physical Device = "REF 542"							ABB Combined Bay Unit	Steuerung und Schutz von
Server = "KO-FLG-K01"								10kV Abgangsfeld K01
Logical Device = "KO-FLG-K01"								
LLN01							Logical Node Zero	
"	AllInfo				X		Data Set for all reports	für alle Meldungen
"	RCAII		rptEna	TRUE			Report Control for all reports	
"	"		rptID	NULL				
"	"		optFlds	<00001>			only reason-for-inclusion	
"	"		datSetNam	"AllInfo"				
"	"		gl	NULL				
"	"		bufTim	NULL				
"	"		trgs	NULL				
"	"		trgOps	<110>			data- / quality-change	bei Wert-/Qualitätsänderung
"	"		rBEPd	NULL				
"	"		intgPd	NULL				
"	"		seqNum				Sequence number	
"	GI					X	Data Set for GI	Generalabfrage
"	RCGI		rptEna	TRUE			Report Control for GI	
"	"		rptID	NULL				
"	"		optFlds	<00001>			only reason-for-inclusion	

LN Object	Data Object Data Set Rpt Control	CDC	Data Attr.	Value	AllInfo	GI	Explanation	Relationship to OCIS
"	"		datSetNam	"GI"				
"	"		gl				Start GI	Generalabfrage starten
"	"		bufTim	NULL				
"	"		trgs	NULL				
"	"		trgOps	<001>			general-interrogation	nur bei Generalabfrage
"	"		rBEPd	NULL				
"	"		intgPd	NULL				
"	"		seqNum	0				
"	LED	SPC	stVal	NULL			LED reset	LED Quittierung Reset
"	IEBI	SPC			x	x	Blocking info exchange	Infspe Nah Kom/Geh
"	Ch	ISC			x	x	Activate characteristic	zu 7.3 Fernaktivierung
"	SupAl	SPS			x	x	Supervision alarm	zu 1.3 Überprüf. Fkt-Blöcke
"	DisDS	SPS			x	x	LN disturbance	zu 1.3 Überprüf. Fkt-Blöcke
"	LocDS	SPS			x	x	Local operation (not remote)	F-O/F RM Fern/Ort
"	PaSet	SPS			x	x	Parameter setting running	zu 7. Parametrierung
"	GI	SPS			x		General interrogation running	
PBRO1							Basic relay object	Schutz-Sammelmeldungen
"	GTr	SPS/T			x		General trip	Uest Aus
"	GAI	SPS			x	x	General alarm	Uest Anr G Kom/Geh
PDEF1							Directional earth fault	Erdschlußrichtungserfassung
"	FoEF	SPS			x	x	Foreward earthfault	Erds Ric Vorwärts
"	ReEF	SPS			x	x	Reverse earthfault	Erds Ric Rückwärts

LN Object	Data Object Data Set Rpt Control	CDC	Data Attr.	Value	AllInfo	GI	Explanation	Relationship to OCIS
PTOC1							Time overcurrent	UMZ-Schutz
"	SupA	SPS			x	x	Measurand supervision I	zu 1.3 Überprüf. Fkt-Blöcke
"	TrPTOC	SPS/T			x		Trip I>	
"	TrPIOC	SPS/T			x		Trip I>>	
"	AIA	SPS			x	x	Alarm L1	Uest Anr L1 Kom/Geh
"	AIB	SPS			x	x	Alarm L2	Uest Anr L2 Kom/Geh
"	AIC	SPS			x	x	Alarm L3	Uest Anr L3 Kom/Geh
"	AIN	SPS			x	x	Alarm N	Uest Anr N Kom/Geh
RREC1							Autorecloser	AWE-Einrichtung
"	EnaFct	SPC			x	x	LN on (not off)	AWE Ein/Aus
"	ARNR	SPS			x	x	Autorecloser not ready	AWE Spe Kom/Geh
"	ARSuc	SPS/T			x		AR cycle successful	AWE
RDRE1							Disturbance recorder	z.Zt nicht vorgesehen
"	DisDS	SPS			x	x	LN disturbance	"
GGIO1								Binär-Ein/Ausgänge
"	PwrSupAlm1	SPS			x	x	External power sup. alarm	220V GS1 Feh Kom/Geh
"	PwrSupAlm2	"			x	x	"	220V GS2 Feh Kom/Geh
"	PwrSupAlm3	"			x	x	"	220V Mot-Spg Feh Kom/Geh
IARC1							Archiving	z.Zt nicht vorgesehen
"	DisDS	SPS			x	x	LN disturbance	"
MMXU1							Measurand unit	Messwertverarbeitung
"	APhsARMS	MV			x	x	IL1 (RMS)	Strom IL1

LN Object	Data Object Data Set Rpt Control	CDC	Data Attr.	Value	AllInfo	GI	Explanation	Relationship to OCIS
CALH1							Alarm handling	z.Zt nicht vorgesehen
"	GrWr1	SPS			x	x	Group warning	"
"	GrAl1	SPS			x	x	Group alarm	"
Q0CSWI1							Switch controller	Leistungsschalter-Steuerung
"	Pos	DPC			x	x	Position	LS Bef/RM Ein/Aus/Stör, Fall
"	"		ctlModel	3 *)			direct-enhanced *) Profibus	beim Profibus Mapping
			ctlModel	4 **)			SBO-enhanced **) MMS	beim MMS Mapping
Q1CSWI2							Switch controller	Trennschalter-Erfassung
"	Pos	DPC	ctlVal	NULL	x	x	Position	SS-T Ein/Aus/Stör
"	"		ctlModel	0			status-only	handbetätigt
Q8CSWI3							Switch controller	Erder-Erfassung
"	Pos	DPC	ctlVal	NULL	x	x	Position	AE Ein/Aus/Stör
"	"		ctlModel	0			status-only	handbetätigt
Physical Device = "7SJ63"							Siemens Combined Bay Unit	Steuerung und Schutz von
Server = "KO-FLG-K02"								10kV Abgangsfeld K02
Logical Device = "KO-FLG-K02"								
<i>all identical to KO-FLG-K01</i>								
Physical Device = "PS 982"							Alstom Combined Bay Unit	Steuerung und Schutz von
Server = "KO-FLG-K03"								10kV Abgangsfeld K03

LN Object	Data Object Data Set Rpt Control	CDC	Data Attr.	Value	AllInfo	GI	Explanation	Relationship to OCIS
Logical Device = "KO-FLG-K03"								
<i>all identical to KO-FLG-K01</i>								
Physical Device = "..."							only example - not realized	Steuerung von 10kV
Server = "FLG-K10"								Erdungs- und Messfeld K10
Logical Device = "FLG-K10"								
... ..								
Q16CSWI1							Switch Controller	Erdungsschalter-Steuerung
"	Pos	DPC					Position	
... ..								
Physical Device = "..."							only example - not realized	Steuerung von 10kV
Server = "Station-Control"								Erdungs- und Messfeld K10
Logical Device = "Station-Control"								
... ..								
Q16CILO1							Interlocking	für Anlagenverriegelung
"	EnaClose	SPS					Enable close	verriegelt einschalten möglich

18.2 Mapping Datenmodell IEC 61850 auf MMS

Abstract Object Model							MMS Object Model				
LN Object	Data Object Data Set Rpt Control	CDC	Data Attr.	Value	AllInfo	GI	Named Variable 1.	2.	(Nesting Level) 3.	4.	5.
	Physical Device = "REF 542"										
	Server = "KO-FLG-K01"						VMD = "KO-FLG-K01"				
	Logical Device = "KO-FLG-K01"						Domain = "KO-FLG-K01"				
	LLN01						LLN01				
	"	AllInfo			X		"	\$AllInfo (Named Variable List)			
									1. LLN01\$ST\$IEBI		
									2. LLN01\$ST\$Ch		
									3. LLN01\$ST\$SupAI		
									...		
							"	\$RC			
	"	RCAII					"	"	\$RCAII		
	"	"	rptEna	TRUE			"	"	"	\$rptID	
	"	"	rptID	NULL							

Abstract Object Model							MMS Object Model				
LN Object	Data Object Data Set Rpt Control	CDC	Data Attr.	Value	AllInfo	GI	Named Variable 1.	2.	(Nesting Level) 3.	4.	5.
"	"		optFlds	<00001>			"	"	"	\$optFlds	
"	"		datSetNam	"AllInfo"			"	"	"	\$datSetNam	
"	"		gl	NULL							
"	"		bufTim	NULL							
"	"		trgs	NULL							
"	"		trgOps	<110>			"	"	"	\$trgOps	
"	"		rBEPd	NULL							
"	"		intgPd	NULL							
"	"		seqNum								
"	GI					X	"	\$GI (Named Variable List)			
									1. LLN01\$ST\$IEBI		
									2. LLN01\$ST\$Ch		
									3. LLN01\$ST\$SupAI		
									...		
	RCGI						"	\$RC	\$RCGI		
"	"		rptEna	TRUE			"	"	"	\$rptID	
"	"		rptID	NULL							
"	"		optFlds	<00001>			"	"	"	\$optFlds	
"	"		datSetNam	"GI"			"	"	"	\$datSetNam	
"	"		gl				"	"	"	\$gl	

Abstract Object Model							MMS Object Model				
LN Object	Data Object Data Set Rpt Control	CDC	Data Attr.	Value	AllInfo	GI	Named Variable 1.	2.	(Nesting Level) 3.	4.	5.
"	"		bufTim	NULL							
"	"		trgs	NULL							
"	"		trgOps	<001>			"	"	"	\$trgOps	
"	"		rBEPd	NULL							
"	"		intgPd	NULL							
"	"		seqNum	0							
"	LED	SPC	stVal	NULL			see below				
"	IEBI	SPC			x	x	see below				
"	Ch	ISC			x	x	see below				
"	SupAI	SPS			x	x	see below				
"	DisDS	SPS			x	x	see below				
"	LocDS	SPS			x	x	see below				
"	PaSet	SPS			x	x	see below				
"	GI	SPS			x		see below				
PBRO1							see below				
"	GTr	SPS/T			x		see below				
"	GAI	SPS			x	x	see below				
PDEF1							see below				
"	FoEF	SPS			x	x	see below				

Abstract Object Model							MMS Object Model				
LN Object	Data Object Data Set Rpt Control	CDC	Data Attr.	Value	AllInfo	GI	Named Variable 1.	2.	(Nesting Level) 3.	4.	5.
"	ReEF	SPS			x	x	see below				
PTOC1							see below				
"	SupA	SPS			x	x	see below				
"	TrPTOC	SPS/T			x		see below				
"	TrPIOC	SPS/T			x		see below				
"	AIA	SPS			x	x	see below				
"	AIB	SPS			x	x	see below				
"	AIC	SPS			x	x	see below				
"	AIN	SPS			x	x	see below				
RREC1							RREC1				
							"	\$ST			
"	EnaFct	SPC			x	x	"	"	\$EnaFct		
"	"		stVal				"	"	"	\$stVal	
"	"		q				"	"	"	\$q	
"	"		t				"	"	"	\$t	
"	"		orCat				"	"	"	\$orCat	
"	"		orAddr				"	"	"	\$orAddr	
"	"		ctlNum				"	"	"	\$ctlNum	
							"	\$CO			

Abstract Object Model							MMS Object Model				
LN Object	Data Object Data Set Rpt Control	CDC	Data Attr.	Value	AllInfo	GI	Named Variable 1.	2.	(Nesting Level) 3.	4.	5.
							"	"	"	\$Oper	
"	"		ctlVal				"	"	"	"	\$ctlVal
"	"		orCat				"	"	"	"	\$orCat
"	"		orAddr				"	"	"	"	\$orAddr
"	"		ctlNum				"	"	"	"	\$ctlNum
							"	"	"	"	\$T
							"	"	"	"	\$Test
							"	"	"	\$AppIError	
							"	"	"	"	\$InvokeID
							"	"	"	"	\$AddCause
							"	\$CF	"		
							"	"	\$Pos		
"	"		ctlModel	0			"	"	"	\$ctlModel	
"	ARNR	SPS			x	x	"	\$ST	\$ARNR		
"	"		stVal				"	"	"	\$stVal	
"	"		q				"	"	"	\$q	
"	"		t				"	"	"	\$t	
"	ARSuc	SPS/T			x		"	\$ST	\$ArSuc		
"	"		stVal				"	"	"	\$stVal	
"	"		q				"	"	"	\$q	

Abstract Object Model							MMS Object Model				
LN Object	Data Object Data Set Rpt Control	CDC	Data Attr.	Value	AllInfo	GI	Named Variable 1.	2.	(Nesting Level) 3.	4.	5.
"	"		t				"	"	"	\$t	
RDRE1							RDRE1				
							"	\$ST			
"	DisDS	SPS			x	x	"	"	\$DisDS		
"	"		stVal				"	"	"	\$stVal	
"	"		q				"	"	"	\$q	
"	"		t				"	"	"	\$t	
GGIO1							GGIO1				
							"	\$ST			
"	PwrSupAlm 1	SPS			x	x	"	"	\$PwrSupAlm1		
"	"		stVal				"	"	"	\$stVal	
"	"		q				"	"	"	\$q	
"	"		t				"	"	"	\$t	
"	PwrSupAlm 2	"			x	x	"	\$ST	\$PwrSupAlm2		
"	"		stVal				"	"	"	\$stVal	
"	"		q				"	"	"	\$q	
"	"		t				"	"	"	\$t	
"	PwrSupAlm	"			x	x	"	\$ST	\$PwrSupAlm3		

Abstract Object Model							MMS Object Model				
LN Object	Data Object Data Set Rpt Control	CDC	Data Attr.	Value	AllInfo	GI	Named Variable 1.	2.	(Nesting Level) 3.	4.	5.
	3										
"	"		stVal				"	"	"	\$stVal	
"	"		q				"	"	"	\$q	
"	"		t				"	"	"	\$t	
IARC1							IARC1				
							"	\$ST			
"	DisDS	SPS			x	x	"	"	\$DisDS		
"	"		stVal				"	"	"	\$stVal	
"	"		q				"	"	"	\$q	
"	"		t				"	"	"	\$t	
MMXU1							MMXU1				
							"	\$MX			
"	APhsARMS	MV			x	x	"	"	\$APhsARMS		
"	"		stVal				"	"	"	\$mVal	
"	"		q				"	"	"	\$q	
"	"		t				"	"	"	\$t	
CALH1							CALH1				
							"	\$ST			
"	GrWr1	SPS			x	x	"	"	\$GrWr1		

Abstract Object Model						MMS Object Model					
LN Object	Data Object Data Set Rpt Control	CDC	Data Attr.	Value	AllInfo	GI	Named Variable 1.	2.	(Nesting Level) 3.	4.	5.
"	"		stVal				"	"	"	\$stVal	
"	"		q				"	"	"	\$q	
"	"		t				"	"	"	\$t	
"	GrAl1	SPS			x	x	"	\$ST	\$GrAl1		
"	"		stVal				"	"	"	\$stVal	
"	"		q				"	"	"	\$q	
"	"		t				"	"	"	\$t	
Q0CSWI1							Q0CSWI1				
							"	\$ST			
"	Pos	DPC			x	x	"	"	\$Pos		
"	"		stVal				"	"	"	\$stVal	
"	"		q				"	"	"	\$q	
"	"		t				"	"	"	\$t	
"	"		orCat				"	"	"	\$orCat	
"	"		orAddr				"	"	"	\$orAddr	
"	"		ctlNum				"	"	"	\$ctlNum	
							"	\$CO			
							"	"	\$Pos		
							"	"	"	\$SelVal	

Abstract Object Model							MMS Object Model				
LN Object	Data Object Data Set Rpt Control	CDC	Data Attr.	Value	AllInfo	GI	Named Variable 1.	2.	(Nesting Level) 3.	4.	5.
"	"		ctlVal				"	"	"	"	\$ctlVal
"	"		orCat				"	"	"	"	\$orCat
"	"		orAddr				"	"	"	"	\$orAddr
"	"		ctlNum				"	"	"	"	\$ctlNum
							"	"	"	"	\$T
							"	"	"	"	\$Test
							"	"	"	\$Cancel	
"	"		orCat				"	"	"	"	\$orCat
"	"		orAddr				"	"	"	"	\$orAddr
"	"		ctlNum				"	"	"	"	\$ctlNum
							"	"	"	"	\$T
							"	"	"	"	\$Test
							"	"	"	\$Oper	
"	"		ctlVal				"	"	"	"	\$ctlVal
"	"		orCat				"	"	"	"	\$orCat
"	"		orAddr				"	"	"	"	\$orAddr
"	"		ctlNum				"	"	"	"	\$ctlNum
							"	"	"	"	\$T
							"	"	"	"	\$Test
							"	"	"	\$CmdTerm	

Abstract Object Model							MMS Object Model				
LN Object	Data Object Data Set Rpt Control	CDC	Data Attr.	Value	AllInfo	GI	Named Variable 1.	2.	(Nesting Level) 3.	4.	5.
							"	"	"	"	\$Success
							"	"	"	"	\$InvokeID
"	"		orCat				"	"	"	"	\$orCat
"	"		orAddr				"	"	"	"	\$orAddr
"	"		ctlNum				"	"	"	"	\$ctlNum
							"	"	"	"	\$T
							"	"	"	"	\$Test
							"	"	"	\$AppIError	
							"	"	"	"	\$InvokeID
							"	"	"	"	\$AddCause
							"	\$CF	"		
							"	"	\$Pos		
"	"		ctlModel	4			"	"	"	\$ctlModel	
Q1CSWI2							Q1CSWI2				
							"	\$ST			
"	Pos	DPC			x	x	"	"	\$Pos		
"	"		stVal				"	"	"	\$stVal	
"	"		q				"	"	"	\$q	
"	"		t				"	"	"	\$t	
"	"		orCat				"	"	"	\$orCat	

Abstract Object Model							MMS Object Model				
LN Object	Data Object Data Set Rpt Control	CDC	Data Attr.	Value	AllInfo	GI	Named Variable 1.	2.	(Nesting Level) 3.	4.	5.
	Server = "KO-FLG-K02"						VMD = "KO-FLG-K02"				
	Logical Device = "KO-FLG-K02"						Domain = "KO-FLG-K02"				
	<i>all identical to KO-FLG-K01</i>										
	Physical Device = "PS982"										
	Server = "KO-FLG-K03"						VMD = "KO-FLG-K03"				
	Logical Device = "KO-FLG-K03"						Domain = "KO-FLG-K03"				
	<i>all identical to KO-FLG-K01</i>										
	Physical Device = "..."										
	Server = "FLG-K10"						VMD = "KO-FLG-K10"				
	Logical Device =						Domain = "KO-FLG-				

Abstract Object Model							Profibus User Data Block																
LN Object	Data Object Data Set Rpt Control	CDC	Data Attr.	Value	AllInfo	GI	IOA	SI	T	OC	QD	AD	CDC	TT	I1	I2	V	TT	TAI1	TAI2			
"REF 542"																							
Server = "KO-FLG-K01"																							
Logical Device = "KO-FLG-K01"																							
LLN01							not visible																
"	AllInfo				X		not visible																
"	RCAI1		rptEna	TRUE			not visible																
"	"		rptID	NULL			not visible																
"	"		optFlds	<00001>			not visible																
"	"		datSetName	"AllInfo"			not visible																
"	"		gl	NULL			not visible																
"	"		bufTim	NULL			not visible																
"	"		trgs	NULL			not visible																
"	"		trgOps	<110>			not visible																
"	"		rBEPd	NULL			not visible																
"	"		intgPd	NULL			not visible																
"	"		seqNum	0			not visible																
"	GI					X	not visible																

18.4 Abstraktes Dienstmodell nach IEC 61850

Service	Service Param.	Value	Explanation	Relationship to OCIS
---------	----------------	-------	-------------	----------------------

Only Services for the normal substation operation are shown.

All "Browser-Services" are not shown here.

SetReportControlValue Request				
	ListOf...Names	LLN01.RCGI.gi		
	ListOf...Values	TRUE	request general interrogation	Befehl GA starten
SetReportControlValue Response+				
	Result			
SetReportControlValue Response-				
Report Request				alle Meldungen u. Messwerte
	rptID	"LLN01.RCGI" "LLN01.RCAII"	for general interrogation for all other reports	
	optFIds	<01001>	only reason-for-inclusion	
	seqNum		Sequence number	
	subSeqNum	NULL	Subsequence number	
	datNam		Data object name	
	value			

Service	Service Param.	Value	Explanation	Relationship to OCIS
	reasonCode	0	periodic	bei Messwerten
		1	integrity	bei schleichender GA
		2	general-interrogation	bei Generalabfrage
		3	spontaneous-change	z.B. LS Fall, Messwerte
		4	commanded-change	nach allen erfolgreichen Bef.
		...		
Operate Request				alle Befehle (außer GA start.)
	ControlObj.Nam.		accessed data object	
	Value			
	T		Timestamp	
	Test	FALSE		
Operate Response+				
	ControlObj.Nam.		same as in Request	
	Value		"	
	T		"	
	Test	FALSE	"	
Operate Response-				Befehls-Fehlermeldungen
	ControlObj.Nam.		same as in Request	
	Value		"	
	T		"	
	Test	FALSE	"	
	AddCause			Fehlerursache
CommandTermination Request+				

Service	Service Param.	Value	Explanation	Relationship to OCIS
	ControlObj.Nam.	Q0CSWI1.Pos	same as in Operate Request	
	Value		"	
	T		Timestamp	
	Test	FALSE	same as in Operate Request	
CommandTermination Request-				Befehls-Fehlermeldungen
	ControlObj.Nam.	Q0CSWI1.Pos	same as in Operate Request	
	Value		"	
	T		Timestamp	
	Test	FALSE	same as in Operate Request	
	AddCause			Fehlerursache
SelectWithValue Request	*)			Anwahl (2-stufiges Schalten)
	ControlObj.Nam.	Q0CSWI1.Pos		
	Value			
	T		Timestamp	
	Test	FALSE		
SelectWithValue Response+	*)			
	ControlObj.Nam.	Q0CSWI1.Pos	same as in Request	
	Value		"	
	T		"	
	Test	FALSE	"	
SelectWithValue Response-	*)			Anwahl-Fehlermeldungen

Service	Service Param.	Value	Explanation	Relationship to OCIS
	ControlObj.Nam.	Q0CSWI1.Pos	same as in Request	
	Value		"	
	T		"	
	Test	FALSE	"	
	AddCause			Fehlerursache
Cancel Request				Anwahl-Abbruch
	ControlObj.Nam.	Q0CSWI1.Pos		
	T		Timestamp	
	Test	FALSE		
Cancel Response+				
	ControlObj.Nam.	Q0CSWI1.Pos	same as in Request	
	T		"	
	Test	FALSE	"	
Cancel Response-				Abbruch-Fehlermeldungen
	ControlObj.Nam.	Q0CSWI1.Pos	same as in Request	
	T		"	
	Test	FALSE	"	
	AddCause			Fehlerursache
*) Only in MMS-Mapping "on the wire". In Profibus Mapping local issue of the Client at Station-Level.				
GetDataValue Request	**)			
	ListOf...Names	Q16CILO1.EnaClose		Check Anlagengerriegelung Q16

Service	Service Param.	Value	Explanation	Relationship to OCIS
GetDataValue Response+	**)			
	ListOf...Names	Q16CILO1.EnaClose		Erder ein erlaubt/nicht erlaubt
	ListOf...Values			
GetDataValue Response-	**)			Kommunikations-Fehler
**) Only example for station-wide interlocking - actually not realized.				

18.5 Mapping Dienstmodell IEC 61850 auf MMS

ACSI Service			MMS Service
Service	Service Param.	Value	

Only Services for the normal substation operation are shown.

All "Browser-Services" are not shown here.

SetReportControlValue Request			Write Request
	ListOf...Names	LLN01.RCGI.gi	
	ListOf...Values	TRUE	
SetReportControlValue Response+			Write Response+
	Result		
SetReportControlValue Response-			Write Response-
Report Request			InformationReport (RPT) <i>transient Named Variable List</i>
			RPT
	rptID	"LLN01.RCGI" "LLN01.RCAII"	1. rptID
	optFlds	<01001>	2. optFlds = <000011> datNam(TRUE) inclBstr(FALSE)
	seqNum		3. seqNum

ACSI Service			MMS Service
Service	Service Param.	Value	
	subSeqNum	NULL	4. NULL
	datNam		5. {datNam
	value		value
	reasonCode		reasonCode}
Operate Request			Write Request (...\$CO\$...\$Oper)
	ControlObj.Nam.		ObjectName
	Value		Part of MMS Object Model
	T		"
	Test	FALSE	"
Operate Response+			Write Response+ (...\$CO\$...\$Oper)
	ControlObj.Nam.		not mapped (known by InvokeID)
	Value		"
	T		"
	Test	FALSE	"
Operate Response-			1. Write Response+ (...\$CO\$...\$Oper)
			failure = <11> "object-value-invalid"
	ControlObj.Nam.		not mapped (known by InvokeID)
	Value		"
	T		"
	Test	FALSE	"
			2. InformationReport (...\$CO\$...\$AppIError)
	AddCause		Part of MMS Object Model
CommandTermination			InformationReport (...\$CO\$...\$CmdTerm)

ACSI Service		MMS Service
Service	Service Param.	Value
Request+		
		...\$Success = TRUE
	ControlObj.Nam.	Q0CSWI1.Pos
	Value	"
	T	"
	Test	FALSE
CommandTermination Request-		InformationReport (...\$CO\$...\$CmdTerm)
		...\$Success = FALSE
	ControlObj.Nam.	Q0CSWI1.Pos
	Value	"
	T	"
	Test	FALSE
	AddCause	Part of MMS Object Model
SelectWithValue Request		Write Request (...\$CO\$...\$SelVal)
	ControlObj.Nam.	Q0CSWI1.Pos
	Value	ObjectName = Q0CSWI1\$CO\$Pos\$SelVal
	T	"
	Test	FALSE
SelectWithValue Response+		Write Response+ (...\$CO\$...\$SelVal)
	ControlObj.Nam.	Q0CSWI1.Pos
	Value	"

ACSI Service			MMS Service
Service	Service Param.	Value	
	T		"
	Test	FALSE	"
SelectWithValue Response-			1. Write Response+ (...\$CO\$...\$SelVal)
			failure = <11> "object-value-invalid"
	ControlObj.Nam.	Q0CSWI1.Pos	not mapped (known by InvokeID)
	Value		"
	T		"
	Test	FALSE	"
			2. InformationReport (...\$CO\$...\$ApplError)
	AddCause		Part of MMS Object Model
Cancel Request			Write Request (...\$CO\$...\$Cancel)
	ControlObj.Nam.	Q0CSWI1.Pos	ObjectName = Q0CSWI1\$CO\$Pos\$SelVal
	T		Part of MMS Object Model
	Test	FALSE	"
Cancel Response+			Write Response+ (...\$CO\$...\$Cancel)
	ControlObj.Nam.	Q0CSWI1.Pos	not mapped (known by InvokeID)
	T		"
	Test	FALSE	"
Cancel Response-			1. Write Response+ (...\$CO\$...\$Cancel)
			failure = <11> "object-value-invalid"
	ControlObj.Nam.	Q0CSWI1.Pos	not mapped (known by InvokeID)
	T		"

ACSI Service			MMS Service
Service	Service Param.	Value	
	Test	FALSE	"
	AddCause		2. InformationReport (...\$CO\$...\$ApplError)
			Part of MMS Object Model
GetDataValue Request	**)		Read Request
	ListOf...Names	Q16CILO1.EnaCl ose	
GetDataValue Response+	**)		Read Response+
	ListOf...Names	Q16CILO1.EnaCl ose	
	ListOf...Values		
GetDataValue Response-	**)		Read Response-
**) Only example for station-wide interlocking - actually not realized.			

19 Anhang D: Modellierung GOMSFE

Datenmodell GOMSFE (V0.8) und Dienstmodell CASM (V1.4)

q (Quality) := BSTR16, Bit Number:

0 := Reserved

1 := Invalid

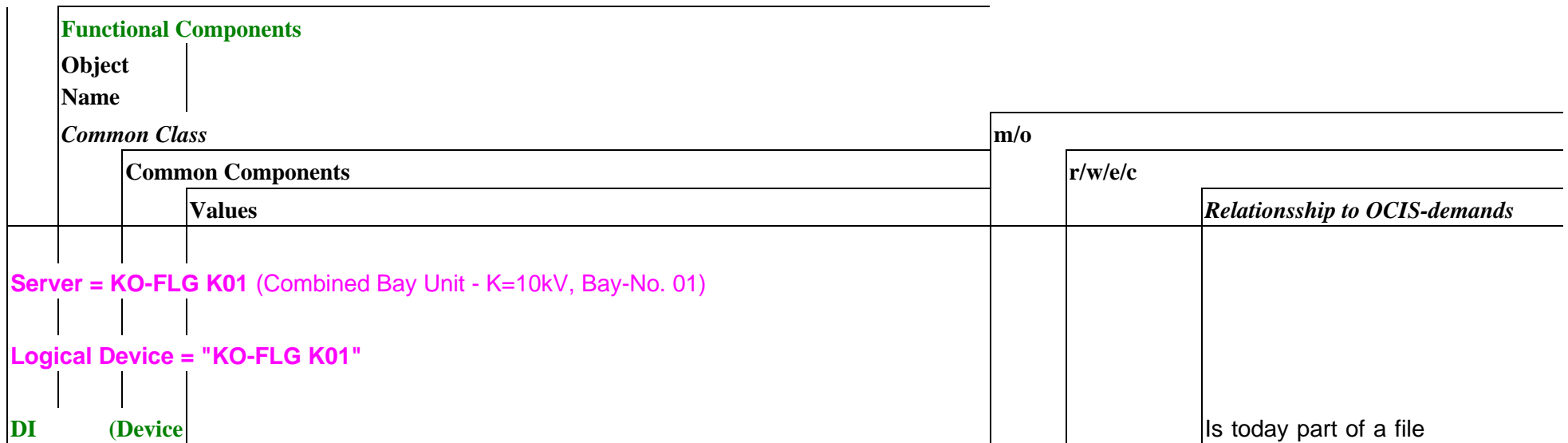
2 := Communications Failure

3 := Forced (not necessarily the actual value)

4 := Over Range

5 := Bad Reference

6-15 := Unassigned (future)



Functional Components				
Object Name				
<i>Common Class</i>		m/o		
	Common Components		r/w/e/c	
	Values			<i>Relationship to OCIS-demands</i>
Identity)				transfer.
	Name (OwnerDeviceName) := VSTR32 "BayUnit K01"	m	rw	What is it good for ? Only information ?
	Class (Device Class) := VSTR32 "Combined Bay Unit"	o	rw	Some automatic configurations ? Self Identification ?
	d (Description) := VSTR32 "Test"	o	rw	
	Own (Owner) := VSTR32 "OCIS-Team"	o	rw	
	Loc (DeviceLocation) := VSTR128 "Station OCIS Bay K01"	o	rw	
	VndID (VendorID) := STRUCT	m	r	
	Vnd (Vendor) := VSTR32 "ABB"	m	r	
	DevMdl (DeviceModel) := VSTR32 "REF 542"	o	rw	
	SerNum (SerialNumber) := VSTR32 "1998 007"	m	r	
	HwRev (Hardware Version) := VSTR8 "1.2"	o	r	
	SwtRev (Software Version) := VSTR8 "2.1"	o	rw	
	CommID (Communication ID) := STRUCT	o	rw	It was the intention of GOMSFE

Functional Components				
Object Name				
<i>Common Class</i>			m/o	
	Common Components			r/w/e/c
	Values			Relationship to OCIS-demands
	CommAdr (Communication Address) := VSTR16 "xyxy"	m	rw	that only devices that are connected to the communication are Logical Devices. It is possible in GOMSFE too, to add an index to the objects for differentiation. (Sw1, Sw2, Sw3,...)
	CommRev (Communication Revision ?) := VSTR8 "blabla"	o	r	
	Prot (Protocol) := ENUM8 11 - CASM+MMS Where defined ?	o	r	
	Med (Medium) := ENUM8 3 - Fiber Optic Where defined ?	o	r	
	MAC (Medium Access Control) := INT8U Where defined ?	o	rw	
7. Basic Building Bricks				
7.1 GLOBE				

Functional Components				
Object Name	Common Class	Common Components	Values	Relationship to OCIS-demands
Globe	ST (Status)			
	ModeDS (Mode Device State)			self-diagnoses
	SIT - Status Input Double Bit			
	b2 (defined by DevSt - Device Status) := BSTR2 <00> := In Test (Not defined under Common Components !?) <01> := Offline <10> := Available <11> := Unhealthy (Not defined under Common Components !?)			
	q (Quality) := BSTR16, Bit Number: <xxxxxx00 00000000>			Should this indicate alarm messages ? No differentiation between alarm

Functional Components				
Object Name	Common Class	m/o	r/w/e/c	Relationship to OCIS-demands
	Common Components			
	Values			
	t (Timestamp) := BTIME6	o		and disturbance is possible !
	LocRemDS (Local Remote Device State)	m	r	<i>local / remote switch</i>
	SIT - Status Input Double Bit			
	b2 (defined by DevSt - Device Status) := BSTR2	m		
	<01> := Local			
	<10> := Remote			
	q (Quality) := BSTR16, Bit Number:	o		
	<xxxxxx00 00000000>			
	t (Timestamp) := BTIME6 (6 Octets)	o		
	LEDDS (LED reset)			Not yet in GOMSFE
	SI - (Status Input single bit)			LED reset
	b (binary value) := BOOL			
	<1> := enabled			

Functional Components				
Object Name				
<i>Common Class</i>		m/o		
	Common Components		r/w/e/c	
	Values			<i>Relationship to OCIS-demands</i>
	q (Quality) := BSTR16, Bit Number: <xxxxxx00 00000000> t (Timestamp) := BTIME6 (6 Octets)			
	ActSG (Active Settings Group) INT8 U <X> := Settings Group No. X in use	o	r	<i>remote switch between different parameter-groups</i>
	EditSG (Edit Settings Group) INT8 U <X> := Settings Group No X selected for editing and visible for all models in Logical Device ?	o	r	Use unclear, do we need it ?
	AuxIn <1> (Block of 16 Auxiliary Inputs)	o	r	220V DC-1

Functional Components				
Object Name				
<i>Common Class</i>		<i>m/o</i>		
<i>Common Components</i>				<i>r/w/e/c</i>
<i>Values</i>				<i>Relationship to OCIS-demands</i>
	BSTR16 <00000000 00000001>			alarm BSTR16 must be changed to SI or "AuxIn <n>" must be changed to "AuxIn" Quality and Timestamp missing !
AuxIn <2> (Block of 16 Auxiliary Inputs)		o	r	220V DC-2 alarm
	BSTR16 <00000000 00000010>			
AuxIn <3> (Block of 16 Auxiliary Inputs)		o	r	220V DC-motor alarm
	BSTR16 <00000000 00000100>			
SP (Set Points and Settings Groups)				

Functional Components					
Object Name					
<i>Common Class</i>			m/o		
	Common Components			r/w/e/c	
	Values				<i>Relationship to OCIS-demands</i>
PreSetDNA	Preset Values for GOOSE-In-/Output-Messages		o	rw	Use unclear, do we need it ?
	DNA (Desoxyribonucleic Acid ??) := BSTR64 Bit: 0,1: Operate Device: Normal / Trip / Close 2,3: Lock Out : Normal / Lock Out 4,5: Initiate Reclosing: Normal /Cancel / Auto Reclosing 6,7: Block Reclosing: Normal / Cancel / Block ... 26, 27: Breaker Device Status: Between / Open / Closed ... 36, 37: Interlock Device Status: Non Interlock / Interlock ... 40,41: Mode: Test / Offline / Available / Unhealthy 42,43: Event: Normal / Event 44, 45: Fault Present: Clear / Present ...				

Functional Components				
Object Name				
<i>Common Class</i>		m/o		
	Common Components		r/w/e/c	
	Values			<i>Relationship to OCIS-demands</i>
ForDNA (forced values nowhere explained !?)		o	rw	Use unclear, do we need it ?
	DNA (Desoxyribonucleic Acid ??) := BSTR64 ?? ?			
DefDNA	Default Values for GOOSE-Input-Messages	o	rw	Use unclear, do we need it ?
	DNA (Desoxyribonucleic Acid ??) := BSTR64 ... see above ...			
CO (Control)				
ODLED (LED reset)				Not yet in GOMSFE
BO - Binary Output				LED reset
	b<n=1> (binary value) := BOOL			

Functional Components					
Object Name					
<i>Common Class</i>			m/o		
	Common Components			r/w/e/c	
	Values				<i>Relationship to OCIS-demands</i>
		<1> := reset			
	CopySetGrp (Copy Settings Group)		o	w	Use unclear, do we need it ?
	INT8 U	<X> := Copies Settings Group No X to visible buffer for editing			
	SaveSetGrp (Save Settings Group)		o	w	Use unclear, do we need it ?
	INT8 U	<X> := Saves edit buffer Settings Group No. X as designated Settings Group			
	ActSetGrp (Activate Settings Group)		o	w	<i>remote switch between different Parameter-Groups</i>
	INT8 U	<X> := Selects designated Settings Group No. X as the active one			

Functional Components				
Object Name				
<i>Common Class</i>		m/o		
	Common Components		r/w/e/c	
	Values			Relationship to OCIS-demands
CF (Configuration)				
Img (Image - Proprietary Information)		o	rw	remote transfer of the parameter-file
AcsBLOB - Access Binary Large Object				
Blob := BLOB <Data> ... <Data> ...		m		
State := BOOL <0> := deselected - access is inhibited <1> := selected		o		
SelTimeout (Select Timeout in seconds) := INT8U		o		What does this exactly means for file transfer ?
Fmt (Format) := INT8 8 := ISO Basic Encoding Rules		o		
DC				Use unclear, do we

Functional Components					
Object Name	Common Class	Common Components	m/o	r/w/e/c	Relationship to OCIS-demands
		Values			
(Description)					need it ?
AuxIn <1> (Block of 16 Auxiliary Inputs)		d (Analog Description) := VSTR32 "220V DC-1 alarm"	o m	rw	
AuxIn <2> (Block of 16 Auxiliary Inputs)		d (Analog Description) := VSTR32 "220V DC-2 alarm"	o m	rw	
AuxIn <3> (Block of 16 Auxiliary Inputs)		d (Analog Description) := VSTR32 "220V DC-motor alarm"	o m	rw	
AX (Access)					
SGEditor (Settings Group Editor)			o	rw	Controls access to editing capability

Functional Components					
Object Name					
<i>Common Class</i>			m/o		
	Common Components			r/w/e/c	
	Values				Relationship to OCIS-demands
Tag	TagID (tag number) := INT8U		m		Use unclear, do we need it ?
	TagType (Tag Type Permitted) := BSTR8, Bit Number: 0 := No Switching 1 := Operation Blocked 2 := Device Failure 3 := Interlock Active		m		
	TagD (Tag Description) := VSTR128 free text		m		
	TagOwn (Tag Owner who placed the tag) := VSTR32 free text		m		
RP					

Functional Components				
Object Name	Common Class	m/o	r/w/e/c	Relationship to OCIS-demands
	Common Components			
	Values			
(Reporting)				
brcbMX (Basic Report Control Block for Measurements)		o	rw	
BasRCB - Basic Report Control Block				
RptID (Report Identification) := IDENT	"NULL"	m		
RptEna (Report Enable) := BOOL	FALSE	m		<i>part of information blocking</i>
	TRUE			<i>All informations must be blocked, also commands !</i>
DatSetNa (Data Set Name) := IDENT	"LogDev.MX" (All Measurements of Logical Device)	m		
OptIFlds (Optional Fields to include) := BSTR8, Bit Number	0 - reserved	m		In CASM defined.
	1 - sequence number			
	2 - report time stamp			
	3 - reason for inclusion			
	4 - output data set name			
BufTim (Buffer Time until combined report) := INT32U (ms)		m		

Functional Components					
Object Name					
<i>Common Class</i>			m/o		
	Common Components			r/w/e/c	
	Values				<i>Relationship to OCIS-demands</i>
	0 - Buffer time attribute not used				
	Trgs (number of event notifications before report) := INT16U		m		
	0 - Trigger attribute not used				
	SeqNum (Sequence Number) := INT8U		m		
	0, 1, 2, 3, ... , 2 ³²				
	TrgOps (Trigger Options) := BSTR8, Bit Number:		m		Equal to bitstring "reason for inclusion".
	0 - reserved				
	1 - (CndDatChg) Change of any value, possibly with deadbanding				
	2 - (CndQuChg) Change of quality field				
	3 - (CndFrzChg) Change of frozen value				?
	default <0110xxxx>				
	RBEPd (Report Period) := INT32U (in milliseconds)		m		
	0 - immediately				
	IntgPd (Integrity Period) := INT32U (in milliseconds)		m		
	0 - immediately				
	brcbST (Basic Report Control Block for Status Points)		o	rw	
	BasRCB - Basic Report Control Block				

Functional Components				
Object Name				
<i>Common Class</i>		m/o		
	Common Components		r/w/e/c	
	Values			<i>Relationship to OCIS-demands</i>
	RptID (Report Identification) := IDENT "NULL"	m		
	RptEna (Report Enable) := BOOL FALSE TRUE	m		<i>part of information blocking</i>
	DatSetNa (Data Set Name) := IDENT "LogDev.ST" (All Status Points of Logical Device)	m		All informations must be blocked, also commands !
	OptIFlds (Optional Fields to include) := BSTR8, Bit Number 0 - reserved 1 - sequence number 2 - report time stamp 3 - reason for inclusion 4 - output data set name	m		In CASM defined. What about a "hidden" general interrogation? Only limited time is available, low priority !
	BufTim (Buffer Time until combined report) := INT32U (ms) 0 - Buffer time attribute not used	m		
	Trgs (number of event notifications before report) := INT16U	m		

Functional Components			
Object Name			
<i>Common Class</i>		m/o	
	Common Components		r/w/e/c
	Values		<i>Relationship to OCIS-demands</i>
	0 - Trigger attribute not used SeqNum (Sequence Number) := INT8U 0, 1, 2, 3, ... , 2 ³² TrgOps (Trigger Options) := BSTR8, Bit Number: 0 - reserved 1 - (CndDatChg) Change of any value, possibly with deadbanding 2 - (CndQuChg) Change of quality field 3 - (CndFrzChg) Change of frozen value default <0110xxxx> RBEPd (Report Period) := INT32U (in milliseconds) 0 - immediately IntgPd (Integrity Period) := INT32U (in milliseconds) 0 - immediately	 m m m m	 Equal to bitstring "reason for inclusion". ? Fulfills this the function of the general-interrogation ?
Information Blocking is still not possible in GOMSFE !			

Functional Components				
Object Name				
Common Class		m/o		
Common Components			r/w/e/c	
Values				<i>Relationship to OCIS-demands</i>
GOOSE (Generic Object Oriented Substation Event)		o	rw	Use unclear, do we need it ?
PACT - Protection Action				
SendingIED (Sending IED Identity) := IDENT				part of system start , newly activated IED
t (GOOSE timestamp) := BTIME6				will send initial GOOSE
SeqNum (message Sequence Number) := INT16U 0, 1, 2, 3, ... incremented by each sended message				
StNum (event State sequence Number) := INT16U 0, 1, 2, 3, ... incremented by each "new" or "changed" sended message				
HoldTim (Hold Time before cancellation) := INT16U e.g. 10, 20, 30, 40, ... 1 min.				
BackTim ((Back) Time since last status change) := INT16U				
PhsID (Identifies Faulted Phases) := INT16U				
DNA (Desoxyribonucleic Acid ??) := BSTR64 ... see above ...				
UserSt (User Status) := BSTR256 User defined Bitstring, used in bit pairs				

Functional Components						
Object Name	Common Class	Common Components	Values	m/o	r/w/e/c	Relationship to OCIS-demands
7.1.1 GLOBE Data Sets						Necessary for Reporting !
<p>There are only the data objects shown in this table that we need in our OCIS-process. The real combined bay unit offers much more informations, so inside this device there will be implemented much more objects that refer to all possible functions.</p> <p>So the following data set is a user defined data set for the OCIS-process user and only a subset of the objects inside the bay unit.</p> <p>Caution ! I think the definition of a data set is the task of the vendor and not a task of the customer.</p> <p>This data set must be defined after the combined bay unit is configured. E.g. the configuration of the auxiliary inputs must be defined before.</p> <p>The following objects of the data set are not combined, to allow the server only to transmit a single status change without transmitting the other status informations.</p>						

Functional Components				
Object Name	Common Class	Common Components	Values	Relationship to OCIS-demands
Data "LogDev.MX" 1. MU.MX.A	Set		A minimum inclusion bitstring BSTR1 is necessary.	
Data "LogDev.ST" 1. Globe.ST.Mode DS 2. Globe.ST.LocRemDS 3. Globe.ST.L EDDS 4. Globe.ST.ActSG 5. Globe.ST.EditSG	Set			

Functional Components			
Object Name	<i>Common Class</i>	<i>m/o</i>	<i>r/w/e/c</i>
<i>Common Components</i>	<i>Values</i>	<i>Relationship to OCIS-demands</i>	
6. Globe.ST.AuxIn <1>			
7. Globe.ST.AuxIn <2>			
8. Globe.ST.AuxIn <3>			
9. TOC.ST.O ut			
10. TOC.ST.Ta r			
11. DOCE.ST. FoEF			
12. DOCE.ST. ReEF			
13. RecR.ST.F ctDS			
14. RecR.ST.A RNR			
15. RecR.ST.A RSuc			

Functional Components						
Object Name	Common Class	Common Components	Values	m/o	r/w/e/c	Relationship to OCIS-demands
16. Sw(1).ST.S wDS						
17. Sw(8).ST.S wDS						
18. Sw(51).ST. SwDS						
19. CB.ST.Sw DS						
A minimum inclusion bitstring BSTR19 is necessary.						
LG (Logging)						
SOE (sequence of events)						<i>make the central logging</i>

not available for globe

Functional Components				
Object Name				
<i>Common Class</i>		m/o		
Common Components			r/w/e/c	
Values				<i>Relationship to OCIS-demands</i>
<p>LCB - Log Control Block</p> <p>LogEna (Logging Enalble) := BOOL</p> <p style="padding-left: 20px;">FALSE</p> <p style="padding-left: 20px;">TRUE</p> <p>LogEnr (ECB to Enroll) := IDENT</p> <p style="padding-left: 20px;">Must be the ECB that belongs to Globe, Protection-Status-Changes must be in consideration too.</p> <p>LogWrp (Wrap ?) := BOOL</p> <p style="padding-left: 20px;">TRUE</p> <p>WrnLev (Warn Level) := INT16U</p> <p>LogSize := INT32U</p> <p>OvrST (Overflow Status) := BOOL</p> <p>WrnST (Warn Status) := BOOL</p> <p style="padding-left: 20px;">FALSE</p> <p>UseST (Utilization Status) := BOOL</p> <p style="padding-left: 20px;">100 (%)</p> <p>OldTim (Oldest Entry Time) := BTIME6</p>				<p>topical after a communication disturbance</p>

Functional Components				
Object Name	Common Class	m/o	r/w/e/c	Relationship to OCIS-demands
	Common Components			
	Values			
	NewTim (Newest Entry Time) := BTIME6 OldEnt (Oldest Entry Number) := INT32U CurEnt (Number of Current Entries) := INT32U			
8. Measurement Unit				
In our sense this is not a Device Object Model but a Basic Building Brick ! So some mandatory objects make no sense in our example.				
MU (Measurement Unit)				
	MX (Measurements)			
A (Current)		o		IL1
WYE - ?				
	PhsAi (Phase A Integer) := INT16S	o		

Functional Components				
Object Name			m/o	
<i>Common Class</i>				r/w/e/c
	Common Components			
	Values			<i>Relationship to OCIS-demands</i>
	PhsAf (Phase A Floiting Point) := FLT32		o	
	q (Quality) := BSTR16 <xxxxxx00 00000000>		o	
	t (Timestamp) := BTIME6 (6 Octets)		o	
	There are further mandatory objects that we dont't need:			
	CF.CLock		m	
	TOD			
	DC.EqRtg		m	
	DC.ConCk		m	
	t			
	DC.A		m	
	RP.brcbM		m	
	X			
7. Basic Building Bricks				
7.3 Protection Functions				

Functional Components						
Object Name	Common Class	Common Components	Values	m/o	r/w/e/c	Relationship to OCIS-demands
TOC (Time Overcurrent)						
How to model "Block I>>" for reverse blocking ?						
ST (Status) - Operation Information						
Out	BOO L		<1> := trip	o	r	general trip Needed only for indication, use for trip-signal is possible too.

Functional Components					
Object Name	Common Class	Common Components	m/o	r/w/e/c	Relationship to OCIS-demands
	Values				
Tar (Targets since last reset)			o	r	
	<i>Common Class is confused with Common Coponent !</i>				
	PhsTar (Phase Targets) := ENUM8				
	0 - None				TOC alarm gone ?
	1 - Phase A				TOC alarm L1 (alone)
	2 - Phase B				TOC alarm L2 (alone)
	3 - Phase C				TOC alarm L3 (alone)
	4 - Ground only				TOC alarm N (alone ?)
	5 - A to ground				Difference to 1,2,3 ?
	6 - B to ground				
	7 - C to ground				
	8 - AB				<i>TOC alarm L1 & L2</i>
	9 - BC				<i>TOC alarm L2 & L3</i>
	10 - CA				<i>TOC alarm L3 & L1</i>

Functional Components					
Object Name					
<i>Common Class</i>			m/o		
	Common Components			r/w/e/c	
	Values				Relationship to OCIS-demands
		11 - AB to ground			TOC alarm L1 & L2 & N
		12 - BC to ground			TOC alarm L2 & L3 & N
		13 - CA to ground			TOC alarm L3 & L1 & N
		14 - ABC			TOC alarm L1 & L2 & L3
		15 - ABC to ground			TOC alarm L1 & L2 & L3 & N
		no corresponding object in GOMSFE possible derivation with use of targets ?!			TOC alarm General
		AS (Association)			
All IEDs	Related		m	rw	???
TBD - ???					Do we need it for reverse blocking ?

Functional Components					
Object Name	Common Class	Common Components	m/o	r/w/e/c	Relationship to OCIS-demands
		Values			
DOCE (Directional Earth Fault Overcurrent)			Not yet in GOMSFE		
ST (Status) - Operation Information					
FoEF earth fault foreward			Not yet in GOMSFE		
SI - (Status Input single bit)					earth fault foreward
b (binary value) := BOOL			m		
<0> := go					
<1> := come					
q (Quality) := BSTR16, Bit Number:			o		
<xxxxxx00 00000000>					
t (Timestamp) := BTIME6 (6 Octets)			o		
ReEF earth fault reverse			Not yet in GOMSFE		
SI - (Status Input single bit)					earth fault reverse
b (binary value) := BOOL			m		
<0> := go					

Functional Components					
Object Name	Common Class	Common Components	m/o	r/w/e/c	Relationship to OCIS-demands
	Values				
BrkDS (Breaker Device Status)			m	rw	Not needed, must be optional ! Same Information in CB.ST.SwDS
SIT - Status Input Double Bit					
RecIDS (Recloser Device Status)					
SIT - Status Input Double Bit					
DevSt (Device State) := b2 (binary 2 bit Value) := BSTR2			m		ARNR and ARSuc.
<00> := ? (Reclose Inhibited)					No Definition found in GOMSFE 0.8
<01> := ? (Successful Reclose)					
<10> := ? (Reclose And Lock Out)					
<11> := ?					
q (Quality) := BSTR16, Bit Number:			o		AR not ready come /go AR cycle successful

Functional Components			
Object Name			
<i>Common Class</i>		<i>m/o</i>	
<i>Common Components</i>			<i>r/w/e/c</i>
<i>Values</i>			<i>Relationship to OCIS-demands</i>
	<xxxxxx00 00000000>		
	t (Timestamp) := BTIME6	o	How to model "Reclose no longer Inhibited" = "AR not ready - go"
	ARNR (Autorecloser is not ready)		
	SI - (Status Input single bit)		
	b (binary value) := BOOL	m	
	<0> := go		
	<1> := come		
	q (Quality) := BSTR16, Bit Number:	o	
	<xxxxxx00 00000000>		
	t (Timestamp) := BTIME6 (6 Octets)	o	
	ARSuc (Autorecloser cycle successful)		
	SI - (Status Input single bit)		
	b (binary value) := BOOL	m	AR cycle successful

Not yet in GOMSFE.

Not yet in GOMSFE.

Functional Components				
Object Name				
<i>Common Class</i>		m/o		
	Common Components		r/w/e/c	
	Values			<i>Relationship to OCIS-demands</i>
	<1> := come			
	q (Quality) := BSTR16, Bit Number: <xxxxxx00 00000000>	o		
	t (Timestamp) := BTIME6 (6 Octets)	o		
	SG (Settings Groups)			
	ReclSeq (Recloser Sequence)	m	rw	Not needed, must be optional !
	SHOTS - Recloser Shots			
	Recl1st Tim (Recloser 1st Time) := INT16S	o		
	CO (Control)			
	EnaDisFct (Enable Disable Function)			<i>reclosing function on/off</i>
	DCO - Double	o	w	

Functional Components					
Object Name					
Common Class	Common Components	Values	m/o	r/w/e/c	
			<i>Relationship to OCIS-demands</i>		
Control	OperDev (Operate Device) := b2	<01> := Disable <10> := Enable	m	rw	
7.2 Switch Functions					
Sw('l) (Switch)					Busbar Disconnecter
ST (Status)					
SwDS (Switch Device State)			o	r	<i>swich state on/off/between</i>
SIT - Status Input Double Bit	DevSt (Device State) := b2 (binary 2 bit Value) := BSTR2	<00> := Between <01> := Off	m		

Functional Components				
Object Name				
<i>Common Class</i>		m/o		
	Common Components		r/w/e/c	
	Values			Relationship to OCIS-demands
	<p><10> := On</p> <p>q (Quality) := BSTR16, Bit Number:</p> <p><xxxxxxx0 00000000> A bit for "quality = substituted" is necessary.</p> <p>t (Timestamp) := BTIME6</p>	<p>o</p> <p>o</p>		<p>state substituted</p>
DC (Description)				
SwRtg (Switch Rating)		m		Not needed, must be optional !

Functional Components				
Object Name				
<i>Common Class</i>		m/o		
	Common Components		r/w/e/c	
	Values			Relationship to OCIS-demands
	<i>EqRtg - Equipment Rating</i>			
	VRtg (Voltage Rating) := VSTR16	o		
	ContCurRtg (Continuous Current Rating) := VSTR16	o		
	HzRtg (Frequency Rating) := VSTR32	o		
	TempRtg (Temperature Rating) := VSTR16	o		
	FltCurRtg (Fault Current Rating) := VSTR16	o		
	FltCurDur (Fault Current Duration) := INT16	o		
	VARtg (VA Rating) := VSTR16	o		
	VArRtg (VAr Rating) := VSTR16	o		
	UnitVArRtg := VSTR32	o		
	NumUnit (Number of Units) := VSTR32	o		
	VTcktPhs (Voltage Transformer Circuit Phases) := ?	o		
	CTcktPhs (Current Transformer Circuit Phases) := ?	o		
	ConCkt (Connected Circuit)	m		Not needed, must be optional !
	<i>ConCkt - Connected Circuit</i>			
	CktID (Circuit Identification) := VSTR32	m		
	CktPhs (Circuit Phases) := ENUM8, Integer Value =	m		

Functional Components					
Object Name	Common Class	Common Components	m/o	r/w/e/c	Relationship to OCIS-demands
		Values			
		<4> := All three			
<p>In GOMSFE it is still not possible to substitute a switch device state. A solution is shown in the IEC draft 7-3 and 7-4.</p>					
Sw(β) (Switch)					Earthing Switch
<p><i>... all identical with Sw(1)</i></p>					
Sw(51) (Switch)					Earthing Cable
<p><i>... all identical with Sw(1) but only substituted state is possible</i></p>					
<p>I think the earthing cable is the most simple "switch" that is possible. It would be excessive to model this with a Switch Controller. If it's not possible to use the Switch Object Model in this simple example, then this model makes no sense - because it will never be used.</p>					

Functional Components				
Object Name	Common Class	Common Components	Values	Relationship to OCIS-demands
CB (Circuit Breaker)				
ST (Status)				
SwDS (Switch Device State)				<i>breaker state on/off/between</i>
SIT - Status Input Double Bit				
		DevSt (Device State) := b2 (binary 2 bit Value) := BSTR2		
		<00> := Between		
		<01> := Off		
		<10> := On		
		q (Quality) := BSTR16, Bit Number:		<i>state substituted</i>
		<xxxxxxx0 00000000> A bit for "quality = substituted" is necessary.		
		t (Timestamp) := BTIME6		

Functional Components				
Object Name	Common Class	m/o	r/w/e/c	Relationship to OCIS-demands
	Common Components			
	Values			
CtlFailInd (Control Failure Indication)		o		Only under SwC or Drv ?!
SI - Status Input Single Bit				Do we need it ?
b (binary value) := BOOL <0> := FALSE <1> := TRUE q (Quality) := BSTR16, Bit Number: <xxxxxx00 00000000> t (Timestamp) := BTIME6		m		
		o		
		o		
CtlTagBlk (Control Tag Blocking)		o		Only under SwC or Drv ?!
SI - Status Input Single Bit				Do we

Functional Components				
Object Name				
<i>Common Class</i>		m/o		
	Common Components		r/w/e/c	
	Values			Relationship to OCIS-demands
	b (binary value) := BOOL <0> := FALSE <1> := TRUE q (Quality) := BSTR16, Bit Number: <xxxxxx00 00000000> t (Timestamp) := BTIME6	m o o o		need it ? Only under SwC or Drv ?! Do we need it ?
	CtlIntlkBlk (Control Interlock Blocking) SI - Status Input Single Bit b (binary value) := BOOL <0> := FALSE <1> := TRUE q (Quality) := BSTR16, Bit Number: <xxxxxx00 00000000> t (Timestamp) := BTIME6	m o o		

Functional Components				
Object Name	Common Class	Common Components	Values	Relationship to OCIS-demands
<p>I think this and more information can be transmitted by use of the "AdditionalCauseDiagnosis" shown in IEC...7-3.</p> <p>no corresponding object in GOMSFE</p>				
CO (Control)				
ODSw (Operate Device Switch)				breaker trip
DCO - Double Control				command breaker on/off
	OperDev (Operate Device) := b2			
	<01> := Off			
	<10> := On			
	SBO (Select before Operate) := IDENT := VSRTING65			SBO Control Object in CASM

Functional Components				
Object Name				
Common Class	Common Components	Values	m/o	r/w/e/c
				<i>Relationship to OCIS-demands</i>
		"BrkC.CO.ODSw.SBO"		
		In GOMSFE it is still not possible to substitute a switch device state. A solution is shown in the IEC draft 7-3 and 7-4.		
		CF (Configuration)		
		ODS w	m	
		CCF - Control Configuration		
		OnDur (On Duration) := INT32U	o	rw
		Off Dur (Off Duration) := INT32U	o	rw
		ODSwSBO SBO Configuration Object in CASM	o	
		In GOMSFE not defined - only in CASM		
		State := BOOL		
		<0> := unselected		
				Not needed, must be optional !

Functional Components				
Object Name				
<i>Common Class</i>		m/o		
	Common Components		r/w/e/c	
	Values			Relationship to OCIS-demands
	<1> := selected SelTimOut := INT8U SBOClass := ENUM8 <1> := operate once <2> := operate many			
DC (Description)				
SwRtg (Switch Rating)		m		Not needed, must be optional !
EqRtg - Equipment Rating				
ConCkt (Connected Circuit)		m		Not needed, must be optional !
ConCkt - Connected Circuit				

Functional Components					
Object Name					
<i>Common Class</i>			m/o		
	Common Components			r/w/e/c	
	Values				<i>Relationship to OCIS-demands</i>
	AX (Access)				
	Tags (the active tags on the switch)		o	rw	Use unclear, do we need it ?
	Tag				
	TagID := INT8U		m		
	TagType (Tag Type Permitted) := BSTR8, Bit Number: 0 := No Switching 1 := Operation Blocked 2 := Device Failure 3 := Interlock Active		m		When will this information be reported ? Must it be included in the data set ?
	TagD (Tag Description) := VSTR128 free text		m		
	TagOwn (Tag Owner) := VSTR32		m		

Functional Components					
Object Name	Common Class	Common Components	m/o	r/w/e/c	Relationship to OCIS-demands
	Values				
	free text				
OpenIntlk (Opening prevented by interlock logic)			o	rw	Use unclear, do we need it ?
SI - Status Input Single Bit					
b (binary value) := BOOL			m		
<0> := FALSE					
<1> := TRUE			o		
q (Quality) := BSTR16, Bit Number:					
<xxxxxx00 00000000>			o		
t (Timestamp) := BTIME6			o		
CloseIntlk (Closing prevented by interlock logic)			o	rw	Use unclear, do we need it ?
SI - Status Input Single Bit					
b (binary value) := BOOL			m		
<0> := FALSE					
<1> := TRUE			o		
q (Quality) := BSTR16, Bit Number:					

Functional Components			
Object Name	Common Class	m/o	
	Common Components		r/w/e/c
	Values		<i>Relationship to OCIS-demands</i>
	<xxxxxx00 00000000> t (Timestamp) := BTIME6	o	

I think this and more information can be transmitted by use of the "AdditionalCauseDiagnosis" shown in IEC...7-3.

20 Anlage E: Testspezifikation

20.1 Anwendungsbereich

Dieses Dokument definiert Prüfungen, die an einzelnen Geräten und an dem Gesamtsystem des Pilotprojekts OCIS durchgeführt werden. Die Prüfungen sind, wenn nicht anders vermerkt, anwendbar für die Profibus/FMS als auch für die Ethernet/MMS Variante. Durch die Prüfungen soll einerseits die Konformität der Implementierungen mit den Entwürfen der zukünftigen Normenreihe IEC 61850 und andererseits die Interoperabilität, die durch IEC 61850 derzeit nicht gewährleistet ist, von Geräten unterschiedlicher Herkunft überprüft werden.

20.2 Allgemeines

Die Prüfungen werden an allen Feldgeräten und Stationsgeräten der beiden Pilotsysteme "Profibus" und "Ethernet" durchgeführt.

Der grundsätzliche Aufbau der Pilotsysteme ist aus Bild 30 ersichtlich. Dieser besteht jeweils aus einem Stationsleitgerät, drei kombinierten Feldleit- und Schutzgeräten (Kombigeräte), einem Monitor und einer Prozesssimulation. Anhand dieser Aufbauten werden alle im folgenden beschriebenen Prüfungen durchgeführt.

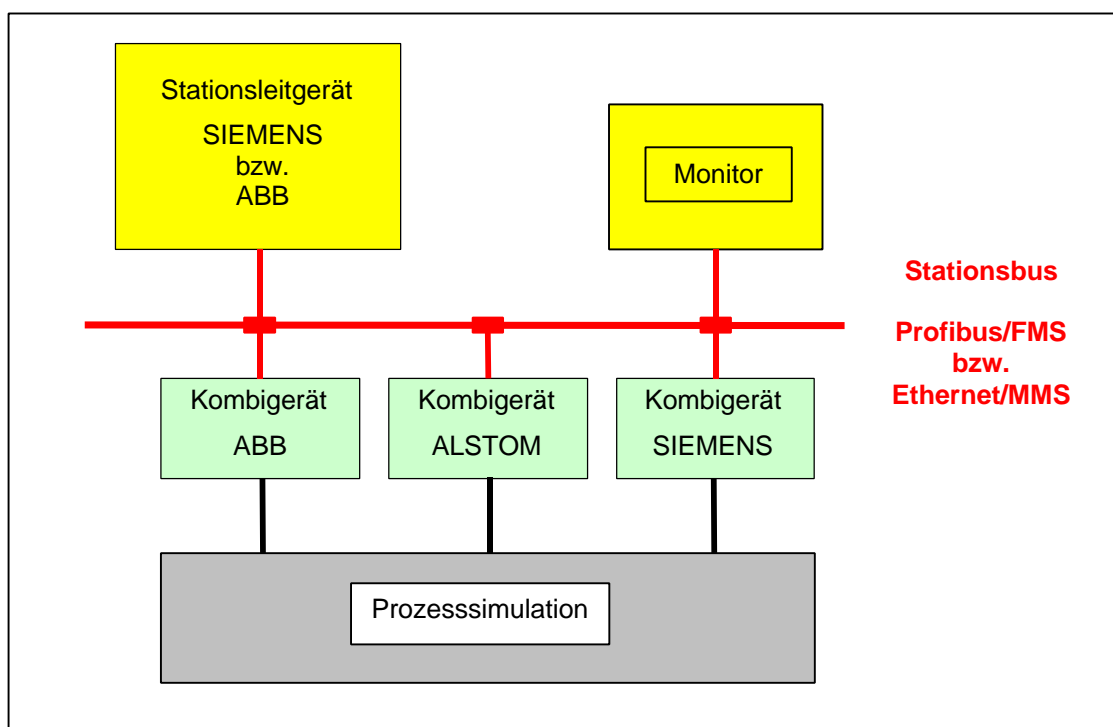


Bild 30 Aufbau der Pilotsysteme

Mit Hilfe des Monitors wird der Datenverkehr auf dem Stationsbus abgehört und analysiert. Die Prozesssimulation besteht aus einem Relais-Prüfgerät und diversen Schaltern. Das Relaisprüfgerät stellt Spannungen und Ströme zur Verfügung und ist in der Lage Störfälle zu generieren. Mit den Schaltern werden binäre Signale generiert, die die Leistungsschalter und Trenner simulieren.

20.2.1 Aufbau des Standards

Ziel des zukünftigen Standards IEC 61850 ist es unter anderem, die Daten- und Dienstmodelle für die Kommunikation in Schaltanlagen herstellerübergreifend zu standardisieren. Um diese Ziele zu erreichen, wurden bei IEC zunächst alle bekannten Funktionen eines Stationsleitsystems identifiziert und diese in Teilfunktionen zerlegt. Diese Teilfunktionen werden logische Knoten genannt. Jedem logischen Knoten werden, abhängig von der Art des logischen Knotens, Datenobjekte zugeordnet. Datenobjekte dienen dazu, die logischen Knoten detaillierter zu beschreiben. Logische Knoten und Datenobjekte sind im Teil 61850-7-4 des Standards beschrieben. Für die Übertragung des Messwerts „Strom I im Leiter L1“ existiert der logische Knoten Messwertaufnahme. Das für diesem Fall zugeordnete Datenobjekt heißt " Strom im Leiter L1". Jedem Datenobjekt wird eine Common Data Class zugeordnet. Die Common Data Class enthält neben dem eigentlichen Wert wie z.B. Messwert oder Schalterposition Zusatzinformationen, die den Wert klassifizieren, z.B. ob der Wert gültig ist oder nicht. Der Aufbau der Common Data Classes wird im Teil 61850-7-3 des Standards beschrieben. Um Informationen, d.h. Datenobjekte übertragen zu können, werden Services benutzt. Services sind zum Beispiel Aktionsanfragen bzw. Anforderungen (Request) und Antworten (Response) auf die Anfragen. Response+ ist eine positive Antwort und Response- eine negative Antwort. Services werden in IEC 61850-7-2 beschrieben. Die Gesamtheit der Services, die für die vollständige Ausführung eines Befehls benötigt werden, werden Befehlssequenzen genannt. Der Aufbau einer Befehlssequenz hängt von der Art des Befehls ab.

20.3 Prüfungen an einem Stationsleitgerät

Diese Prüfungen werden in einer späteren Version hinzugefügt.

20.4 Prüfungen an einem Feldgerät

20.4.1 Allgemeines

Zur Durchführung dieser Prüfungen wird eine Kommunikationsverbindung zwischen dem Stationsleitgerät und jeweils nur einem der Feldgeräte aufgebaut, damit eine eventuelle Beeinflussung durch die anderen Busteilnehmer ausgeschlossen werden kann. Außerdem erlaubt dieses Vorgehen eine einfachere und schnellere Analyse des Datenverkehrs.

20.4.1.1 Stationsleitgerät – Feldgerät

Die Konformität der Kommunikation zwischen Stationsleitgerät und Feldgerät soll folgendermaßen überprüft werden:

- Überprüfung des Verbindungsaufbaus und -abbaus (siehe Abschnitt 20.4.2).
- Überprüfung der Datenobjekte (siehe Abschnitt 20.4.3). Die Datenobjekte enthalten in ihren Attributen die Informationen, die übertragen werden. Es wird getestet, ob die Attribute die korrekten Werte enthalten und ob einem Datenobjekt die richtige Common Data Class zugeordnet worden ist.
- Überprüfung der Services (siehe Abschnitt 20.4.4). Die Informationen der Datenobjekte werden mit Services übertragen. Es wird überprüft, ob die Services das korrekte Format haben und die Attribute die korrekten Werte enthalten.
- Überprüfung der Common Data Class (siehe Abschnitt 20.4.7). Die Common Data Class werden Datenobjekten zugeordnet. Es wird getestet, ob die Common Data Class das korrekte Datenformat hat.
- Überprüfung der Befehlssequenzen (siehe Abschnitt 20.4.5). Um einen Befehl auszuführen, werden zwischen Client und Server verschiedene Services benutzt. Es wird überprüft, ob die Services die korrekte Reihenfolge einhalten und vollständig sind.
- Globale Überprüfung anhand eines Beispiels (siehe Abschnitt 20.8.7). Ein Leistungsschalter soll geschaltet werden.

20.4.1.2 Feldgerät – Prozess

Die Standardisierung eines Prozessbusses ist in IEC 61850 vorgesehen, aber noch nicht realisiert.

20.4.2 Aufbau und Abbau von Kommunikationsverbindungen

20.4.2.1 Associate

Zum Aufbau einer Verbindung wird in IEC 61850 im Teil 7.2 der Service Associate verwendet. Dieser Service kann auf MMS oder FMS abgebildet werden.

Testen ob das Stationsleitgerät eine Verbindung mit dem Feldgerät korrekt aufbauen kann.

Abbildung auf MMS -Service: Initiate

Abbildung auf FMS Service: Connect

20.4.2.2 DisAssociate

Zum korrekten Beenden einer Verbindung wird in IEC 61850 im Teil 7.2 der Service DisAssociate verwendet. Dieser Service kann auf MMS oder FMS abgebildet werden.

Testen ob das Stationsleitgerät eine Verbindung mit dem Feldgerät korrekt beenden kann.

Abbildung auf MMS -Service: Conclude

Abbildung auf FMS Service: Disconnect

20.4.2.3 Abort

Zum Abbrechen einer Verbindung wird in IEC 61850 im Teil 7.2 der Service Abort verwendet. Dieser Service kann auf MMS oder FMS abgebildet werden.

Testen ob das Stationsleitgerät eine Verbindung mit dem Feldgerät korrekt abbrechen kann.

Abbildung auf MMS -Service: Reject/Abort

Abbildung auf FMS Service: Disconnect

20.4.3 Prüfungen an Datenobjekten

20.4.3.1 Allgemeines

Der Teil 4.3 wird verwendet, um die Datenobjekte auf Konformität bezüglich des Standardentwurfs IEC 61850 zu überprüfen.

Verwendung der Tabellen:

Test-Nr.	Test	Beschreibung	1/F
Spalte 1	Spalte 2	Spalte 3	Spalte 4

Spalte Test-Nr.: Diese Spalte enthält eine fortlaufend nummerierte Test Nummer (Referenznummer).

Spalte Test: Diese Spalte enthält die IEC-Kurzform des Testobjektes (Attribute der Common Data Class).

Spalte Beschreibung: Diese Spalte enthält eine Beschreibung des Testobjektes (Attribute der Common Data Class).

Spalte 1/F: Diese Spalte wird verwendet um die korrekte Testdurchführung zu bestätigen.

1 = Normkonform

F = Fehler

Eine nicht ausgefüllte Spalte bedeutet optional.

20.4.3.2 LED reset

Bei Feldgeräten verwendet man zur Anzeige von Störungen LEDs. Um zu bestätigen, dass die Meldung erkannt wurde und diese akzeptiert wird, müssen die LED-Anzeigen quittiert werden. Die Quittierung kann entweder über das Stationsleitgerät mit einem Befehl an das Feldgerät erfolgen oder über einem Quittierungstaster direkt am Feldgerät. Im ersten Fall gibt das Stationsleitgerät dem Feldgerät mit dem Service Operate Request den Befehl die Status LED zu quittieren. Das Feldgerät führt den Befehl aus und antwortet bei erfolgreicher Befehlsdurchführung mit dem Service Operate Response+ und bei nicht erfolgreicher Befehlsdurchführung mit dem Service Operate Response-. Der Service Operate Request überträgt das Datenobjekt LED reset. Im zweiten Fall wird die Quittierung des Status LED

direkt am Feldgerät durch den Quittierungs-Taster ausgeführt. Das Feldgerät muss dieses dem Stationsleitgerät mit dem Service Report mitteilen. Der Service Report überträgt das Datenobjekt LED reset.

Table 1: LED reset, station unit to bay unit

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPC für diesen Befehl. Die Änderung des Statuswertes wird mit der Common Data Class SPS dargestellt.	
	Object Identification	Variablenname ist LLN0.LED	
	Control Value	Bei Befehl vom Stationsleitgerät zum Feldgerät ist der Wert 1	
	Status Value	Bei Statusmeldung des Feldgerätes zum Stationsleitgerät ist der Wert 1	
	Quality Descriptor	Der Status des Wertes eines Datenobjektes wird als SET OF BOOLEAN dargestellt. Jedes Bit des Quality Descriptor stellt eine Eigenschaft des Statuswertes dar und ist auf FALSE voreingestellt. Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. CONTROL	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Address	Die Adresse des Verursacher der Statuswertänderung sollte richtig angezeigt werden.	
	Originator Category	0 bei Statusmeldungen 1 bei Steuerungsvorgängen des Feldgerätes 2 bei Steuerungsvorgängen des Stationsleitgerätes	
	Control Sequence Number	Die ControlSequenceNumber hat den Wert n und muss während der gesamten Befehlssequenz den Wert n beibehalten. Integer Wert (0-255)	

20.4.3.3 Blocking information exchange

Bei den Feldgeräten verwendet man Informationssperren damit bei Wartungsarbeiten am Feld keine Meldungen zum Stationsleitgerät übertragen werden. Bei einer gesetzten Infosperre müssen alle im Feldgerät auftretenden Meldungen verworfen werden. Nach Aufheben der Infosperre muss eine automatische Generalabfrage gestartet werden. Aus dem Zeitraum der gesetzten Infosperre dürfen keine Meldungen übertragen werden.

Das Stationsleitgerät gibt dem Feldgerät mit dem Service Operate Request den Befehl alle anfallenden Daten zu verwerfen. Das Feldgerät führt den Befehl aus und antwortet bei erfolgreicher Befehlsdurchführung mit dem Service Operate Response+ und bei nichterfolgreicher Befehlsdurchführung mit dem Service Operate Response-. Der Service Operate Request überträgt das Datenobjekt Blocking information exchange.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPC für diesen Befehl. Die Änderung des Statuswertes wird mit der Common Data Class DPS dargestellt.	
	Object identification	Variablenname ist LLN0.IEBI	
	Control Value	Wert ist 1 = ON oder 0 = OFF	
	Status Value	Wert ist 1 = ON oder 0 = OFF	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. CONTROL	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Address	Die Adresse des Verursacher der Statuswertänderung muss richtig angezeigt werden. 0 bei Statusmeldungen 2 bei Steuerungsvorgängen	
	Originator Category		

Test-Nr.	Test	Beschreibung	ok
	Control Sequence Number	Die ControlSequenceNumber hat den Wert n und muss während der gesamten Befehlssequenz den Wert n beibehalten. Integer Wert (0-255)	

20.4.3.4 activate characteristic

In den Feldgeräten besteht die Möglichkeit beim Schutz zwischen den verschiedenen parametrisierten Kennlinien 1-n zu wechseln. Im Augenblick der Kennlinienumschaltung (nicht gleichbedeutend mit Parameterumschaltung) ist ein Feldgerät gesperrt.

Das Stationsleitgerät gibt dem Feldgerät mit dem Service Operate Request den Befehl, auf eine andere Kennlinie umzuschalten. Das Feldgerät führt den Befehl aus und antwortet bei erfolgreicher Befehlsdurchführung mit dem Service Operate Response+ und bei nichterfolgreicher Befehlsdurchführung mit dem Service Operate Response-. Der Service Operate Request überträgt das Datenobjekt activate characteristic.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPC für den Befehl. Die Änderung des Statuswertes wird mit der Common Data Class DPS dargestellt.	
	Object identification	Variablenname ist LLN0.Ch	
	Control Value	Wert ist 1 - n	
	Status Value	Wert ist 1 - n	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. CONTROL	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Address	Die Adresse des Verursacher der Statuswertänderung muss richtig angezeigt werden.	
	Originator Category	0 bei Statusmeldungen 2 bei Steuerungsvorgängen	

Test-Nr.	Test	Beschreibung	ok
	Control Sequence Number	Die ControlSequenceNumber hat den Wert n und muss während der gesamten Befehlssequenz den Wert n beibehalten. Integer Wert (0-255)	

20.4.3.5 supervision alarm

Aktion: Warnmeldung Gerät

Eine Warnmeldung eines Funktionsblocks z.B. Display defekt muss das Feldgerät dem Stationsleitgerät mit dem Service Report Request mitteilen. Warnung bedeutet, dass in einem Funktionsblock ein Fehler aufgetreten ist, der Funktionsblock aber trotzdem noch betriebsbereit ist.

Der Report Request überträgt das Datenobjekt supervision alarm.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPS	
	Object identification	Variablenname ist LLN0.SupAl	
	Status Value	Wert ist 1 = ON oder 0 = OFF	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung muss richtig angezeigt werden. OriginatorCategory ist 0 .	

20.4.3.6 LN Disturbance

Eine Störung innerhalb des Feldgerätes z.B. UMZ-Schutz Störung muss das Feldgerät dem Stationsleitgerät mit dem Service Report Request mitteilen. Störung bedeutet, dass der Funktionsblock vollständig ausgefallen ist.

Der Service Report Request überträgt das Datenobjekt LN Disturbance.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPS	
	Object identification	Variablenname ist LLN0.DisDS	
	Status Value	Wert ist 1 = ON oder 0 = OFF	
	Quality Descriptor	– Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung muss richtig angezeigt werden. OriginatorCategory ist 0 .	

20.4.3.7 local operation (not remote)

Um ein Gerät der Primärtechnik einer Schaltanlage direkt vom Feldgerät zu schalten, muss die Steuerungsart von Nahsteuerung auf Ortsteuerung geändert werden. Die Steuerungsart muss am Feldgerät angezeigt werden und sie muss dem Stationsleitgerät mit dem Service Report Request mitgeteilt werden. Der Service Report Request überträgt das Datenobjekt local operation (not remote).

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPS	
	Object	Variablenname ist LLN0.Loc	

Test-Nr.	Test	Beschreibung	ok
	identification		
	Status Value	Wert ist 1 für remote und 0 für local	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung sollte richtig angezeigt werden. OriginatorCategory ist 0 .	

20.4.3.8 parameter setting

Im Augenblick der Parameterumschaltung (nicht gleichbedeutend mit Kennlinienumschaltung) ist ein Feldgerät gesperrt.

Bei einem Feldgerät wird eine Parametrierung durchgeführt. Das Feldgerät muss dem Stationsleitgerät dieses mit dem Service Report Request mitteilen. Der Service Report Request überträgt das Datenobjekt parameter setting.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPS	
	Object identification	Variablenname ist LLN0.PaSet	
	Status Value	Wert ist 1 = ON oder 0 = OFF	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht	

Test-Nr.	Test	Beschreibung	ok
		wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung sollte richtig angezeigt werden. OriginatorCategory ist 0 .	

20.4.3.9 earth fault forward

In der Prozesssimulation wird ein Erdkurzschluss in Vorwärtsrichtung erzeugt. Dieser Störfall muss dem Stationsleitgerät mit dem Service Report Request vom Feldgerät mitgeteilt werden. Der Service Report Request überträgt das Datenobjekt earth fault forward.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPS	
	Object identification	Variablenname ist PDEF.FoEF	
	Status Value	Wert ist 1 = ON oder 0 = OFF	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung sollte richtig angezeigt werden. OriginatorCategory ist 0	

20.4.3.10 earth fault reverse

In der Prozesssimulation wird ein Erdkurzschluss in Rückwärtsrichtung erzeugt. Dieser Störfall muss dem Stationsleitgerät mit dem Service Report Request vom Feldgerät mitgeteilt werden. Der Service Report Request überträgt das Datenobjekt earth fault reverse.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPS	
	Object identification	Variablenname ist PDEF.ReEF	
	Status Value	Wert ist 1 =ON oder 0 =OFF	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung sollte richtig angezeigt werden. OriginatorCategory ist 0 .	

20.4.3.11 measurand supervision I

In der Prozesssimulation soll eine eingestellte Stromgrenze überschritten werden. Dieses muss dem Stationsleitgerät mit dem Service Report Request vom Feldgerät mitgeteilt werden. Der Service Report Request überträgt das Datenobjekt measurand supervision I.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPS	
	Object identification	Variablenname ist PBRO.SupA	
	Status Value	Wert ist 1 =ON oder 0 =OFF	

Test-Nr.	Test	Beschreibung	ok
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung sollte richtig angezeigt werden. OriginatorCategory ist 0	

20.4.3.12 alarm L1

Das Feldgerät enthält die Schutzfunktion Überstromzeitschutz. In der Prozesssimulation soll der Strom im Leiter 1 (L1 bzw. A) den eingestellten Grenzwert überschreiten. Dieser Fehler muss dem Stationsleitgerät mit dem Service Report Request vom Feldgerät mitgeteilt werden. Der Service Report Request überträgt das Datenobjekt alarm L1

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPS	
	Object identification	Variablenname ist PTOC1.AIA	
	Status Value	Wert ist 1 = ON oder 0 = OFF	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE)	

Test-Nr.	Test	Beschreibung	ok
		notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung sollte richtig angezeigt werden. OriginatorCategory ist 0	

20.4.3.13 alarm L2

Das Feldgerät enthält die Schutzfunktion Überstromzeitschutz. In der Prozesssimulation soll der Strom im Leiter 2 (L2 bzw. B) den eingestellten Grenzwert überschreiten. Dieser Fehler muss dem Stationsleitgerät mit dem Service Report Request vom Feldgerät mitgeteilt werden. Der Service Report Request überträgt das Datenobjekt alarm L2

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPS	
	Object identification	Variablenname ist PTOC1.AIB	
	Status Value	Wert ist 1 = ON oder 0 = OFF	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung sollte richtig angezeigt werden. OriginatorCategory ist 0	

20.4.3.14 alarm L3

Das Feldgerät enthält die Schutzfunktion Überstromzeitschutz. In der Prozesssimulation soll der Strom im Leiter 3 (L3 bzw. C) den eingestellten Grenzwert überschreiten. Dieser Fehler muss dem Stationsleitgerät mit dem Service Report Request vom Feldgerät mitgeteilt werden. Der Service Report Request überträgt das Datenobjekt alarm L3.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPS	
	Object identification	Variablenname ist PTOC1.AIC	
	Status Value	Wert ist 1 =ON oder 0 =OFF	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung sollte richtig angezeigt werden. OriginatorCategory ist 0	

20.4.3.15 alarm N

Das Feldgerät enthält die Schutzfunktion Überstromzeitschutz. In der Prozesssimulation soll der Strom im Neutralleiter (N) den eingestellten Grenzwert überschreiten. Dieser Fehler muss dem Stationsleitgerät mit dem Service Report Request vom Feldgerät mitgeteilt werden. Der Service Report Request überträgt das Datenobjekt alarm N

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPS	
	Object identification	Variablenname ist PTOC1.AIN	
	Status Value	Wert ist 1 = ON oder 0 = OFF	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung sollte richtig angezeigt werden. OriginatorCategory ist 0	

20.4.3.16 general alarm

Aktion: Generalanregung

Das Feldgerät enthält die Schutzfunktion Überstromzeitschutz. In der Prozesssimulation soll der Strom in einem Leiter den eingestellten Grenzwert überschreiten. Diese Überschreitung muss dem Stationsleitgerät mit dem Service Report Request vom Feldgerät mitgeteilt werden. Der Service Report Request enthält das Datenobjekt general alarm.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPS	
	Object identification	Variablenname ist PBRO.GAI	
	Status Value	Wert ist 1 = ON oder 0 = OFF	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung sollte richtig angezeigt werden. OriginatorCategory ist 0	

20.4.3.17 general trip

Aktion: Generalauslösung (Wischer)

Das Feldgerät enthält die Schutzfunktion Überstromzeitschutz. In der Prozesssimulation soll der Strom in einem Leiter den eingestellten Grenzwert überschreiten. Das Feldgerät muss den Leistungsschalter ausschalten. Diese Auslösung muss dem Stationsleitgerät mit dem Service Report Request vom Feldgerät mitgeteilt werden. Der Service Report Request enthält das Datenobjekt general trip.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPS	
	Object identification	Variablenname ist PBRO.GTr	
	Status Value	Wert ist 1=ON	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung sollte richtig angezeigt werden. OriginatorCategory ist 0	

20.4.3.18 trip I>

Aktion: I> Auslösung (Wischer)

Das Feldgerät enthält die Schutzfunktion Überstromzeitschutz. In der Prozesssimulation soll der Strom in L1 den kleineren eingestellten Grenzwert überschreiten. Das Feldgerät muss den Leistungsschalter ausschalten. Diese Auslösung muss dem Stationsleitgerät mit dem Service Report Request vom Feldgerät mitgeteilt werden. Der Service Report Request enthält das Datenobjekt trip I>.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPS	
	Object identification	Variablenname ist PTOC1.TrPTOC	
	Status Value	Wert ist 1 = ON	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung sollte richtig angezeigt werden. OriginatorCategory ist 0	

20.4.3.19 trip I>>

Aktion: I>> Auslösung (Wischer)

Das Feldgerät enthält die Schutzfunktion Überstromzeitschutz. In der Prozesssimulation soll der Strom in L1 den größeren eingestellten Grenzwert überschreiten. Das Feldgerät muss den Leistungsschalter ausschalten. Diese Auslösung muss dem Stationsleitgerät mit dem Service Report Request vom Feldgerät mitgeteilt werden. Der Service Report Request enthält das Datenobjekt trip I>>.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPS	
	Object identification	Variablenname ist PTOC1.TrPIOC	
	Status Value	Wert ist 1 = ON	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung sollte richtig angezeigt werden. OriginatorCategory ist 0	

20.4.3.20 LN ON (not OFF)

Aktion: Meldung AWE Ein/Aus

Feldgerät enthält die Funktion Automatische Wiedereinschaltung. Das Stationsleitgerät gibt dem Feldgerät mit dem Service Operate Request den Befehl die Funktion Automatische Wiedereinschaltung zu aktivieren. Das Feldgerät führt den Befehl aus und antwortet bei erfolgreicher Befehlsdurchführung mit dem Service Operate Response+ und bei nichterfolgreicher Befehlsdurchführung mit dem Service Operate Response-. Der Service Operate Request überträgt das Datenobjekt LN ON (not OFF).

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPC für den Befehl. Die Änderung des Statuswertes wird mit der Common Data Class SPS dargestellt.	
	Object identification	Variablenname ist RREC.EnaFct	
	Control Value	Wert ist 1 = ON oder 0 = OFF	
	Status Value	Wert ist 1 = ON oder 0 = OFF	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. CONTROL	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Address	Die Adresse des Verursacher der Statuswertänderung sollte richtig angezeigt werden.	
	Originator Category	0 bei Statusmeldungen 2 bei Steuerungsvorgängen	
	Control Sequence	Die ControlSequenceNumber hat den Wert n und muss während der gesamten Befehlssequenz den Wert n	

Test-Nr.	Test	Beschreibung	ok
	Number	beibehalten. Integer Wert (0-255)	

20.4.3.21 AR is not ready

Aktion: Meldung AWE nicht bereit

Feldgerät enthält die Funktion Automatische Wiedereinschaltung. Wenn die automatische Wiedereinschaltung nicht bereit ist, muss das Feldgerät dem Stationsleitgerät dieses mit dem Datenobjekt AR is not ready mitteilen.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPS	
	Object identification	Variablenname ist RREC.ARRR	
	Status Value	Wert ist 1 = ON oder 0 = OFF	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung sollte richtig angezeigt werden. OriginatorCategory ist 0	

20.4.3.22 AR cycle successful

Aktion: Meldung AWE war erfolgreich

Feldgerät enthält die Funktion Automatische Wiedereinschaltung. Nach erfolgreicher Durchführung einer Automatischen Wiedereinschaltung muss das Feldgerät dem Stationsleitgerät dieses mit dem Datenobjekt AR cycle successful mitteilen.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPS	
	Object identification	Variablenname ist RREC.ARSuc	
	Status Value	Wert ist 1 = ON	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung sollte richtig angezeigt werden. OriginatorCategory ist 0	

20.4.3.23 power supply alarm 1

Aktion: Meldung Spannungsüberwachung Alarm 1

Das Feldgerät muss einen Fehler in der Spannungsversorgung 1 (Spannungsunterbrechung, Spannungsausfall oder Spannungsschwankung) mit dem Datenobjekt power supply alarm dem Stationsleitgerät mitteilen.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPS	
	Object identification	Variablenname ist GGIO.PwrSupAlm1	
	Status Value	Wert ist 1 = ON oder 0 = OFF	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung sollte richtig angezeigt werden. OriginatorCategory ist 0	

20.4.3.24 power supply alarm 2

Aktion: Meldung Spannungsüberwachung Alarm 2

Das Feldgerät muss einen Fehler in der Spannungsversorgung 2 (Spannungsunterbrechung, Spannungsausfall oder Spannungsschwankung) mit dem Datenobjekt power supply alarm dem Stationsleitgerät mitteilen.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPS	
	Object identification	Variablenname ist GGIO.PwrSupAlm2	
	Status Value	Wert ist 1 = ON oder 0 = OFF	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung sollte richtig angezeigt werden. OriginatorCategory ist 0	

20.4.3.25 power supply alarm 3

Aktion: Meldung Spannungsüberwachung Alarm 3

Das Feldgerät muss einen Fehler in der Spannungsversorgung 3 (Spannungsunterbrechung, Spannungsausfall oder Spannungsschwankung) mit dem Datenobjekt power supply alarm dem Stationsleitgerät mitteilen.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPS	
	Object identification	Variablenname ist GGIO.PwrSupAlm3	
	Status Value	Wert ist 1 = ON oder 0 = OFF	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung sollte richtig angezeigt werden. OriginatorCategory ist 0	

20.4.3.26 IL1 RMS present value

Aktion: Messwert I wird übermittelt

Bei einer vorgegebenen Änderung des Stromes I muss das Feldgerät dem Stationsleitgerät diese Änderung durch das Datenobjekt IL1 RMS present value mitteilen.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist MV	
	Object identification	Variablenname ist MMXU.APhsA_RMS	
	Value	Der Wert kann dargestellt werden als: – Integer – Fließkommazahl	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist. Überprüfen, ob das entsprechende Bit auf overflow gesetzt wird bei einem Statuswert, der den vordefinierten Wert überschritten hat.	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	

20.4.3.27 Position (Circuit breaker controller)

Aktion: Schalten des Leistungsschalters

Das Stationsleitgerät sendet dem Feldgerät mit dem Service Select with value Request eine Anfrage, dass der Leistungsschalter angewählt werden soll.

Das Feldgerät beantwortet die Anfrage mit dem Service Select with value Response+ wenn der Leistungsschalter angewählt worden ist. Das Feldgerät beantwortet die Anfrage mit dem Service Select with value Response- wenn der Leistungsschalter nicht angewählt werden kann.

Nach dem Erhalt des Service Select with value Response+ sendet das Stationsleitgerät den Service Operate Request an das Feldgerät. Dieser Service ist der Schaltbefehl.

Wenn das Feldgerät den Ausgang aktiviert hat (den Schaltbefehl an den Schalter gegeben hat), gibt das Feldgerät als Bestätigung den Service Operate Response+ an das

Stationsleitgerät. Wenn das Feldgerät den Ausgang nicht aktivieren konnte, gibt es als Bestätigung den Service Operate Response- an das Stationsleitgerät.

Der Service Operate überträgt das Datenobjekt Position.

Nach jeder Veränderung sendet das Feldgerät den Service Report Request. Er enthält das Datenobjekt Position.

Zum Abschluss der Sequenz sendet das Feldgerät den Service Command Termination_Request+, um den Schaltvorgang zu beenden.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist SPC für den Schaltbefehl. Die Änderung des Statuswertes wird mit der Common Data Class DPS dargestellt.	
	Object identification	Variablenname ist CSWI1.Pos	
	Control Value	Wert ist 1 = ON oder 0 = OFF	
	Status Value	Die korrekte Schalterstellung sollte übermittelt werden. ON OFF INTERMEDIATE_STATE BAD_STATE	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. CONTROL , für jeden Service einer Befehlssequenz, wenn der Leistungsschalter durch Schaltbefehl geschaltet wurde. SPONTANEOUS , wenn der Leistungsschalter nicht durch einen Schaltbefehl geschaltet wurde, z.B. durch eine Schutzauslösung.	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	

Test-Nr.	Test	Beschreibung	ok
	Originator Address	Der Adresse des Verursacher der Änderung des Statuswertes sollte richtig angezeigt werden.	
	Originator Category	0 bei Statusmeldungen 2 bei Steuerungsvorgängen	
	Control Sequence Number	Die ControlSequenceNumber hat den Wert n und muss während der gesamten Befehlssequenz den Wert n beibehalten. Integer Wert (0-255)	

20.4.3.28 Position (Busbar disconnecter)

Aktion: Nachführen des Trenners

Beim Projekt OCIS ist der Sammelschienentrennschalter nur von Hand zu betätigen. Dieses bedeutet, dass der Trenner nicht vom Stationsleitgerät geschaltet werden kann. Der Trenner ist nur in Melderichtung an die Stationsleittechnik angebunden.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist DPS	
	Object identification	Variablenname ist CSWI2.Pos	
	Wert	Die korrekte Schalterstellung sollte übermittelt werden. TRUE, FALSE, INTERMEDIATE_STATE, BAD_STATE	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE	

Test-Nr.	Test	Beschreibung	ok
		eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung sollte richtig angezeigt werden. OriginatorCategory ist 0	

20.4.3.29 Position (Line earthing switch)

Aktion: Statusmeldung des Erders

Beim Projekt OCIS ist der Abgangserder nur von Hand zu betätigen. Dieses bedeutet, dass der Erder nicht vom Stationsleitgerät geschaltet werden kann. Der Erder ist nur in Melderichtung an die Stationsleittechnik angebunden.

Test-Nr.	Test	Beschreibung	ok
	Common Data Class	Common Data Class ist DPS	
	Object identification	Variablenname ist CSWI3.Pos	
	Wert	Die korrekte Schalterstellung sollte übermittelt werden. TRUE, FALSE, INTERMEDIATE_STATE, BAD_STATE	
	Quality Descriptor	Überprüfen, ob jedes Bit des Quality Descriptors auf FALSE eingestellt ist.	
	ReasonFor StatusChange	Überprüfen, ob die Änderung des Statuswertes durch einen normalen Vorgang (SPONTANEOUS) oder durch einen direkten Steuervorgang (CONTROL) verursacht wurde. SPONTANEOUS	
	Time stamp	Zeit = Zeit der Meldungsgenerierung GENERALIZED_TIME	
	Time quality	notStandardTime; (TRUE / FALSE) notSynchronised; (TRUE / FALSE) invalid; (TRUE / FALSE) Überprüfen, ob die Attribute der Time quality auf FALSE eingestellt sind.	
	Originator Category	Der Verursacher der Statuswertänderung sollte richtig angezeigt werden. OriginatorCategory ist 0	

20.4.4 Services

Für die Befehlssequenzen werden Services aus IEC 61850-7-2 verwendet. Diese müssen den aus den folgenden Abschnitten ersichtlichen Aufbau haben.

20.4.4.1 Report

Der Service Report wird verwendet, um eine Meldung in Form eines unbestätigten Dienstes vom Server zum Client zu senden. Ein Report wird vom Server zum Client stets gesendet, wenn sich der Wert, der im Datenobjekt enthalten ist, ändert. In der nachfolgenden Tabelle ist der grundsätzliche Aufbau eines Reports dargestellt.

Test-Nr.	Test	Beschreibung	ok
	Object Identification	Die ObjectIdentification enthält die Adresse des Datenobjektes, das durch den Report gemeldet werden soll. Überprüfen, ob das Attribut ist vom Datentyp VSTR32 ist.	
	OptFldsRpt	In diesem Feld kann ausgewählt werden, welche optionalen Attribute im Report enthalten sein können. Datentyp: BSTR8 Auswahl der Attribute: Bit 0: RESERVED Bit 1: SEQUENCE_NUMBER Bit 2: REPORT_TIME_STAMP Bit 3: REASON_FOR_INCLUSION Bit 4: DATA_SET_NAME Bit 5: DATA_OBJECT_NAMES_INCLUSION_BITSTRING	
	SeqNum	Ist nur im Report enthalten, wenn Bit 1 in OptFldsRpt den Wert TRUE besitzt. Datentyp ist INT8U (Wertebereich 0 bis 232)	
	SubSeqNum	Datentyp ist INT8U (Wertebereich 0 bis 255)	
	RptTim	Zeigt den Zeitpunkt an, in dem der Report generiert wurde. Ist nur im Report enthalten, wenn Bit 2 in OptFldsRpt den Wert TRUE besitzt. Datentyp ist BTIME6	
	ListOf {	STRUCTURE	
	Value	Der Datentyp der Attribute des Datenobjektes das durch den Report gemeldet werden soll.	
	ReasonCode}	Ist nur im Report enthalten, wenn Bit 3 in OptFldsRpt den Wert TRUE besitzt.	

Test-Nr.	Test	Beschreibung	ok
		SET OF Periodic [0] BOOLEAN OPTIONAL Integrity [1] BOOLEAN OPTIONAL Change [2] DEFAULT TRUE Hi-limit [3] BOOLEAN OPTIONAL Lo-limit [4] BOOLEAN OPTIONAL Percent_change [5] BOOLEAN OPTIONAL Gen_interrog [6] BOOLEAN OPTIONAL Test [7] BOOLEAN OPTIONAL	

20.4.4.2 Operate Request

Ein Operate Request wird vom Client zum Server gesendet. Es handelt sich hierbei um einen bestätigten Dienst. Der Operate Request wird verwendet, um den Wert eines Datenobjektes zu ändern. Der Aufbau eines Operate Request ist in der nachfolgenden Tabelle dargestellt.

Test-Nr.	Test	Beschreibung	ok
	Object Identification	Die ObjectIdentification enthält die Adresse des Datenobjektes, auf das zugegriffen wird. Überprüfen, ob das Attribut ist vom Datentyp VSTR32 ist.	
	Value	Der Wert richtet sich nach der Common Data Class des Datenobjekts. Überprüfen, ob der Wert den korrekten Datentyp hat.	
	CtlNum	Die ControlSequenceNumber dient als Kennung von Steuerbefehlen. Mit Hilfe der ControlSequenceNumber kann eine Response dem entsprechenden Request zugeordnet werden. Sie wird fortlaufend nummeriert. Der Wert der ControlSequenceNumber muss einen Integer-Wert zwischen 0 und 255 haben.	
	ControlDetail	Das ControlDetail besteht aus folgenden optionalen Attributen:	
		Test Der Test-Status definiert, ob die Information durch eine normale Operation oder durch einen Test verursacht wurde. Es wird ein boolescher Wert angegeben. 0=NoTest 1=Test	
		OriginatorCategory	
		OriginatorAddress	

Test-Nr.	Test	Beschreibung	ok
		Time class Die Time class definiert, welche Zeitschritte für den Zeitstempel benutzt werden. Es wird ein ENUM4-Wert angegeben. 0 = reserved 1 = 1ms 2 = 1/10ms 3 = 1/100 ms	
		Time Stamp	
	Check	Das Attribut Check spezifiziert, welche Art von Prüfungen ein Server ausführt, bevor der Control-Vorgang eingeleitet wird. Check wird als SET OF BOOLEAN übertragen.	

20.4.4.3 Operate Response+

Operate Response+ wird als positive Antwort auf ein Operate Request gesendet. Der Aufbau einer Operate Response+ ist in der nachfolgenden Tabelle dargestellt.

Test-Nr.	Test	Beschreibung	ok
	Object Identification	Die ObjectIdentification enthält die Adresse des Datenobjektes, auf das zugegriffen wird. Überprüfen, ob das Attribut ist vom Datentyp VSTR32 ist.	
	Value	Der Wert richtet sich nach der Common Data Class des Datenobjekts. Überprüfen, ob der Wert den korrekten Datentyp hat.	
	CtlNum	Die ControlSequenceNumber dient als Kennung von Steuerbefehlen. Mit Hilfe der ControlSequenceNumber kann eine Response dem entsprechenden Request zugeordnet werden. Sie wird fortlaufend nummeriert. Der Wert der ControlSequenceNumber muss einen Integer-Wert von 0 bis 255 haben.	
	ControlDetail	Das ControlDetail besteht aus folgenden optionalen Attributen:	
		Test Der Test-Status definiert, ob die Information durch eine normale Operation oder durch einen Test verursacht wurde. Es wird ein boolescher Wert angegeben. 0 =NoTest 1 =Test	
		OriginatorCategory	
		OriginatorAddress	

Test-Nr.	Test	Beschreibung	ok
		Time class Die Time class definiert, welche Zeitschritte für den Zeitstempel benutzt werden. Es wird ein ENUM4-Wert angegeben. 0 = reserved 1 = 1ms 2 = 1/10ms 3 = 1/100 ms	
		Time Stamp	

20.4.4.4 Operate Response–

Operate Response– wird als negative Antwort auf ein Operate Request gesendet. Der Aufbau einer Operate Response– ist in der nachfolgenden Tabelle dargestellt.

Test-Nr.	Test	Beschreibung	ok
	Object Identification	Die ObjectIdentification enthält die Adresse des Datenobjektes, auf das zugegriffen wird. Überprüfen, ob das Attribut ist vom Datentyp VSTR32 ist.	
	Value	Der Wert richtet sich nach der Common Data Class des Datenobjekts. Überprüfen, ob der Wert den korrekten Datentyp hat.	
	CtlNum	Die ControlSequenceNumber dient als Kennung von Steuerbefehlen. Mit Hilfe der ControlSequenceNumber kann eine Response dem entsprechenden Request zugeordnet werden. Sie wird fortlaufend nummeriert. Der Wert der ControlSequenceNumber muss einen Integer-Wert von 0 bis 255 haben.	
	ControlDetail	Das ControlDetail besteht aus folgenden optionalen Attributen:	
		Test Der Test-Status definiert, ob die Information durch eine normale Operation oder durch einen Test verursacht wurde. Es wird ein boolescher Wert angegeben. 0 =NoTest 1 =Test	
		OriginatorCategory	
		OriginatorAddress	

Test-Nr.	Test	Beschreibung	ok
		<p>Time class Die Time class definiert, welche Zeitschritte für den Zeitstempel benutzt werden. Es wird ein ENUM4-Wert angegeben. 0 = reserved 1 = 1ms 2 = 1/10ms 3 = 1/100 ms</p>	
		<p>Time Stamp</p>	
	<p>Cause</p>	<p>Das Attribut Cause gibt Auskunft über die Fehlerursache, die zur Abweisung des Services geführt hat. Cause wird als ENUM angegeben. Folgende Ursachen sind möglich:</p> <ul style="list-style-type: none"> [0] = not_supported [1] = blocked_by_switching_hierarchy [2] = blocked_by_event [3] = blocked_by_interlocking [4] = blocked_by_setpoint_command [5] = target_exists [6] = parameter_error [7] = time_limit_over [8] = address_error [9] = hardware_error [10] = 1_of_n_control [11] = system_crash [12] = step_limit [13] = command_already_in_execution [14] = plausibility_error [15] = blocked_by_synchrocheck [16] = debounce_active [17] = abortion [18] = parameter_charge_in_execution [19] = CB_alarm 	

20.4.4.5 Command termination Request+

Die Beendigung der Befehlssequenz wird mit dem Service Command termination eingeleitet. Bei einem Command termination Request+ handelt es sich um eine positive Beendigung der Verbindung. Der Aufbau eines Command termination Requests+ ist in der nachfolgenden Tabelle dargestellt.

Test-Nr.	Test	Beschreibung	ok
	Object Identification	Die ObjectIdentification enthält die Adresse des Datenobjektes, auf das zugegriffen wird. Überprüfen, ob das Attribut ist vom Datentyp VSTR32 ist.	
	CtlNum	Die ControlSequenceNumber dient als Kennung von Steuerbefehlen. Mit Hilfe der ControlSequenceNumber kann eine Response dem entsprechenden Request zugeordnet werden. Sie wird fortlaufend nummeriert. Der Wert der ControlSequenceNumber muss einen INTEGER-Wert von 0 bis 255 haben.	
	ControlDetail	Das ControlDetail besteht aus folgenden optionalen Attributen:	
		Test Der Test-Status definiert, ob die Information durch eine normale Operation oder durch einen Test verursacht wurde. Es wird ein boolescher Wert angegeben. 0 =NoTest 1 =Test	
		OriginatorCategory	
		OriginatorAddress	
		Time class Die Time class definiert, welche Zeitschritte für den Zeitstempel benutzt werden. Es wird ein ENUM4-Wert angegeben. 0 = reserved 1 = 1ms 2 = 1/10ms 3 = 1/100 ms	
		Time Stamp	

20.4.4.6 Command termination Request–

Die Beendigung der Befehlssequenz wird mit dem Service Command termination eingeleitet. Bei einem Command termination Request– handelt es sich um eine negative Beendigung der Verbindung. Der Aufbau eines Command termination Requests– ist in der nachfolgenden Tabelle dargestellt.

Test-Nr.	Test	Beschreibung	ok
	Object Identification	Die ObjectIdentification enthält die Adresse des Datenobjektes, auf das zugegriffen wird. Überprüfen, ob das Attribut ist vom Datentyp VSTR32 ist.	
	CtlNum	Die ControlSequenceNumber dient als Kennung von Steuerbefehlen. Mit Hilfe der ControlSequenceNumber kann eine Response dem entsprechenden Request zugeordnet werden. Sie wird fortlaufend nummeriert. Der Wert der ControlSequenceNumber muss einen Integer-Wert von 0 bis 255 haben.	
	ControlDetail	Das ControlDetail besteht aus folgenden optionalen Attributen:	
		Test Der Test-Status definiert, ob die Information durch eine normale Operation oder durch einen Test verursacht wurde. Es wird ein boolescher Wert angegeben. 0 =NoTest 1 =Test	
		OriginatorCategory	
		OriginatorAddress	
		Time class Die Time class definiert, welche Zeitschritte für den Zeitstempel benutzt werden. Es wird ein ENUM4-Wert angegeben. 0 = reserved 1 = 1ms 2 = 1/10ms 3 = 1/100 ms	
		Time Stamp	
	Cause	Das Attribut Cause gibt Auskunft über die Fehlerursache, die zur Abweisung des Services geführt hat. Cause wird als ENUM angegeben. Folgende Ursachen sind möglich: [0] = not_supported [1] = blocked_by_switching_hierarchy [2] = blocked_by_event [3] = blocked_by_interlocking	

Test-Nr.	Test	Beschreibung	ok
		[4] = blocked_by_setpoint_command [5] = target_exists [6] = parameter_error [7] = time_limit_over [8] = address_error [9] = hardware_error [10] = 1_of_n_control [11] = system_crash [12] = step_limit [13] = command_already_in_execution [14] = plausibility_error [15] = blocked_by_synchrocheck [16] = debounce_active [17] = abortion [18] = parameter_charge_in_execution [19] = CB_alarm	

20.4.5 Befehlssequenzen

Folgende Befehlssequenzen werden auf Ethernet MMS wie nachfolgend beschrieben abgebildet. Bei der Abbildung auf Profibus FMS entfällt jeweils der Service Select. Für die Abarbeitung eines Befehls werden Services aus dem Abschnitt 20.4.4 verwendet. Es soll überprüft werden, dass die Reihenfolge der Befehle korrekt eingehalten wird.

20.4.5.1 Befehle mit normaler Sicherheit

Befehle, die sich nicht auf Schalter beziehen, sind Befehle mit normaler Sicherheit. Dieses sind z.B. Befehle wie:

- LED reset
- Übertragungssperre einrichten
- Kennlinienumschaltung

20.4.5.1.1 Erfolgreiche Ausführung des Befehls

Der Befehl konnte erfolgreich ausgeführt werden.

Test-Nr.	Test	Beschreibung	ok
	Befehlssequenz	Operate Request	
		Operate Response+	
		Report Request	

20.4.5.1.2 Ablehnung des Befehls

Der Befehl wurde abgelehnt. Die Begründung, die zur Ablehnung des Befehls führte, ist in der Antwort Operate Response- enthalten.

Test-Nr.	Test	Beschreibung	ok
	Befehlssequenz	Operate Request	
		Operate Response-	

20.4.5.2 Befehle mit erhöhter Sicherheit

Beschreibung: Befehle, die sich direkt auf Schalter beziehen, sind Befehle mit erhöhter Sicherheit. Dieses ist in OCIS z.B. der Schaltbefehl für den Leistungsschalter.

20.4.5.2.1 Erfolgreiche Ausführung des Befehls

Der Schaltbefehl konnte erfolgreich ausgeführt werden.

Test-Nr.	Test	Beschreibung	ok
	Befehlssequenz	Select with value Request	
		Select with value Response+	
		Operate Request	
		Operate Response+	
		Report Request	
		Command Termination Request+	

20.4.5.2.2 Negatives Ergebnis

Der Schalter wurde mit dem "Select-Service" angewählt. Nach positiver Antwort vom Client wurde vom Server der Schaltbefehl gegeben (Operate Request). Der Server leitet den Schaltbefehl zum Schalter weiter und versendet eine positive Rückantwort auf den Schaltbefehl zum Client. Der Schaltbefehl konnte vom Schalter nicht korrekt durchgeführt werden. Deshalb wurde vom Server ein „Command Termination Request-“ zum Client gesendet. Der Schaltbefehl konnte nicht erfolgreich ausgeführt werden.

Test-Nr.	Test	Beschreibung	ok
	Befehlssequenz	Select with value Request	
		Select with value Response+	
		Operate Request	
		Operate Response+	
		Report Request	
		Command Termination Request-	

20.4.5.2.3 Ablehnung

Der Leistungsschalter wurde mit dem "Select-Service" angewählt. Nach positiver Antwort vom Client wurde vom Server der Schaltbefehl gegeben (Operate Request). Der Server leitet den Schaltbefehl zum Schalter weiter. Aufgrund einer Störung des Schalters wird der Befehl mit dem Service "Operate Response-" zurückgewiesen.

Test-Nr.	Test	Beschreibung	ok
	Befehlssequenz	Select with value Request	
		Select with value Response+	
		Operate Request	
		Operate Response-	

20.4.5.2.4 Abweisung

Der Leistungsschalter wurde mit dem "Select-Service" angewählt. Der Server erkennt eine Störung des Schalters und weist den Befehl direkt ab.

Test-Nr.	Test	Beschreibung	ok
	Befehlssequenz	Select with value Request	
		Select with value Response-	

20.4.6 Generalabfrage

Die nachfolgende Tabelle beschreibt den Aufbau der Generalabfrage im Projekt OCIS

Test-Nr.	Test	Beschreibung	ok
	Vorgang	Die Generalabfrage sollte stattfinden: <ul style="list-style-type: none"> - Nach Anlauf eines Feldgerätes - Nach Ausfall der Kommunikation - Die Variablen von transienten Datenobjekten sollten nicht in der Generalabfrage enthalten sein. 	
	Inhalt	Folgende Variablen werden abgefragt:	
	Allgemeiner logischer Knoten LLN0	LLN0.EnaDisDS LLN0.IEBI LLN0.Ch LLN0.PaSet LLN0.LocDS LLN0.DisDS LLN0.SupAl	
	Logischer Knoten	PTOC1.SupA	

Test-Nr.	Test	Beschreibung	ok
	Überstromzeitschutz	PTOC1.Gal PTOC1.AIA PTOC1.AIB PTOC1.AIC PTOC1.AIN	
	ErdschlussErfassung	PDEF1.ReEF PDEF1.FoEF	
	AWE	RREC1.EnaFct RREC1.ARN	
	Messwert-Aufnahme	MMXU.AphsA_RMS	
	Leistungsschalter	CSWI1.Pos	
	Trenner	CSWI2.Pos	
	Erder	CSWI3.Pos	
	Spannungs-Überwachung	GGIO.PwrSupAlm1 GGIO.PwrSupAlm2 GGIO.PwrSupAlm3	

20.4.7 Modellierung der Common Data Classes

Es soll überprüft werden, ob die jeweiligen "Common Data Classes" die richtigen Attribute enthalten. Die Attribute werden auf Richtigkeit der Variablentypen geprüft. Es wird nicht überprüft, ob der Inhalt logisch richtig ist.

20.4.7.1 Single Point Status SPS

Die "Common Data Class" Single Point Status wird verwendet, um Einzelmeldungen vom Server zum Client zu senden.

Test Nr:	Attribut:	Variablentyp:	ok
	StVal (Status Value)	BOOLEAN	
	q(Quality Descriptor)	SET OF BOOLEAN	
	invalid	BOOLEAN	
	CommFailure	BOOLEAN	
	Blocked	BOOLEAN	
	Substituted	BOOLEAN	
	not Topical	BOOLEAN	
	overflow	BOOLEAN	
	overRange	BOOLEAN	
	badReference	BOOLEAN	
	defaultValue	BOOLEAN	
	t (Time Stamp)	GENERALIZED TIME	
	tq (Time Quality)	SET OF BOOLEAN	
	notStandardTime	BOOLEAN	
	notSynchronized	BOOLEAN	
	invalid	BOOLEAN	
	reasonForStatusChange	ENUM	
	control	ENUM	
	spontaneous	ENUM	
	originator		
	originatorCategory	ENUM	
	NOT_SUPPORTED		
	BAY_CONTR		
	STAT_CONTR		
	REMOTE_CONTR		
	AUT_BAY		
	AUT_STAT		
	SERVICE		
	PROCESS		
	originatorAddress	VSTR32	
	ControlSequenceNumber	INTEGER	

20.4.7.2 Double Point Status DPS

Die "Common Data Class" Double Point Status wird verwendet, um Doppelmeldungen vom Server zum Client zu senden.

Test Nr:	Attribut:	Variablentyp:	ok
	StVal (Status Value)	ENUM	
	TRUE		
	FALSE		
	INTERMEDIATE_STATE		
	BAD_STATE		
	q(Quality Descriptor)	SET OF BOOLEAN	
	invalid	BOOLEAN	
	CommFailure	BOOLEAN	
	Blocked	BOOLEAN	
	Substituted	BOOLEAN	
	not Topical	BOOLEAN	
	overFlow	BOOLEAN	
	overRange	BOOLEAN	
	badReference	BOOLEAN	
	defaultValue	BOOLEAN	
	t(Time Stamp)	GENERALIZED TIME	
	tq(Time Quality)	SET OF BOOLEAN	
	notStandardTime	BOOLEAN	
	notSynchronized	BOOLEAN	
	invalid	BOOLEAN	
	reasonForStatusChange	ENUM	
	control		
	spontaneous		
	originator		
	originatorCategory	ENUM	
	NOT_SUPPORTED		
	BAY_CONTR		
	STAT_CONTR		
	REMOTE_CONTR		
	AUT_BAY		
	AUT_STAT		
	SERVICE		
	PROCESS		

Test Nr:	Attribut:	Variablentyp:	ok
	origintorAddress	VSTR32	
	ControlSequenceNumber	INTEGER	

20.4.7.3 Measured Value MV

Die "Common Data Class" Measured Value wird benutzt, um Messwerte zum Client zu übertragen.

Test Nr:	Attribut:	Variablentyp:	ok
	Value	AV	
	q(Quality Descriptor)	SET OF BOOLEAN	
	invalid	BOOLEAN	
	CommFailure	BOOLEAN	
	Blocked	BOOLEAN	
	Substituted	BOOLEAN	
	not Topical	BOOLEAN	
	overflow	BOOLEAN	
	overRange	BOOLEAN	
	badReference	BOOLEAN	
	defaultValue	BOOLEAN	
	t(Time Stamp)	GENERALIZED TIME	
	tq(Time Quality)	SET OF BOOLEAN	
	notStandardTime	BOOLEAN	
	notSynchronized	BOOLEAN	
	invalid	BOOLEAN	

20.4.7.4 Single Point Control SPC

Die "Common Data Class" Single Point Control wird verwendet um Befehle vom Client zum Server zu senden (Möglichkeiten EIN/AUS).

Test Nr:	Attribut:	Variablentyp:	ok
	ctlVal (Control Value)	ENUM	
	ON		
	OFF		
	StVal (Status Value)	BOOLEAN	
	TRUE		
	FALSE		
	q(Quality Descriptor)	SET OF BOOLEAN	
	invalid	BOOLEAN	

Test Nr:	Attribut:	Variablentyp:	ok
	CommFailure	BOOLEAN	
	Blocked	BOOLEAN	
	Substituted	BOOLEAN	
	not Topical	BOOLEAN	
	overFlow	BOOLEAN	
	overRange	BOOLEAN	
	badReference	BOOLEAN	
	defaultValue	BOOLEAN	
	t(Time Stamp)	GENERALIZED TIME	
	tq(Time Quality)	SET OF BOOLEAN	
	notStandardTime	BOOLEAN	
	notSynchronized	BOOLEAN	
	invalid	BOOLEAN	
	reasonForStatusChange	ENUM	
	control		
	spontaneous		
	originator		
	originatorCategory	ENUM	
	NOT_SUPPORTED		
	BAY_CONTR		
	STAT_CONTR		
	REMOTE_CONTR		
	AUT_BAY		
	AUT_STAT		
	SERVICE		
	PROCESS		
	origintorAddress	VSTR32	
	ControlSequenceNumber	INTEGER	

20.4.8 Uhrzeitsynchronisierung

In den Standardentwürfen IEC 61850 existiert ein Datenobjekt für die Zeitsynchronisation. Dieses Datenobjekt wird dem logischen Knoten LLN0 zugeordnet und kann als Befehl oder nur für Statusmeldungen verwendet werden.

Von UCA wurde ein Modell für die Realisierung von Zeitsynchronisation vorgestellt. Dabei wird ein Measure Request vom Master zum Slave gesendet. Nach Erhalt dieses Signals sendet der Slave eine Measure Response zum Master. Für beide Übertragungen werden die Zeiten gemessen.

Danach wird ein Sync Request vom Master zum Slave gesendet. Dieser sendet eine Sync Response zurück.

Dieses Modell ist allerdings erst in der Diskussionsphase und noch nicht in den Standardentwürfen IEC 61850 übernommen worden.

20.4.9 Performance

- Laufzeiten von Befehlen, Meldungen
Die Ermittlung der Laufzeiten von Befehlen ist möglich. Das Pilotsystem im Projekt OCIS ist jedoch so aufgebaut, dass z.B. Schaltbefehle auf den Leistungsschalter mit Hilfe eines Relais simuliert werden. Reale Leistungsschalter können jedoch schneller schalten als diese Simulation. Die Ermittlung der Laufzeiten kann mit Hilfe der Zeitstempel erfolgen. Es besteht eine Globalforderung, dass die Befehle eine Bearbeitungszeit von 1s nicht überschreiten dürfen.
- Belastung des Netzes
Die Performance kann getestet werden, indem das Netz belastet wird. Dieses kann bei Ethernet realisiert werden, indem zwei PCs in das Netzwerk integriert werden und sich gegenseitig Daten schicken. Dadurch wird das Netz belastet. In diesem belasteten Netz kann danach die Laufzeit der Befehle überprüft werden. Beim Profibus ist die Übertragungszeit konstant, da das Recht zu Senden durch einen Master verteilt wird (deterministisches Bussystem).

20.5 Prüfungen am Gesamtsystem

Das erste Pilotsystem im Projekt OCIS wurde mit Profibus-FMS realisiert. Er bestand aus einem Stationsleitgerät und drei Feldgeräten. Als HMI (Human Machine Interface) wurde die Software WinCC der Firma Siemens verwendet. Die Feldgeräte wurden von den Firmen ABB, ALSTOM und Siemens zur Verfügung gestellt, wobei das Feldgerät der Firma ABB nicht funktionstüchtig war. Die Tests wurden zunächst auf die Kommunikation zwischen dem Stationsleitgerät und den jeweiligen Feldgeräten beschränkt. Im Laufe des Projektes ist geplant das FMS-Protokoll mit Hilfe eines Profibus-Monitors zu überprüfen. Dieser soll in der Lage sein Profibus-FMS-Telegramme sichtbar zu machen.

20.5.1 Störung eines Feldgerätes

Die vor Auftreten der Störung an einem Feldgerät vorhandenen Stellungsmeldungen müssen festgehalten werden und sind in Anzeigen und Archiven als nicht aktualisiert zu kennzeichnen.

- Es dürfen keine verbotenen Zustände auftreten.
- Bei Wiederaufnahme des Betriebs muss das Feldgerät eine automatische Generalabfrage durchführen.
- Die Funktion anderer Feldgeräte darf nicht gestört werden.

20.6 Vergleich Profibus/FMS und Ethernet/MMS

Der Vergleich der Abbildungen (ACSI, SCSM) wird auf Wunsch der Hersteller nicht veröffentlicht.

20.7 Negativtest

Unter einem Negativtest versteht man das absichtliche Herbeiführen von ungewollten Zuständen. Dabei werden die Inhalte von Attributen geändert und die Reaktion des Systems wird beobachtet. Es wird überprüft, ob das System auf vorgegebene Fehlerfälle konform zu IEC 61850 reagiert. Dabei wird jeweils vorgegeben, welche Attribute der Common Data Classes für den entsprechenden Fall betroffen sind und welche Bits auf den Fehlerfall reagieren müssen.

20.7.1 Attribute der Common Data Classes

Dieser Teil wird verwendet um die Attribute der Common Data Classes zu überprüfen.

20.7.1.1 Quality (QUALITY-DESCR)

Der Quality Descriptor hat die Aufgabe, die Eigenschaften des Statuswertes zu beschreiben. Er wird als SET OF BOOLEAN dargestellt. Jedes Attribut in diesem SET OF BOOLEAN stellt eine Eigenschaft des Statuswertes dar. Wird der Statuswert korrekt übertragen, so sind alle Attribute auf FALSE eingestellt. Jedes Attribut stellt einen möglichen Fehlerfall, bezogen auf den Statuswert, dar. Sollte dieser Fehlerfall eingetreten sein, so wird dieses Attribut auf TRUE gesetzt. In der nachfolgenden Tabelle sind die Attribute des Quality Descriptors mit den zugehörigen Fehlerfällen aufgelistet.

Test-Nr.	Test	Beschreibung	ok
	Invalid	Es wird ein Statuswert übertragen, der nicht definiert ist. Invalid wird auf TRUE gesetzt.	
	CommFail	Das Datenobjekt hat keine Kommunikationsverbindung. CommFail wird auf TRUE gesetzt.	
	Substituted	Der Wert des Informationsobjektes wird vom Client für einen Schreibvorgang bereits benutzt. Substituted wird auf TRUE gesetzt.	
	NotTopical	Ein update des Statuswertes während einer vordefinierten Zeit war nicht erfolgreich. NotTopical wird auf TRUE gesetzt.	
	Overflow	Der Value übersteigt die physikalischen Fähigkeiten der Hardware oder die definierten Grenzen der Software. Overflow wird auf TRUE gesetzt.	
	OverRange	Der Wert des Datenobjektes ist jenseits des vordefinierten Wertes. OverRange wird auf TRUE gesetzt.	
	BadReference	Der Statuswert ist ungültig, weil der für die Kalibrierung benutzte Referenzwert ungültig ist. BadReference wird auf TRUE gesetzt.	

20.7.1.2 Time Stamp Quality (TIME-QUAL)

Die Attribute der Time stamp quality stellen die Eigenschaften des Zeitstempels als ein SET OF BOOLEAN dar. Sie sind voreingestellt auf FALSE. Sollte der Zeitstempel fehlerhaft sein, so wird das entsprechende Bit auf TRUE gesetzt. In der nachfolgenden Tabelle sind die Attribute der Time stamp quality mit den zugehörigen Fehlerfällen aufgelistet.

Test-Nr.	Test	Beschreibung	ok
	Not StandardTime	Der Zeitstempel wurde nicht in Sommerzeit angegeben. NotStandardTime wird auf TRUE gesetzt.	
	Not Synchronized	Der Zeitstempel ist nicht synchronisiert. NotSynchronized wird auf TRUE gesetzt.	

	invalid	Der Zeitstempel ist ungültig: <ul style="list-style-type: none"> – Falsches Format (nicht GENERALIZED_TIME) – Zeit befindet sich in der Zukunft – Zeitstempel hat nicht die Zeit in der die Meldung generiert wurde Invalid wird auf TRUE gesetzt.	
--	---------	--	--

20.7.1.3 ReasonForStatusChange (STATUS-CHANGE-REAS)

In diesem Attribut wird angegeben, welche Gründe für die Änderung des Statuswertes bekannt sind. Dabei stehen zwei Möglichkeiten zur Auswahl:

- ? CONTROL: Der Client greift über den Service Operate Request auf den Leistungsschalter zu. Nach Änderung des Statuswertes vom Leistungsschalter sollte überprüft werden, ob das Attribut ReasonForStatusChange den Wert 1 (CONTROL) aufweist.
- ? SPONTANEOUS: Eine Anregung des Überstromzeitschutzes führt zu einer Betätigung des Leistungsschalters. Das Attribut ReasonForStatusChange wird auf SPONTANEOUS gesetzt. SPONTANEOUS wird als 0 dargestellt.

Test-Nr.	Test	Beschreibung	ok
	Value = 0	Die Änderung des Statuswertes wurde durch einen normalen Vorgang verursacht. Das Attribut ReasonForStatusChange wird auf SPONTANEOUS gesetzt. SPONTANEOUS wird als 0 dargestellt.	
	Value = 1	Die Änderung des Statuswertes wurde durch einen Steuervorgang verursacht. Das Attribut ReasonForStatusChange wird auf CONTROL gesetzt. CONTROL wird als 1 dargestellt.	

20.7.1.4 Originator

Im Originator sind Informationen über den Verursacher der Statuswertänderung zu finden. Der Originator wird in drei Bereiche aufgeteilt:

- OriginatorCategory
- OriginatorAddress
- ControlSequenceNumber

20.7.1.4.1 originatorCategory

In der originatorCategory sind mehrere Kategorien aufgelistet, denen der Verursacher der Statuswertänderung zugeordnet werden kann. Jeder Kategorie wird eine Ziffer zugeordnet. Der Datentyp der originatorCategory ist ENUM.

Test-Nr.	Test	Beschreibung	ok
	NOT_SUPPORTED	Der Verursacher ist unbekannt. Der originatorCategory wird der Value 0 zugewiesen.	
	BAY_CONTR	Die Änderung wird von der Feldsteuerung verursacht. Der originatorCategory wird der Value 1 zugewiesen	
	STAT_CONTR OL	Die Änderung wird von der Station verursacht. Der originatorCategory wird der Value 2 zugewiesen	
	REMOTE_CO NTR	Die Änderung wird von der Fernsteuerung verursacht. Der originatorCategory wird der Value 3 zugewiesen	
	AUT_BAY	Der originatorCategory wird der Value 4 zugewiesen	
	AUT_STAT	Der originatorCategory wird der Value 5 zugewiesen	
	SERVICE	Der originatorCategory wird der Value 6 zugewiesen	
	PROCESS	Der originatorCategory wird der Value 7 zugewiesen	

20.7.1.4.2 originatorAddress

Die originatorAddress zeigt die Adresse des Urhebers der Statuswertänderung an. Der Datentyp der originatorAddress ist VSTR32.

Test-Nr.	Test	Beschreibung	ok
	value	Die Adresse des Urhebers wird fehlerhaft angegeben: <ul style="list-style-type: none"> – Entspricht nicht dem Variablentyp VSTR32 – Adresse unbekannt – Adresse stellt nicht den Urheber dar – In der OriginatorCategory muss eine 0 erscheinen, weil der Verursacher unbekannt ist. 	

20.7.1.4.3 controlSequenceNumber

Die controlSequenceNumber zeigt die Nummer des Control-Services, wenn die Änderung des Statuswertes durch einen Control Vorgang verursacht wurde.

Beispiel: Von einer Station werden Befehle zu mehreren Feldgeräten gesendet. Dem ersten Befehl zum Feld 1 wird die controlSequenceNumber 1, dem zweiten Befehl zum Feld 2 die controlSequenceNumber 2 usw. zugeordnet. Die Antworten der Feldgeräte enthalten diese

zugeordnete Zahl. Somit kann die Station zuordnen von welchem Feld die Antwort auf dem Befehl, also die jeweilige Meldung, gesendet wurde.

Test-Nr.	Test	Beschreibung	ok
	value	Die controlSequenceNumber dient als Kennung von Steuerbefehlen. Mit Hilfe der controlSequenceNumber kann eine Response dem entsprechenden Request zugeordnet werden. Sie wird fortlaufend nummeriert. Der Wertebereich der controlSequenceNumber liegt von 0 bis 255.	

20.8 Testszenarien

20.8.1 Leistungsschalter in Störstellung

Folgende Vorgaben sollen gelten: Der Leistungsschalter soll über das Stationsleitgerät eingeschaltet werden. Da es sich um einen Schaltbefehl für den Leistungsschalter handelt, wird ein Befehl mit erhöhter Sicherheit verwendet. Der Leistungsschalter soll vom Stationsleitgerät mit einem Select Request angewählt werden. Das Feldgerät überprüft, ob der Leistungsschalter geschaltet werden kann und bestätigt mit einer Select Response+. Daraufhin schickt das Stationsleitgerät den Schaltbefehl (Operate Request) zum Feldgerät. Das Feldgerät gibt dem Leistungsschalter den Einschaltbefehl und bestätigt den Auftrag, indem es eine positive Rückantwort (Operate Response+) zum Stationsleitgerät gibt. Der Leistungsschalter bleibt beim Einschaltvorgang in der Störstellung. Der negative Ausgang des Schaltbefehls wird dem Stationsleitgerät mit dem Service Command Termination Request– mitgeteilt.

20.8.2 Leistungsschalter blockiert

Nach einer Anregung des Überstromzeitschutzes folgt eine Generalauslösung. Bei der Auslösung blockiert der Leistungsschalter. Die Ausschaltung kann nicht durchgeführt werden.

20.8.3 Doppelter Schaltbefehl

Der Leistungsschalter soll eingeschaltet werden. Dabei wird der Schaltbefehl doppelt versendet. Der erste Befehl muss verarbeitet werden, so dass der Leistungsschalter eingeschaltet wird. Der zweite Schaltbefehl muss bereits nach der Anwahl mit dem Service „Select“ abgewiesen werden.

20.8.4 Falsche Adresse

Der Leistungsschalter soll eingeschaltet werden. Dabei wird die Adresse falsch angegeben. Die Adresse befindet sich in der ObjectIdentification. Der Schaltbefehl muss direkt nach der Anwahl durch den Service „Select“ mit einer Select Response– abgewiesen werden.

20.8.5 Kommunikation unterbrochen

Für die Kommunikation zwischen dem Stationsleitgerät und dem Feldgerät wird ein Lichtwellenleiter verwendet. Die Verbindung des Lichtwellenleiters zum Feldgerät löst sich. Dabei kommt es zu einer Kommunikationsstörung zwischen Stationsleitgerät und Feldgerät.

20.8.6 Netzbelastung

Beim Pilotsystem mit Ethernet MMS wird das Netz belastet. Dieses wird durch zwei PCs realisiert, die in das Netz integriert werden und sich gegenseitig große Mengen an Daten schicken. Dadurch wird das Netz belastet. Im belasteten Netz soll nun die Einhaltung der Laufzeiten von Befehlen überprüft werden.

Im folgenden Kapitel soll anhand eines konkreten Beispiels überprüft, ob ein Schaltbefehl gemäß den Standardentwürfen IEC 61850 ausgeführt wird.

20.8.7 Beispiel eines Schaltvorgangs

Es soll der Leistungsschalter Q0 geschaltet werden. Der Schaltzustand soll von Aus auf Ein geändert werden (Einschalten des Leistungsschalters). Dazu wird vom Client zum Server ein Befehl gesendet, um eine Änderung des Schaltzustandes zu erreichen. Das Verhalten des Feldgerätes auf das folgende Szenario soll überprüft werden.

20.8.7.1 Vorgaben

Test-Nr.	Vorgaben	Beschreibung	ok
	– Control Value = 1	Der Leistungsschalter soll eingeschaltet werden.	
	ObjectIdentification= CSWI1.Pos	– Die Adresse des Leistungsschalters	
	ControlSequenceNumber = 1	– Wenn ein Schaltbefehl ausgeführt wird, wird diesem Befehl eine ControlSequenceNumber zugeordnet, die im Wertebereich von 0 bis 255 liegt.	
	Test = 0	– Das Attribut Test hat den Wert 0. Die Null steht dafür, dass der Wert durch einen Steuervorgang und nicht durch einen Test geändert wird.	
	OriginatorCategory = 2	– Die OriginatorCategory weist den Wert 2 auf, weil die Änderung des Statuswertes durch das Stationsleitgerät durchgeführt wird.	
	Address	– Die Originator Address weist die Adresse des Stationsleitgerätes auf. Er muss dem Datentyp VSTR32 entsprechen.	
	GENERALIZED_TIME	– Der Zeitstempel gibt den Zeitpunkt an, in dem der Befehl generiert wurde. Er muss dem Format GENERALIZED_TIME entsprechen.	

Nach diesen Vorgaben besteht die Möglichkeit, dass der Befehl ausgeführt werden kann oder zurückgewiesen wird. Diese Fälle werden in den nächsten Abschnitten betrachtet.

20.8.7.2 Zurückweisung des Operate Requests

Als erstes werden Fälle betrachtet, die zu einer negativen Antwort (Operate Response-) führen.

- Der gleiche Befehl wird doppelt versendet.
Der erste Operate Befehl wird verarbeitet und führt, wenn keine Fehler vorliegen, zu einer Umschaltung des Leistungsschalters (siehe Beispiele für erfolgreiche Ausführung eines Operate Request). Der zweite Operate-Befehl muss mit einer Operate Response- beantwortet werden. Im Attribut Cause einer Operate Response- wird die Ursache für die Abweisung des Befehls angegeben. Das Attribut enthält den Wert 13 (command_already_in_execution).
- Die Adresse des Leistungsschalters wird falsch angegeben.
Der Operate Befehl wird zurückgewiesen, weil die Adresse des Leistungsschalters fehlerhaft ist. Im Attribut Cause einer Operate Response- wird die Ursache für die Abweisung des Befehls angegeben. Das Attribut enthält den Wert 8 (address_error).

- Die Ausführung des Schaltens im Prozess dauert länger als die im Feldleitgerät parametrisierte Zeit.
Der Operate Befehl wird zurückgewiesen, weil die vordefinierte Zeit für die Ausführung des Befehls überschritten wurde. Im Attribut Cause einer Operate Response– wird die Ursache für die Abweisung des Befehls angegeben. Das Attribut enthält den Wert 7 (time_limit_over).
- Eine Verriegelung ist aktiv (z.B. Trenner ist noch nicht geschlossen). Das Schließen des Leistungsschalters ist unzulässig.
Der Operate Befehl wird zurückgewiesen, weil eine Verriegelung aktiv ist, die die Umschaltung des Leistungsschalters verhindert. Im Attribut Cause einer Operate Response– wird die Ursache für die Abweisung des Befehls angegeben. Das Attribut enthält den Wert 3 (blocked_by_interlocking).
- Es tritt ein Hardwarefehler auf, der das Schalten des Leistungsschalters unmöglich macht.
Der Operate Befehl wird zurückgewiesen, weil ein Hardwarefehler aufgetreten ist. Im Attribut Cause einer Operate Response– wird die Ursache für die Abweisung des Befehls angegeben. Das Attribut enthält den Wert 9 (hardware_error).

20.8.7.3 Erfolgreiche Ausführung des Operate Requests

Der Operate Request kann erfolgreich ausgeführt werden. Es wird eine Operate Response+ vom Server zum Client gesendet. Folgende Überprüfungen an der Operate Response können durchgeführt werden:

Test-Nr.	Inhalt	Beschreibung	ok
	Zurückgesendeter Service = Operate Response+	– Als positive Antwort auf ein Operate Request muss eine Operate Response+ zurückgesendet werden.	
	ObjectIdentification = CSWI1.Pos	– Die ObjectIdentification weist die Adresse des Datenobjektes auf, auf das zugegriffen wurde, also CSWI1.Pos.	
	Status Value = 1	– Der Status Value muss den Wert 1 haben.	
	Zeitstempel= GENERALIZED_TIME	– Der Zeitstempel muss die Zeit enthalten, in der die Meldung generiert wurde.	
	OriginatorCategory = 2	– Die OriginatorCategory muss den Wert 2 enthalten.	
	OriginatorAddress = VSTR32	– Die OriginatorAddress gibt die Adresse des Stationsleitgerätes an.	
	Test= 0	– Das Attribut Test weist den Wert 0 auf.	
	ControlSequenceNumber = 1	– Es wird eine ControlSequenceNumber aus dem Wertebereich 0..255 versendet. Die ControlSequenceNumber muss mit der Nummer übereinstimmen, die für den Operate Request benutzt wurde.	

Mit Versendung der Operate Response+ an den Client signalisiert der Server, dass die Ausführung des Befehls möglich ist und eingeleitet wurde. Nach Ausführung des Schaltvorgangs verschickt der Server ein Report Request zum Client. Ein Report Request wird immer dann versendet, wenn sich ein Statuswert geändert hat. Der Report Request sollte dann den Aufbau haben, der im nächsten Abschnitt dargestellt ist.

20.8.7.4 Report Request

Vorgaben:

- Es soll vom Server an den Client gemeldet werden, dass sich der Statuswert des Leistungsschalters Q0 von 0 auf 1 geändert hat.
- Der Leistungsschalter wurde von der Station geschaltet.
- Der gesendete Report Request muss folgendes enthalten:

Test-Nr.	Inhalt	Beschreibung	ok
	ObjectIdentification= CSWI1.Pos	– Die ObjectIdentification weist die Adresse des Datenobjektes auf, auf das zugegriffen wurde, also CSWI1.Pos.	
	Status Value= 1	– Als Statuswert wird eine 1 übermittelt.	
	Quality-Descriptor= 00000000	– Im Quality-Descriptor stehen alle booleschen Werte auf 000000... (FALSE), weil der Statuswert gültig ist.	
	Zeitstempel= GENERALIZED_TIME	– Der Zeitstempel muss die Zeit enthalten, in der die Meldung generiert wurde.	
	TimeQuality= 000	– In der TimeQuality sind alle drei booleschen Werte auf 000 (FALSE) eingestellt, wenn der Zeitstempel gültig und synchronisiert ist.	
	ReasonForStatusChange= 1	– ReasonForStatusChange weist den Wert 1 auf (CONTROL).	
	OriginatorCategory= 2	– Die OriginatorCategory muss den Wert 2 enthalten.	
	Adresse	– Die OriginatorAddress gibt die Adresse des Stationsleitgerätes an.	

20.9 Prüfungen unter außerordentlichen Betriebsbedingungen

20.9.1 Betrieb bei extremen Temperaturen

Die einwandfreie Funktion des Gerätes muss im erlaubten Temperaturbereich gewährleistet sein.

Im Standard 61850-3 Kapitel 5.2 wird auf IEC 60870-2-2 Kapitel 3 verwiesen.
IEC 60870-2-2: Klassifizierung von klimatischen Umweltbedingungen für Langzeitlagerung und Betrieb

Klasse B3 Temperatur: $5 < T < 40^{\circ}\text{C}$

Temperaturänderungsgeschwindigkeit: $0,5^{\circ}\text{C}/\text{min}$

20.9.2 Betrieb bei Änderung oder Störung der Versorgungsspannung

20.9.2.1 Toleranzbereich bei Versorgung mit Gleichspannung

Die einwandfreie Funktion des Gerätes muss im Toleranzbereich gewährleistet sein.

IEC 60870-2-1 Fernwirkrichtungen und Systeme

Teil 2 : Betriebsbedingungen

Hauptabschnitt 1: Stromversorgung und elektromagnetische Verträglichkeit

4.3.1 Klasse der Gleichspannungs-Grenzabweichungen

Klasse DC3 Spannung: -20% bis $+15\%$ der Nenn-Versorgungsspannung

20.9.2.2 Toleranzbereich bei Versorgung mit Wechselspannung

Die einwandfreie Funktion des Gerätes muss im Toleranzbereich gewährleistet sein.

- Spannung: -15% bis $+10\%$
- Frequenz: $\pm 5\%$
- Oberschwingungsgehalt: $< 5\%$

20.9.2.3 Langsame Änderung der Versorgungsspannung

Die Versorgungsspannung wird innerhalb von ca. 10s ausgehend von ihrem Nennwert bis auf 0 heruntergefahren und wieder zurück. Bei diesem Test dürfen keine ungewollten Betriebszustände auftreten.

20.9.2.4 Schnelle Änderung der Versorgungsspannung

Die Versorgungsspannung wird nach verschiedenen Pulsmustern geändert (wie Flicker). Bei diesem Test dürfen keine ungewollten Betriebszustände auftreten.

20.9.2.5 Unterbrechung der Spannungsversorgung für 1 Minute

Die Spannungsversorgung wird ausgeschaltet und nach 1 Minute wieder eingeschaltet. Bei diesem Test dürfen keine ungewollten Betriebszustände auftreten.

20.9.2.6 Kurze Unterbrechungen der Spannungsversorgung

Mit einem Zufallsgenerator werden kurze Unterbrechungen der Spannungsversorgung von 10 ms erzeugt. Bei diesem Test dürfen keine ungewollten Betriebszustände auftreten.

20.9.3 Störungen oder Unterbrechungen des Stationsbusses

20.9.3.1 Galvanische Verbindung

Unterbrechungen müssen bemerkt werden. Bei diesem Test dürfen keine ungewollten Betriebszustände auftreten.

- Überprüfung der Signalqualität durch Messung des Signal /Rauschabstands
- Die Erhöhung der Bitfehlerrate durch zufällige Verfälschung von Bits
- Erhöhung der Dämpfung der Leiterverbindung
- Störung durch Beschädigung der Leitungs- Verbindung
- Kurzschluss; Isolationsfehler; Unterbrechung einzelner Adern; Übergangswiderstände
- Kontaktprobleme an den Anschlüssen
- Wackelkontakt; Kurzunterbrechung
- Minimale/maximale Leitungslänge

20.9.3.2 Verbindungen mit Lichtwellenleitern

Unterbrechungen müssen bemerkt werden. Bei diesem Test dürfen keine ungewollten Betriebszustände auftreten.

- Überprüfung der Signalqualität durch Messung des Lichtpegels
- Erhöhung der Dämpfung der Lichtwellenleiterverbindung
- Unterbrechungen müssen bemerkt werden.
- Störung durch Beschädigung der Leitungs- Verbindung
- Kontaktprobleme an den Anschlüssen/Steckern
- Überprüfung: Minimale/Maximale Leitungslänge

21 Anhang F: Entwicklung eines Testsystems

21.1 Einführung

Zur Überprüfung der Kommunikation zwischen Geräten der Stations- und Feldebene soll ein Testsystem entwickelt werden, das als Grundlage den Standardentwurf IEC 61850 verwendet. Das Testsystem soll die Möglichkeit bieten, mit Stations- und Feldgeräten zu kommunizieren. Anhand eines Vergleichs zwischen den erfassten Daten und einer Datenbank wird geprüft, ob sich die Geräte konform zu IEC 61850 verhalten. Das Testsystem soll die Simulation einer Feldeinheit (Server-Verhalten) sowie einer Stationseinheit (Client-Verhalten) beinhalten. Da das kommunikative Verhalten nach IEC 61850 der zu simulierenden Stations- und Feldeinheiten im Vordergrund steht und nicht die Simulation eines kompletten Stations- bzw. Feldgerätes, ist es sinnvoll, das Testsystem auf die für die Kommunikation relevanten Funktionen zu beschränken. Für die eigentlichen Prüfungen an den realen Geräten werden entsprechende Module erstellt, die unter dem Oberbegriff Test- und Analysemodule zusammengefasst sind. Das Testsystem soll zusätzlich die Möglichkeit bieten, einen manuell gesteuerten Prozess sowie einen Prozess nach festgelegten Abläufen zu simulieren. Die für das Testsystem benötigten Daten werden aus einer bereits vorhandenen Datenbank abgerufen, die konform IEC 61850 aufgebaut ist. Die Datenbank enthält alle LN, DO, Funktionen und Dienste, die für die Kommunikation benötigt werden, sowie die entsprechenden Beziehungen der Daten untereinander. Um das Testsystem übersichtlich und ausbaufähig zu halten, werden alle Test- und Simulationsfunktionen in verschiedene Module implementiert.

Das Kernstück des Testsystems besteht aus den Modulen Datenbanken, Client, Server und Prozesssimulation. Unter „Test- und Analysemodule“ sind alle Module gruppiert, in denen die verschiedenen Tests und Analysen implementiert sind.

21.2 Analyse

Als ausführbares Programm soll eine **Windows-Applikation (WA)** erstellt werden, in der über die Bedienungsoberfläche die einzelnen Module mit den zugehörigen Funktionen ausgewählt werden können. Weiterhin sollen die Windows-üblichen Standardfunktionen wie „Datei speichern“, „Datei öffnen“ usw. enthalten sein. Die Bedienoberfläche der WA soll Windows-konform gestaltet werden, damit bei der Bedienung intuitiv vorgegangen werden kann.

Das Testsystem soll objektorientiert programmiert werden, da dieser Programmierstil entscheidende Vorteile gegenüber der strukturierten Programmierung aufweist und darüber hinaus die objektorientierte Programmierung in Zukunft überwiegend eingesetzt wird.

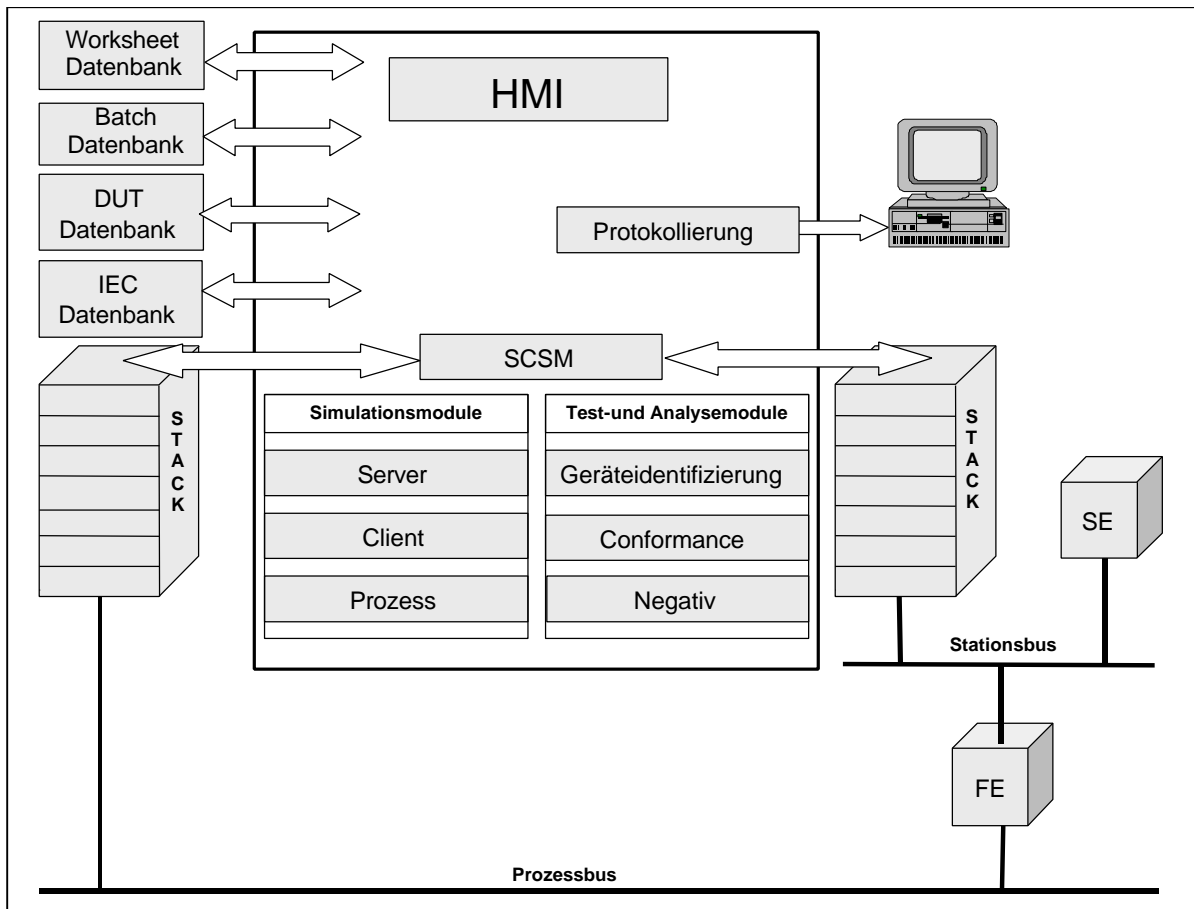


Bild 31 Schematischer Aufbau des Testsystems

Als Programmiersprache wird C++ verwendet, die neben Java einen Standard der objektorientierten Programmierung darstellt. Als Programmierumgebung wird MS-Visual C++ eingesetzt, die sich als sehr leistungsfähige Programmierumgebung in der professionellen Softwareentwicklung etabliert hat. MS-Visual C++ beinhaltet die Microsoft Foundation Classes (MFC) Klassenbibliothek, die ideal zur Programmierung einer WA ist. Das Testsystem soll über das SCSM auf unterschiedliche Stacks aufsetzbar sein. Dadurch, dass keine Abbildung implementiert wird, ist die Applikation offen für zukünftige Kommunikationsstrukturen. Es ist daher für die Erstellung des Testsystems unerheblich, welche Abbildung standardisiert bzw. in Zukunft entwickelt wird. In Abschn. 21.8 ist eine Übersicht über gefundene Anwendungsfälle mit den zugehörigen Beschreibungen gegeben.

21.3 Module des Testsystems

Die folgenden Module sind Bestandteile des Testsystems:

- Server
- Client
- Prozesssimulation
- Datenbankzugriff
- Protokollierung
- Blockschaltbild
- Geräteidentifizierung
- Testszenarien
- Negativtest

21.3.1 Das Modul Server

Zur Darstellung bzw. Simulation der Kommunikation einer Feldeinheit (Server-Verhalten) dient das Modul *Server*. Mit diesem Modul kann das Verhalten einer Stationseinheit (Client-Verhalten) hinsichtlich IEC 61850 getestet werden. Das Modul *Server* kann Befehle von der Stationseinheit empfangen und diese ausführen. Weiterhin können über das Modul *Server* Messwerte zyklisch oder auch auf Anfrage der Stationseinheit hin erfasst und übertragen werden. Die Messwernerfassung greift auf den Prozess zu, der software-technisch nachgebildet wird durch das Modul *Prozesssimulation* (Abschn. 21.3.3). Die Möglichkeit einer realen Prozessanbindung von physikalischen Geräten an das Modul *Server* ist nur sinnvoll, wenn die Kommunikation zwischen Prozess und Feldeinheit getestet wird. Es ist für das kommunikative Verhalten der zu testenden Stationseinheit unerheblich, ob das Modul *Server* an einen „realen“ Prozess oder an eine Simulation angebunden ist.

Funktionen des Moduls *Server* :

- Übertragung des eigenen Datenmodells (Adresse, elektr. Typenschild, ...) nach Aufforderung durch die Stationseinheit.
- Einlesen und Übertragen der Prozessdaten nach Aufforderung durch die Stationseinheit.
- Messwernerfassung zyklisch /spontan.
- Befehle der Stationseinheit empfangen.
- Senden von positiven/negativen Empfangsbestätigungen für Befehle und Bestätigung der Befehlsausführung. Senden von spontanen und zyklischen Meldungen sowie Fehlermeldungen nach spontaner Eingabe durch das **H**uman **M**achine **I**nterface (HMI).

21.3.2 Das Modul Client

Zur Darstellung bzw. Simulation des Kommunikationsverhaltens einer Stationseinheit dient das Modul *Client*. Dieses Modul kann Befehle an die Feldeinheit erteilen und die Statusinformationen über die Befehlsausführung und den Prozess abrufen. Das Modul muss alle Funktionen und Eigenschaften einer Stationseinheit abbilden können, die für die Überprüfung der Kommunikation relevant sind, sowie aus der Abbildung die entsprechenden Dienste zur Verfügung stellen. Mit Hilfe dieser Dienste ist das Modul *Client* in der Lage, Schaltbefehle an die Feldeinheit zu senden, die Ausführung der Befehle zu überwachen und den stationären Zustand der Feldeinheit abzurufen. Die Feldeinheit kann z.B. zur Messwerterfassung angewiesen werden. Die erfassten Messwerte sendet die Feldeinheit dann an das Modul *Client*, das diese dann weiter verarbeiten bzw. über ein Ausgabegerät (z.B. Bildschirm, Drucker) darstellen kann. Für das Modul *Client* ist die Art des Prozesses, den die Feldeinheit erfasst, nicht weiter von Bedeutung. Es kann sich dabei also um eine Prozesssimulation auf Software-Basis oder auch um einen Aufbau mit physikalischen Prüfgeräten handeln. Die Prozessfassung übernimmt die Feldeinheit, welche die Daten des Prozesses umsetzt und an das Modul *Client* IEC 61850-konform übermittelt.

Folgende Funktionen soll das Modul *Client* ausführen können:

- Anfordern von Messwerten von der Feldeinheit.
- Abfragen des Feldes.
- Befehle an die Feldeinheit erteilen.
- Empfangen von Bestätigungen/Meldungen.

21.3.3 Das Modul Prozesssimulation

In diesem Modul soll eine gesteuerte Prozesssimulation realisiert werden. In Teil 9 des zukünftigen Standards IEC 61850 wird die Kommunikation für den Prozessbus standardisiert. Da an den Prozessbus andere Anforderungen gestellt werden, wie an den Stationsbus, kann es von Vorteil sein, unterschiedliche Bussysteme zu verwenden. Das Kommunikationsaufkommen ist beim Prozessbus erheblich höher als beim Stationsbus. Alle Messwerte und Meldungen in müssen so schnell wie möglich übertragen werden, daher ist der Prozessbus zeitkritisch. Für die Feldeinheiten allerdings wäre ein einheitliches Bussystem für den Prozessbus und Stationsbus von Vorteil, da die Feldeinheiten an beiden Bussystemen angebunden sind und Daten austauschen müssen. Unterschiedliche Bussysteme (z.B. Ethernet für den Stationsbus und PROFIBUS für den Prozessbus) würden also eine zusätzliche Umsetzungen erforderlich machen. Das Modul *Prozesssimulation* kann zwei verschiedene Prozessabläufe zur Verfügung stellen. Zum einen ist das ein Prozess, der manuell über das HMI steuerbar ist und zum anderen ein Prozess, der festgelegten Abläufen folgt, die aus einer Datei eingelesen werden.

21.3.3.1 Manuell gesteuerter Prozess

Um einen Prozess zu simulieren, müssen in dem Betriebsmodus „Manuell gesteuerter Prozess“ die einzelnen Elemente, Aktionen und Zustände des Prozesses durch den User festgelegt werden. Über eine Eingabemaske können verschiedene Schalter, Wandler, etc. ausgewählt werden und mit entsprechenden Aktionen (z.B. schalten, messen, übertragen) und Zuständen (z.B. Erdungsschalter geöffnet) belegt werden. Die Feldeinheit muss ein Abbild des Prozesses erstellen können, daher muss das Modul *Prozesssimulation* die Daten des Prozesses zur Verfügung stellen.

Folgende Elemente können für die Prozesssimulation ausgewählt werden:

- Leistungsschalter
- Trennschalter
- Erdungsschalter
- Stromwandler
- Spannungswandler
- Transformator

21.3.3.2 Automatisch gesteuerter Prozess

In diesem Betriebsmodus der Prozesssimulation ist der Prozessablauf festgelegt in Ablauf Tabellen, die in der Batch-Datenbank gespeichert sind. Die Ablauf Tabellen werden von dem Modul *Prozesssimulation* eingelesen und in einen Prozess umgewandelt.

21.4 Datenbanken und Protokollierung

21.4.1 IEC-Datenbank

Die IEC-Datenbank wurde gemäß IEC 61850 von der Firma Siemens erstellt. Die Tabellen der IEC-Datenbank beschreiben die Beziehungen der einzelnen Teile von IEC 61850 untereinander sowie die Funktionen und Aufgaben. Dadurch werden auch die Beziehungen der LNs zu den physikalischen Geräten festgelegt.

21.4.2 DUT-Datenbank

In der **Device Under Test (DUT)** Datenbank werden die Daten der zu prüfenden Geräte gespeichert. Nach der Geräteidentifizierung speichert das Testsystem das eingelesene Datenmodell in der DUT-Datenbank. Im Gegensatz zur IEC-Datenbank ist die DUT Datenbank dynamisch.

21.4.3 Worksheet-Datenbank

Die Worksheet-Datenbank dient der Speicherung bereits erstellter Projektdaten, die auch wieder in das Testsystem zur weiteren Bearbeitung geladen werden können. Diese Datenbank ist ebenfalls dynamisch.

21.4.4 Batch-Datenbank

Die Batch-Datenbank dient der Aufnahme von tabellarischen Ablaufplänen, die einen Prozess, eine Schaltsequenz o.ä. vorgeben. Die Batch-Datenbank soll erweiterbar sein, damit auch neugestaltete Prozessabläufe gespeichert und durch das Modul *Prozesssimulation* durchgeführt werden können. Die Dateneingabe in die Batch-Datenbank erfolgt durch den User.

21.4.5 Das Modul Datenbankzugriff

Das Modul *Datenbankzugriff* ist ein Werkzeug zum gezielten Zugriff auf die unter MS-Access erstellten Datenbanken IEC-Datenbank, DUT-Datenbank, Batch-Datenbank und Worksheet.

Für die IEC-Datenbank erhält das Modul *Datenbankzugriff* nur Lese-Status, da diese Datenbank die in IEC 61850 festgelegten Vereinbarungen und Daten enthält.

Auf die Batch-Datenbank erhält das Modul *Datenbankzugriff* im Simulationsmodus ebenfalls nur Lese-Status, da die Ablauf Tabellen als Vorgabe für die Prozesssimulation dienen und diese nicht verändert werden sollen.

Für die DUT-Datenbank und Worksheet-Datenbank erhält das Modul *Datenbankzugriff* Lese- und Schreibe-Status. In der DUT-Datenbank wird das Datenmodell des zu testenden Gerätes abgespeichert. Für die entsprechenden Test- und Analyseverfahren wird das Datenmodell von den Test- und Analysemodulen angefordert und bei Bedarf aktualisiert. Die Datenbank Worksheet dient der Sicherung der in Arbeit befindlichen Projekte mit den zugehörigen Testparametrisierungen.

21.4.6 Das Modul Protokollierung

Das Modul *Protokollierung* sorgt für eine Aufzeichnung der erhaltenen Testergebnisse und eingetretener Ereignisse. Eine Aufzeichnung erfolgt in einer Textdatei oder einer Datenbank. Eine zusätzliche optionale Ausgabe auf einem Drucker oder/und Ausgabe in einem separaten Fenster soll vorgesehen werden.

21.5 Das Modul Blockschaltbild

Das Modul *Blockschaltbild* dient der graphischen Darstellung des angebundenen Prozesses. In der erzeugten Darstellung durch das Modul *Blockschaltbild* sollen alle Komponenten des

Prozesses enthalten sein. Zur Erstellung eines Blockschaltbildes können drei Verfahren angewendet werden:

- Manuelle Auswahl der Komponenten durch den Anwender.
- Automatische Generierung nach vorheriger Abfrage der Feldeinheit.
- Einlesen aus einer bestehenden Datei.

21.6 Das Modul Geräteidentifizierung

Das Modul *Geräteidentifizierung* dient der Abfrage, was für ein Gerät vorhanden ist sowie der Sicherung des Datenmodells, welches das Gerät zur Verfügung stellt. Nach der Identifizierung des Gerätes (Auslesen der IP-Adresse und des elektronischen Typenschildes) wird abgefragt, welche Dienste, LNs und DOs vorhanden sind. Das komplette Datenmodell wird dann in der **Device Under Test**-Datenbank gespeichert. Die Abfrage soll nach folgendem Ablaufplan erfolgen:

- Abfrage, was für ein Gerät ist vorhanden?
- Einlesen der Adresse sowie des elektronischen Typenschildes des gefundenen Gerätes.
- Einlesen der Dienste (services), LN und DO des Gerätes.
- Speichern des ermittelten Datenmodells in der DUT-Datenbank.

21.7 Das Modul Testsznarien

21.7.1 Einführung

Konformität (Conformance)/Interoperabilität (Interoperability)

Unter Konformität wird verstanden, dass der Istzustand eines Gerätes oder einer Anwendung einen definierten Sollzustand einnimmt. Als Interoperabilität wird die Fähigkeit von Geräten oder Anwendungen bezeichnet, miteinander sinnvoll zu kommunizieren. Durch Systeme, die interoperabel sind, werden Überwachungs- und Steuerungsfunktionen, Prozessdatenerfassung und Datenübertragung ermöglicht, die herstellerunabhängig sind, da die Schnittstellen standardisiert sind. Hierbei braucht der Anwender sich nicht um die Kommunikation der Anwendung zu kümmern, da diese auf einem einheitlichen Standard abläuft. In diesem Fall muss ein Gerät konform zu IEC 61850 sein.

Austauschbarkeit (Interchangeability)

Die Kriterien Interoperabilität und Konformität führen nicht zwangsläufig zu Geräten, die austauschbar sind. Damit die Austauschbarkeit gewährleistet ist, müssen weitere Bedingungen wie gleiche Abmessungen, Befestigungen, Anschlüsse usw. ebenfalls standardisiert bzw. gleich sein.

21.7.2 Konformitätsüberprüfung

Das Modul *Testsznarien* erstellt ein IEC 61850-konformes Vergleich-Datenmodell aus der IEC-Datenbank. In diesem Datenmodell sind alle Dienste, LN und DO enthalten, die auch in

dem zu testenden Gerät enthalten sein müssen. Die Attribute der einzelnen DO werden klassifiziert in vorgeschriebene (mandatory) und zusätzliche, nicht vorgeschriebene (optional) Attribute. Das Vergleich-Datenmodell und das DUT-Datenmodell werden jetzt auf Übereinstimmung geprüft. Dabei ist zu beachten, dass für alle vorgeschriebenen Attribute eine Übereinstimmung vorhanden sein muss. Die optionalen Attribute werden dahingehend geprüft, ob diese Attribute vorhanden sein dürfen (ausgenommen von diesem Vergleich sind die herstellerspezifischen Attribute, die explizit als solche gekennzeichnet sein müssen). Sollte dies der Fall sein, ist das Gerät zumindest vom Inhalt des Datenmodells her konform zu IEC 61850 und kann weiteren Prüfungen unterzogen werden. Ist keine Übereinstimmung der Datenmodelle festzustellen, sind die Abweichungen dahingehend zu prüfen, ob der Betrieb des zu prüfenden Gerätes negativ beeinflusst wird. Positive wie negative Ergebnisse werden von dem Modul *Testszzenarien* an das Modul *Protokollierung* (Abschn. 21.4.6) übergeben.

Ablaufplan:

- Geräteidentifizierung
- Einlesen des DUT-Datenmodells aus der DUT-Datenbank.
- Erstellung eines Vergleich-Datenmodells unter Verwendung der IEC-Datenbank.
- Aufteilung der Attribute in „mandatory“ und „optional“.
- Überprüfung der Daten und der Datenstruktur des DUT-Datenmodells auf Vollständigkeit.
- Vergleich der „mandatory“ Attribute: Sind die vorgeschriebenen Attribute des DUT-Datenmodells und die des Vergleich-Datenmodells identisch?
- Herausfiltern der optionalen, herstellerspezifischen Attribute, die nicht verglichen werden sollen.
- Vergleich der „optional“ Attribute: Sind die optionalen Attribute des DUT-Datenmodells in dem Vergleich-Datenmodell vorhanden?
- Bei positiver Bestätigung der Vergleiche senden der Ergebnisse an Modul *Protokollierung*. Bei negativer Bestätigung prüfen, ob das Datenmodell in der DUT-Datenbank fehlerhaft ist (zu viele, undefinierte Daten, abgewandelte Daten, Datentypen). Senden der Ergebnisse an Modul *Protokollierung*.

21.7.3 Kommunikations- und Sequenzprüfung

Die Kommunikationsprüfung umfasst alle Befehle und Meldungen, die das DUT-Gerät verstehen und umsetzen muss. Das Modul *Testszzenarien* lädt dazu die Ablauftabellen aus der Batch-Datenbank, in denen alle möglichen Befehlsabläufe (positive und negative) enthalten sind. Diese Befehlsabläufe werden unter Nutzung eines Simulationsmoduls (Modul *Client*, wenn es sich bei dem DUT-Gerät um eine Feldeinheit handelt und Modul *Server*, wenn es sich bei dem DUT-Gerät um eine Stationseinheit handelt) abgearbeitet und die Reaktionen des DUT-Gerätes dahingehend geprüft, ob die Befehle und Meldungen richtig interpretiert wurden und ob die Sequenz der Befehle und Meldungen richtig erkannt und umgesetzt wurde. Die Ergebnisse dieser Prüfung werden an das Modul *Protokollierung* (Abschn. 21.4.6) weitergeleitet.

21.7.4 Das Modul Negativtest

Der Zweck dieses Moduls ist eine Untersuchung des Verhaltens eines DUT auf verfälschte Daten bzw. Befehle. Mögliche Fehlerquellen können sich im Bereich der Datenübertragung sowie in den Datentelegrammen selbst befinden. Zur gezielten Verhaltensanalyse eines DUT werden in Negativtests bewusst falsche Daten übergeben oder falsche Befehle/Rückantworten erteilt. Danach wird die Reaktion eines DUT aufgezeichnet und dahingehend geprüft, ob sie den Vereinbarungen in IEC 61850 entspricht.

Einige mögliche Negativtests sind:

- Bestätigungsmeldungen werden nicht versendet.
- Fehlerhafte Bestätigungsmeldungen werden versendet.
- Bestätigungsmeldung ohne vorherige Anforderung wird gesendet.
- Spontane Schalthandlung.
- Befehle/Bestätigungsmeldungen mit fehlerhafter Datenstruktur werden gesendet.

21.8 Übersicht der Anwendungsfälle

In Bild 32 ist eine Übersicht über alle Anwendungsfälle, die in einem ersten Ansatz für das Testsystem aufgestellt wurden gegeben, ohne die Beziehungen zwischen den Anwendungsfällen zu berücksichtigen. In (Abschn. 21.8.1) werden die Anwendungsfälle detailliert, mit ihren Beziehungen untereinander und den Beziehungen zu ihren Akteuren dargestellt.

- Anwendungsfall 1.1-Server simulieren: Der Anwendungsfall Server simulieren, beinhaltet die Anwendungsfälle Befehle Empfangen und Eigendaten bereitstellen. Ein Server nach IEC 61850 muss das Datenmodell, welches durch die vorgegebenen Geräte, bzw. die enthaltenen LNs, oder im Fall eines Stationsleitgeräts durch die enthaltenen LDs gegeben ist einem anfragenden Client zur Verfügung stellen. Der simulierte Server muss auf gesendete Befehle des Clients in vorgegebener Weise reagieren und die angeforderten Daten zur Verfügung stellen, oder auf Anforderung des Clients, zur Wertänderung, reagieren. Voraussetzung für eine Reaktion ist die Berechtigung eines Clients auf den Server zuzugreifen, bzw. Werte zu ändern. In jedem Fall muss der Server eine Rückmeldung senden (Response). Diese Rückmeldung kann positiv (Response+) oder negativ (Response-) ausfallen, abhängig von der Zugriffsberechtigung oder der Möglichkeit die geforderte Aktion auszuführen.
- Anwendungsfall 1.1.1-Befehle empfangen: Befehle müssen über die zugrundeliegende Kommunikationseinrichtung empfangen werden, protokolliert werden, zwischengespeichert und ausgewertet werden. Anhand der Auswertung muss eine folgerichtige Reaktion auf den gesendeten Befehl erfolgen, die ebenfalls zur späteren Auswertung protokolliert werden muss.
- Anwendungsfall 1.1.2-Eigendaten bereitstellen: Auf Anforderung müssen die Eigendaten, wie z.B. IP-Adresse, elektronisches Typenschild, vorhandene Dienste und vorhandene Datenobjekte, übermittelt werden.
- Anwendungsfall 1.2-Client simulieren: Der Anwendungsfall der Client-Simulierung beinhaltet unter Anderem die Anwendungsfälle Datenmodell anfordern, Messwert

anfordern, Befehl senden und Datenmodell ändern. Bei der Simulierung eines Client muss die Möglichkeit bestehen, Daten von einem Server anzufordern oder Zustände in einem Datenmodell zu ändern. Die Möglichkeit der Messwerterfassung und die Reaktion auf zyklisch übertragene Messwerte sowie die Überprüfung auf eingegangene Meldungen oder Quittungen muss vorhanden sein.

- Anwendungsfall 1.2.1-Datenmodell anfordern: Das komplette Datenmodell eines Servers wird angefordert und zwischengespeichert, damit es zu Vergleichsaufgaben zur Verfügung steht.
- Anwendungsfall 1.2.2-Messwert anfordern: Ein Messwert wird von einem Server zur Datenübertragung angefordert.
- Anwendungsfall 1.2.3-Befehl senden: Ein Befehl, der zur Durchführung oder Anforderung einer Aktion oder Datenübertragung, gesendet wird.
- Anwendungsfall 1.2.4-Datenmodell ändern: Der Anwendungsfall Datenmodell ändern wird durch Befehl senden (**Fehler! Verweisquelle konnte nicht gefunden werden.**) erweitert, wenn der Benutzer (User) das Datenmodell manuell ändert. Sonst werden Änderungen des Datenmodell durch den zugrundeliegenden Prozess oder durch das Testsystem ausgelöst.
- Anwendungsfall 1.3-Prozessabbild bereitstellen: Das Abbild eines Prozesses wird zur Verfügung gestellt. Der Prozess kann sowohl simuliert werden, oder aber von realen Geräten gebildet werden.
- Anwendungsfall 1.4-Datenbank abfragen/schreiben: Das Testsystem muss mit Hilfe von Structured Query Language-Befehlen (SQL, Datenbankabfragesprache) auf Datenbanken zugreifen können, um Daten in definierter Form in eine Datenbank schreiben oder auslesen zu können.
- Anwendungsfall 1.5-Prozess simulieren: Die Prozesssimulation muss ein Datenabbild bereitstellen. Dieses Datenabbild beinhaltet Meldungen, Schalter, Transformatoren, Messwerte und Schutzfunktionen. Der Prozessablauf bzw. der Eintritt von Ereignissen kann von Hand beeinflusst werden oder anhand eines in einer Datei gespeicherten Ereignisablaufs vorgegebenen werden.
- Anwendungsfall 1.6-Datei speichern: Dateien bzw. die darin enthaltenen Daten können zur späteren Verwendung oder Bearbeitung gespeichert werden. Datenspeicherung kann entweder in einer Datenbank erfolgen oder aber in einer sequenziellen Textdatei, in Form von ASCII-Zeichen.
- Anwendungsfall 1.7-Daten zwischenspeichern: Es muss eine Möglichkeit geben, Daten temporär zwischenspeichern, um z.B. den Arbeitsspeicher bei großen Datenmodellen und eventuell notwendigen Vergleichen zu entlasten.
- Anwendungsfall 1.8-Vergleichsdatenmodell erzeugen: Anhand der Identifizierten Geräte (**Fehler! Verweisquelle konnte nicht gefunden werden.**) muss ein Vergleichsdatenmodell anhand der IEC-Datenbank erzeugt werden, um einen Vergleich der Datenmodelle durchführen zu können.
- Anwendungsfall 1.9-Geraet identifizieren: Dieser Anwendungsfall umfasst das Auslesen des elektronischen Typenschildes und die Feststellung der Geräteart. Die Geräteart kann z.B. anhand einer Datenbankabfrage bestimmt werden und wird z.B. bei einer grafischen Darstellung zur Auswahl der entsprechenden Symbol-Grafikdatei benötigt.

- Anwendungsfall 1.10-Quittierung anfordern/ausgeben: Für Dienste müssen Quittierungen ausgegeben werden. Der Rückgabewert kann positiv (Response+) oder negativ (Response-) sein. Diese Quittierungen müssen auch von dem Testsystem in der Server-Betriebsart ausgegeben werden. In der Client-Betriebsart muss auf eine Bestätigung gewartet, bzw. bei falschen oder fehlenden Quittierungen entsprechend z.B. durch erneute Anforderung reagiert werden.
- Anwendungsfall 1.11-VirtualDevice hinzufügen/entfernen: Dieser Anwendungsfall findet sowohl in der Betriebsart Server, als auch in der Betriebsart Prozesssimulation Verwendung. Hierbei kann ein Gerät der Simulation hinzugefügt werden oder aus den vorhandenen Geräten kann ein Gerät entfernt werden.
- Anwendungsfall 1.12-Datenmodell überprüfen: In diesem Anwendungsfall wird ein aus einem Server ausgelesenes und zwischengespeichertes Datenmodell mit den Vorgaben aus der IEC-Datenbank verglichen. Bei Abweichungen der Datenmodelle durch nicht vorhandene optionale Daten oder hinzugefügten herstellerspezifischen Daten, wird eine Warnung ausgegeben. Wenn notwendige Daten fehlen, wird abgefragt, ob der Testvorgang weitergeführt werden soll, sonst wird der Testvorgang abgebrochen.
- Anwendungsfall 1.13-Prozessvorgabe hinzufügen/ändern: Dieser Anwendungsfall existiert nur bei der Betriebsart Prozesssimulation. Durch den Benutzer können Vorgaben für einen simulierten Prozess hinzugefügt oder geändert werden.
- Anwendungsfall 1.14-Ereignisprotokoll führen: Der Anwendungsfall einer Ereignisprotokollführung umfasst die Aufzeichnung aller eingetretenen Ereignisse, versandten Befehlen und den erfolgten Reaktionen. Dieses Ereignisprotokoll kann zur Dokumentation durchgeführter Testläufe dienen. Die Anwendungsfälle Datei speichern (**Fehler! Verweisquelle konnte nicht gefunden werden.**), Daten zwischenspeichern (**Fehler! Verweisquelle konnte nicht gefunden werden.**) und Datenbank abfragen/schreiben (**Fehler! Verweisquelle konnte nicht gefunden werden.**) werden mitbenutzt.
- Anwendungsfall 1.15-Konfiguration durchführen: Der Anwendungsfall Konfiguration durchführen beinhaltet verschiedene Szenarien. Zur Konfiguration zählen sowohl die Grundeinstellungen des Testsystems, wie z.B. Arbeitsverzeichnisse für Dateien festlegen, die Parametrisierung von simulierten Geräten und Prozessen und auch die Festlegung von Ereignissen.
- Anwendungsfall 1.16-Meldungen anzeigen: Der Anwendungsfall der Meldungsanzeige beinhaltet die Informationsausgabe an den Benutzer. Wichtige Meldungen werden über eine einheitlich aufgebaute Dialogbox an den Benutzer weitergegeben.
- Anwendungsfall 1.17-Fehlerliste abarbeiten: Im Anwendungsfall der Fehlerlistenabarbeitung wird eine vor dem Testbeginn festgelegte Fehlerliste seriell abgearbeitet. Die Fehlerliste kann Fehler wie z.B. fehlende Quittierung, falsche Reaktion auf Befehle oder doppelte Ausgabe von Befehlen beinhalten.
- Anwendungsfall 1.18-Event auslösen: Der Anwendungsfall der Eventauslösung kann entweder durch den Benutzer durchgeführt werden (manuelle Ereignisauslösung oder durch eine dem Testsystem vorgegebene Ereignisliste durchgeführt werden. Dieser Anwendungsfall umfasst Ereignisse wie z.B. Messwertänderung, Schutzauslösung oder Kommunikationsstörung.

- Anwendungsfall 1.19-Modul auswählen: Der Prozess der Modulauswahl wird durch den Benutzer (User) durchgeführt und legt die Betriebsart des Testsystems auf einem PC fest.

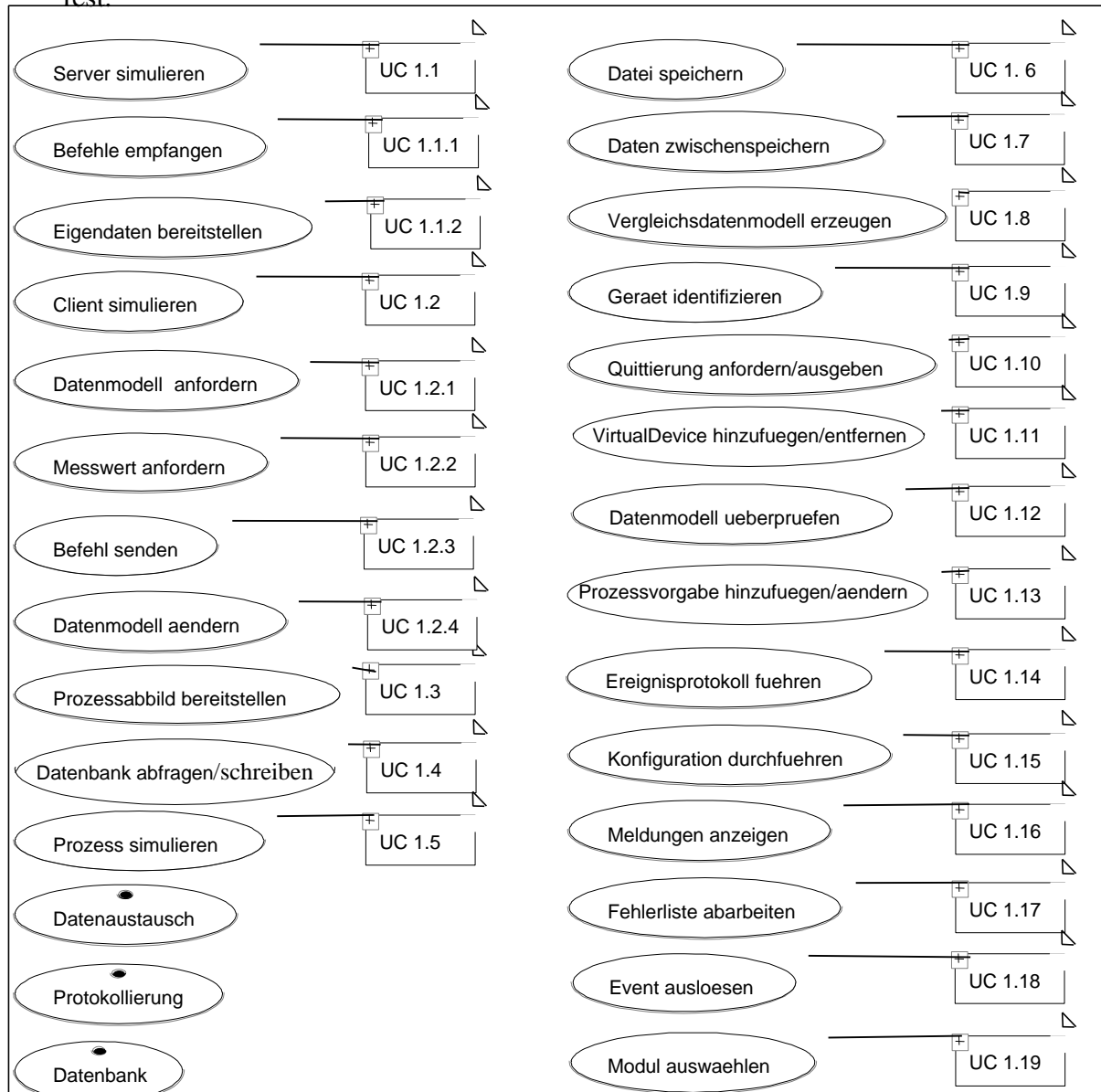


Bild 32 Übersicht Anwendungsfälle

Die Anwendungsfälle Datenaustausch, Protokollierung und Datenbank sind als abstrakt gekennzeichnet, weil sie als Oberbegriff für unterschiedliche Mechanismen stehen. Der Anwendungsfall Datenaustausch z.B. steht für die Kommunikation über ein Netzwerk mit Hilfe von MMS oder FMS welcher aber nicht Bestandteil des Testsystems ist, sondern von externen Stacks zur Verfügung gestellt wird. Auch die Datenbank wird z.B. nur durch das Testsystem abgefragt, aber unter anderen Anwendungen (z.B. MS-Access) erstellt. Die Protokollierung kann auf einem externen Drucker ausgegeben werden oder/und in eine Datei geschrieben werden.

21.8.1 Details der Anwendungsfälle

In Bild 33 ist eine grobe Übersicht über die Funktionen und die Kommunikation des Testsystems in einem Anwendungsfall-Diagramm dargestellt.

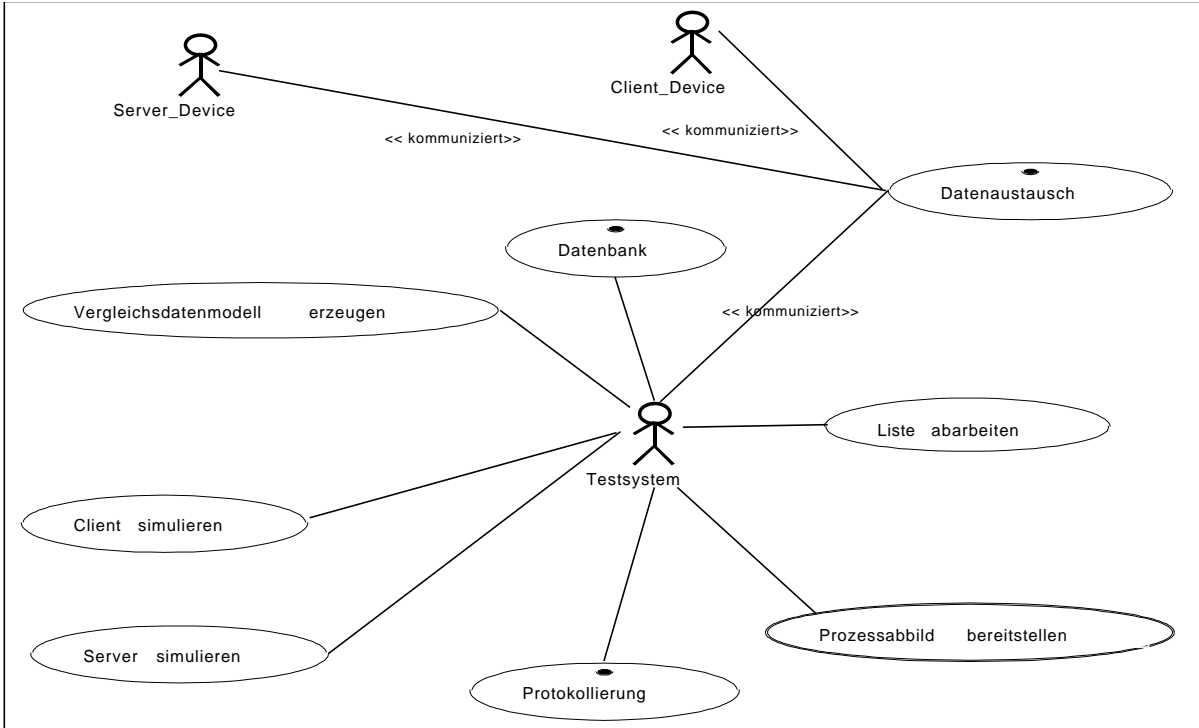


Bild 33 Anwendungsfalldiagramm Kommunikation des Testsystems

In Bild 34 sind die zugrundeliegenden Prozesse und Funktionen bei der Simulation eines Clients als Anwendungsfall-Diagramm dargestellt.

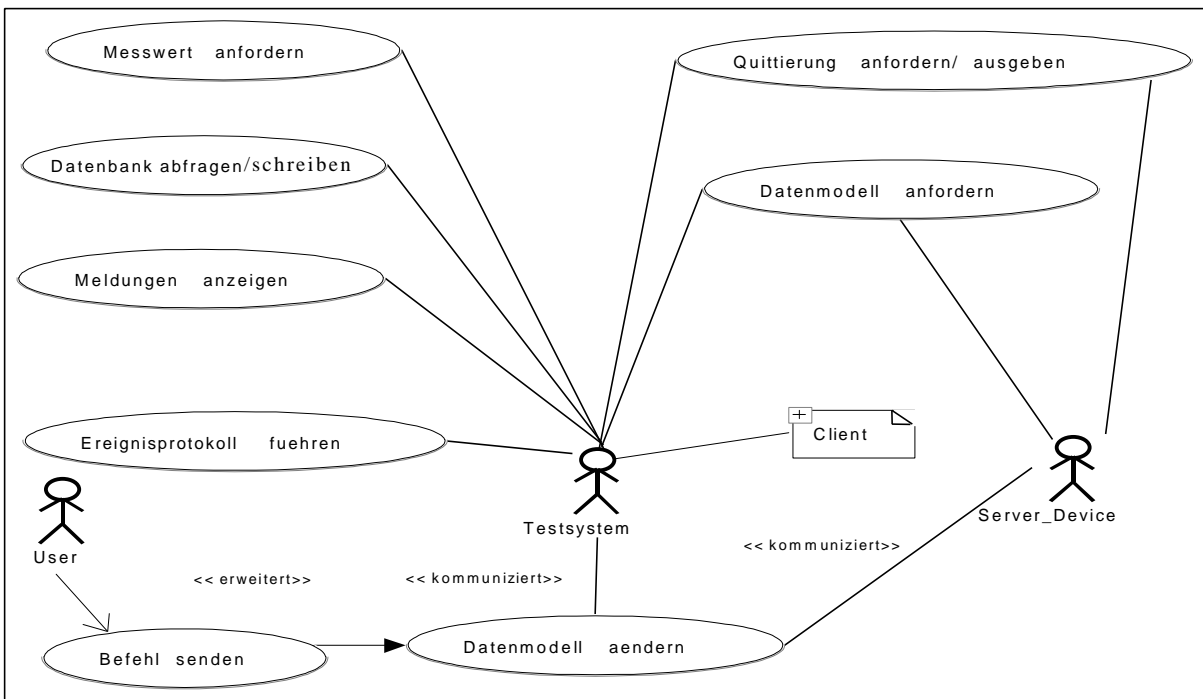


Bild 34 Anwendungsfalldiagramm Client-Simulation

In Bild 35 sind die zugrundeliegenden Prozesse und Funktionen bei der Simulation eines Servers als Anwendungsfalldiagramm dargestellt.

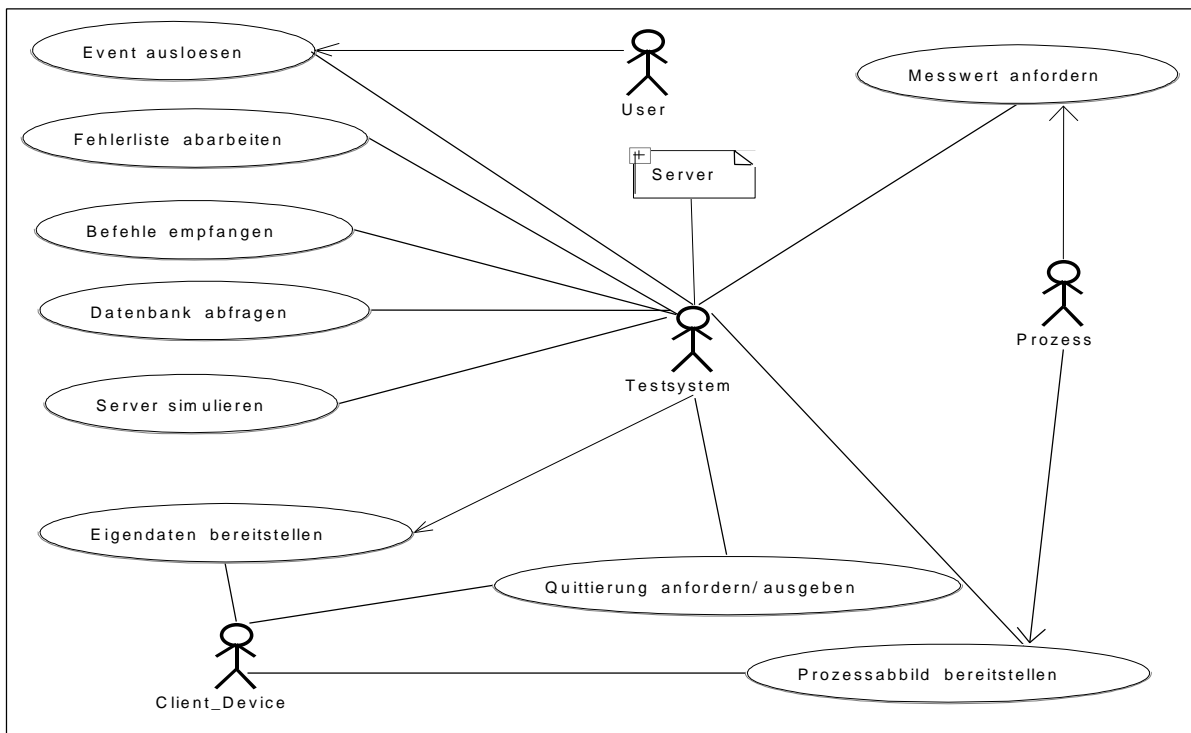


Bild 35 Anwendungsfalldiagramm Server-Simulation

In Bild 36 ist eine Übersicht über die zugrundeliegenden Prozesse und Funktionen des Testsystems, in Verbindung mit der Client-, Server- und Prozesssimulation dargestellt.

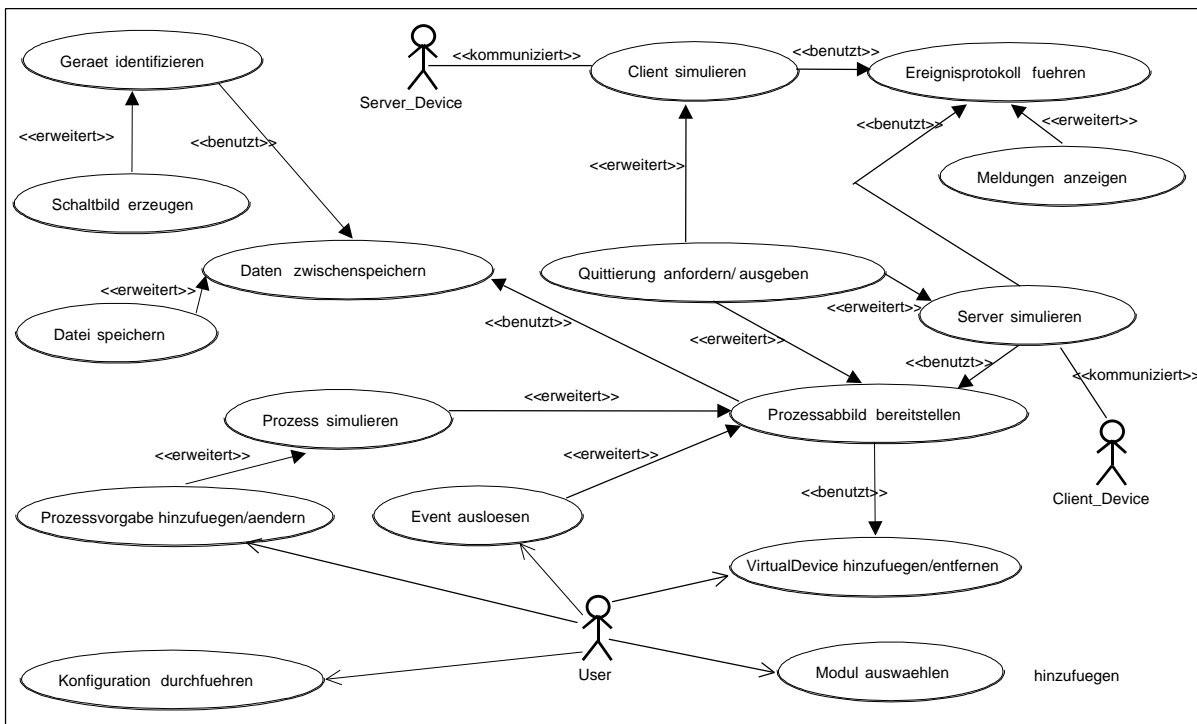


Bild 36 Anwendungsfalldiagramm für das Testsystem

22 Anhang G: ISO/OSI Referenzmodell

22.1 Architektur des ISO/OSI Referenzmodells

Damit zwei Computersysteme miteinander kommunizieren können, bedarf es genauer Spezifikationen, die die Details und den Ablauf der Kommunikation festlegen. Die internationale Normungsorganisation „International Standardisation Organisation (ISO)“ hat innerhalb der Norm ISO 7498 [17] ein grundlegendes Modell beschrieben, das den Aufbau der technischen Kommunikation von offenen Systemen enthält: das ISO/OSI Referenzmodell „Open Systems Interconnection (OSI)“. Unter den genannten offenen Systemen versteht man Systeme, deren Kommunikationsschnittstellen nach außen und die Schnittstellen der physikalischen Systemkomponenten herstellerneutral genormt sind. Dadurch wird gewährleistet, dass normkonforme Systeme untereinander kommunizieren können, d.h. interoperabel sind. Interconnection stellt innerhalb OSI im wesentlichen die Übertragung von Informationen zwischen kommunizierenden Systemen dar.

Der Prozess der Kommunikation wird durch das ISO/OSI Referenzmodell in sieben logisch aufeinander aufbauenden Schichten (engl. Layer) strukturiert (Bild 37). Eine Schicht (Layer) besteht aus einer oder mehreren Instanzen (engl. Entity). Eine Instanz führt innerhalb einer Schicht die ihr zugedachten Arbeiten aus. Dabei stützt sich eine (n)-Instanz jeweils auf die Leistungen der (n-1)-Instanz [18]. Jede Schicht erbringt somit Leistungen für die darüberliegende Schicht. Eine Ausnahme bildet die Schicht sieben als höchste Schicht im Referenzmodell. Sie erbringt ihre Dienste unmittelbar dem Anwendungsprozess [19].

Für die Durchführung einer Kommunikation müssen die Instanzen Daten austauschen. Der Austausch erfolgt immer zwischen den Instanzen derselben Ebene (Partnerschichten Instanz, Peer Layer Entity) in Form von Protokolldateneinheiten (Protocol Data Unit, PDU). Diese logische Kommunikation auf der horizontalen Ebene ist Gegenstand der Normung. Der wirkliche Weg der Datenübertragung verläuft natürlich in vertikaler Richtung von System I durch das physikalische Medium zu System II. Die Kommunikation in der vertikalen Ebene ist herstellerspezifisch. Durch das Referenzmodell sind lediglich die Aufgaben jeder einzelnen Schicht fest umrissen. Auch die Protokolle (Abschn. 22.3, S.246) der jeweiligen Schichten werden nicht vom Referenzmodell vorgeschrieben, daher müssen ergänzend zum ISO/OSI Referenzmodell Protokolle für die einzelnen Schichten in eigenen Normen festgelegt werden [20].

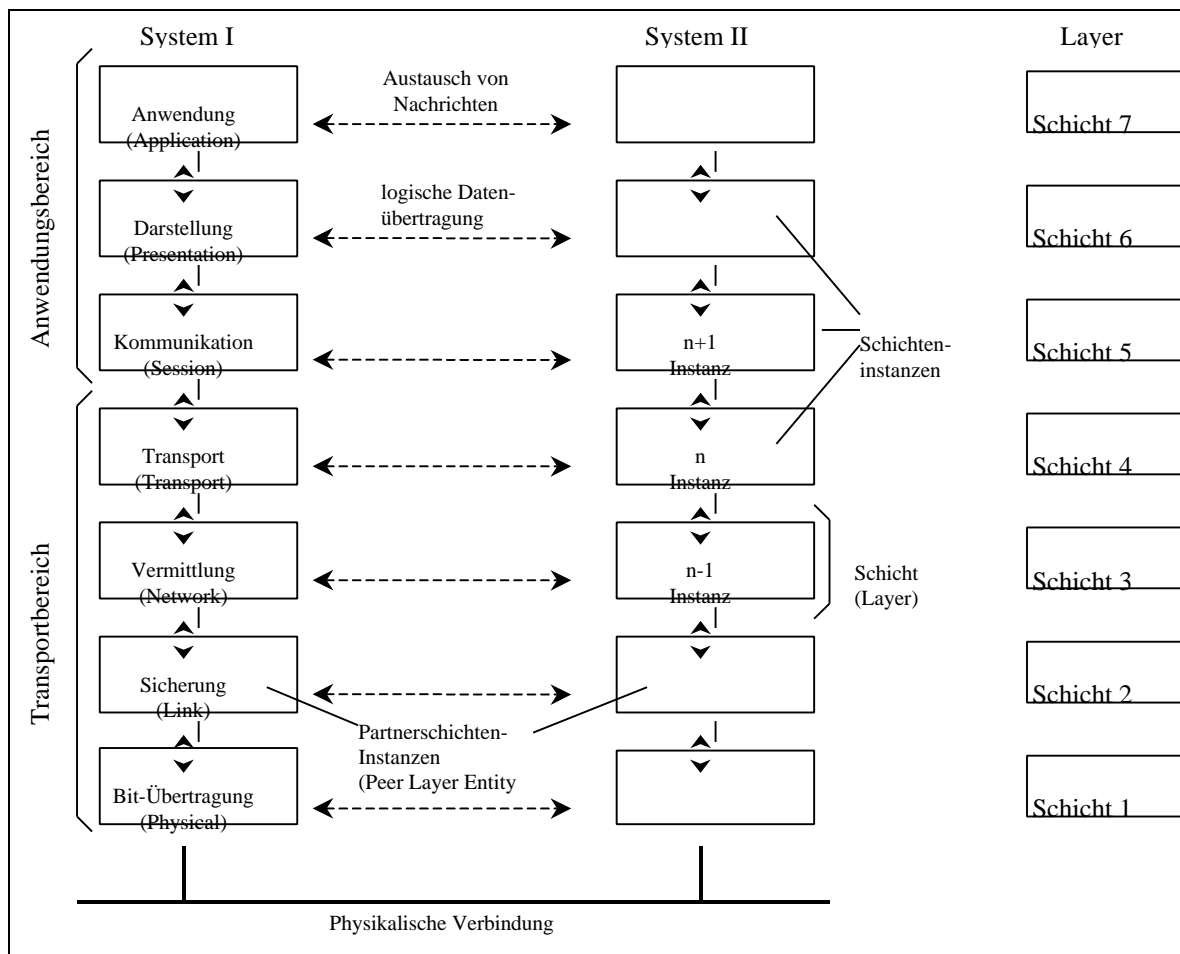


Bild 37 Das sieben Schichten ISO/OSI Referenzmodell

22.2 Funktionen der Schichten

Schicht 1: Physical Layer

Sie stellt die Verbindung eines Systems zum physikalischen Medium dar und ermöglicht so die Übertragung einzelner Informationseinheiten. Weiterhin werden die elektrischen (z.B. Signalverlauf) und die mechanischen (z.B. Stecker, Kabel) Anforderungen festgelegt sowie die Übertragungsart (z.B. duplex/halbduplex, synchron/asynchron). Der Physical Layer aktiviert und deaktiviert die Übertragung auf Anforderung der Schicht zwei und meldet dieser entdeckte Fehler [18, 21].

Schicht 2: Link Layer

Dieser Layer stellt die Übertragung durch den Physical Layer zwischen zwei Systemen sicher. Weiterhin wird die Synchronisation zwischen Sender und Empfänger eingestellt. In lokalen Netzen sorgt eine Subebene innerhalb dieser Schicht zusätzlich für den Zugang zum Übertragungsmedium [18]. Zu diesem Zweck ist der Link Layer in die zwei Teilschichten LLC (Logical Link Control) und MAC (Media Access Control) gesplittet.

Schicht 3: Network Layer

Sollen Endsysteme miteinander kommunizieren, die nicht unmittelbar miteinander verbunden sind, so müssen die Daten gegebenenfalls über Transitsysteme zum Empfänger weiter vermittelt werden. Die Hauptaufgabe dieser Schicht ist somit der Aufbau eines Pfades zwischen zwei Endsystemen (Wegwahl, Routing). Eine weitere Aufgabe ist die Fehlererkennung und Signalisierung an die darüberliegende Schicht vier [18, 21].

Schicht 4: Transport Layer

Die Schicht vier nimmt die Anforderungen der Anwendungsprozesse hinsichtlich der Übertragungsqualität entgegen und stellt sicher, dass eine zuverlässige Ende zu Ende Datenübertragung angeboten wird, unabhängig von den Eigenschaften und der Zuverlässigkeit des zugrunde liegenden Netzes, das möglicherweise viele Teilnetze umfasst. Eine wichtige Funktion ist deshalb die Fehlerbehandlung von Daten, die über Teilnetze gesendet werden, die weniger zuverlässig sind. Diese Schicht stellt fünf Klassen (0 bis 4) zur Verfügung. Die Klasse vier stellt dabei alle Eigenschaften zur Verfügung, die nötig sind, um Fehler bei der Übermittlung zu entdecken und zu beheben [22]. Der Transport Layer kann prinzipiell auf jedes beliebige Netz (Schicht 1 bis 3) aufsetzen, hier macht sich die modulare Schichtenstruktur bemerkbar. Weiterhin bietet diese Schicht eine Flusskontrolle zwischen Empfänger und Sender. Der Sender erhält dazu einen Kredit, der ihm mitteilt, wie viele Daten er senden darf, bevor er eine Quittung vom Empfänger erhält. Hat er die vereinbarte Menge gesendet, muss er warten, bis eine weiterer Kredit zugeteilt wird. Damit werden Verstopfungen auf der Verbindung verhindert. Zudem wird eine Segmentierung langer Nachrichten unterstützt.

Schicht 5: Session Layer

Diese Schicht stellt dem Application Layer zum einen Dienstelemente zur allgemeinen Dialogsteuerung zwischen kommunizierenden Systemen zur Verfügung. Zum anderen ermöglicht diese Schicht Wiederaufsetzpunkte in den Datenfluss einzusetzen und bei Bedarf, z.B. wenn der Datenfluss aufgrund einer Störung unterbrochen wurde, die Kommunikation auf einen dieser Aufsetzpunkte zurückzusetzen [23]. Diese Methode hilft bei der Vermeidung von unnötigen Wiederholungen der bisher gesendeten Daten.

Schicht 6: Presentation Layer

Damit alle an einer Kommunikation beteiligten Systeme Daten austauschen können, müssen sie eine gemeinsam verständliche Sprache sprechen. Dazu müssen zwei kooperierende Anwendungen den formalen Aufbau ihrer auszutauschenden Daten kennen. Jede Anwendung für sich betrachtet, beschreibt den formalen Aufbau ihrer Daten (PDU) in der sogenannten Typdefinition. Die Typdefinition erfolgt im Rahmen der Programmentwicklung. Die Summe aller Typdefinitionen wird als abstrakte Syntax bezeichnet. Sie ist für jede Anwendung spezifisch. Die gemeinsame Sprache entspricht nun der abstrakten Syntax, die zwischen den kooperierenden Systemen identisch sein muss. Die abstrakte Syntax einer Anwendung kombiniert mit den Codierregeln des verwendeten Compilers ergeben die lokale, konkrete Syntax. Die lokalen Syntaxen weichen aufgrund unterschiedlicher Interna eines jeden Systems meist voneinander ab (Programmiersprache, Compiler etc.). Um es trotz

unterschiedlicher lokaler, konkreter Syntaxen zu einer sinnvollen Zusammenarbeit zwischen den Anwendungsinstanzen kommen zulassen, müssen die o.g. unterschiedlichen lokalen Syntaxen in eine von beiden Systemen verständliche, gemeinsame und konkrete Syntax, die Transfersyntax, konvertiert werden. Diese Aufgabe übernimmt in der Schicht sechs je ein Encoder/Decoder. Er bedient sich dabei sogenannter lokaler Codierregeln, der Local Encoding Rules (LER). Sie werden bei bekanntem lokalem Compiler durch die abstrakte Syntax und die Transfersyntax bestimmt [23, 24]. Von der ISO wurde zur formalen Beschreibung von Datentypen eine allgemeine Datenbeschreibungssprache für abstrakte Datentypen geschaffen, die Abstract Syntax Notion One (ASN.1) [25]. Die zuvor genannten Codierregeln sind systemabhängig. Zu ASN.1 passend existieren nun systemunabhängige Codierregeln, die Basic Encoding Rules (BER), die ihrerseits die ASN.1 Typbeschreibungen in die entsprechende Transfersyntax umsetzen.

Schicht 7: Application Layer

Der Application Layer ist die höchste Schicht im ISO/OSI Referenzmodell. Er stellt prinzipiell den Anwendern die Kommunikationsdienste eines Systems zur Verfügung. Unterstützung erhält die Schicht sieben dabei von der darunterliegenden Schichten eins bis sechs. Die Programme greifen als Anwendung auf systemunabhängige Dienste zurück, die ihrerseits von entsprechenden Anwendungsinstanzen ausgeführt werden [22]. Um die systemunabhängigen Dienste als Anwendungen des Systems überhaupt nutzen zu können, ist eine konkrete Schnittstelle zwischen Anwendungsinstanz und der Anwendung erforderlich. Diese Schnittstelle ist eindeutig systemabhängig, da sie u. a. über systemkonforme Schnittstellenaufrufe Zugriff auf systemspezifische Betriebsmittel (z.B. Speicher, Festplatte etc) gewährleisten muss. Die programmtechnische Realisierung dieser Schnittstelle wird Ap am Interfac at (Bild 38).

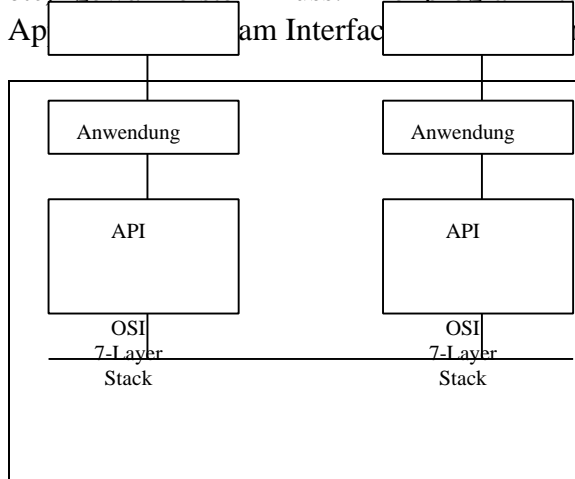


Bild 38 Schnittstelle zwischen OSI Stack und Anwendung: das API

Nicht jede der sieben Schichten des OSI Referenzmodells ist bei bestimmten Netztopologien notwendig. Bei erhöhten Leistungsanforderungen an die Kommunikation wird daher auf bestimmte Schichten verzichtet. Vor allem bei einfachen Topologien sind keine Vermittlungsfunktionalitäten (Network Layer) notwendig. Auch die Funktionalitäten des Transport und Session Layers sind oft entbehrlich oder ihre Aufgaben werden von den Anwendungen übernommen. Dadurch entstand das EPA Modell (Enhanced Performance

Architecture), ein Model mit erhöhter Leistungsfähigkeit, dass nur die Schichten eins, zwei und sieben beinhaltet.

22.3 Protokolle und Profile

Protokolle sind allgemeingültige Verhaltensrichtlinien, auf deren Grundlage zwei Systeme (z.B. Client und Server) kommunizieren und vor allem sich auch verstehen können. Zum einen wird ein Transportprotokoll benötigt (Abschn. 22.2, S.243), welches sicherstellt, dass zwei Systeme miteinander in Verbindung treten können. Zum andern ein Anwendungsprotokoll, damit die Systeme, die miteinander kommunizieren wollen, die gleiche Sprache sprechen. Die allgemein benannten Anwendungs- und Transportprotokolle (Bild 37, Anwendungs- und Transportbereich) bestehen wiederum aus den Protokollen der einzelnen Schichten des ISO/OSI Referenzmodells. Damit nicht mehrere Systeme gleichzeitig ihre Daten übertragen, wurden Zugangsverfahren entwickelt, die von den Transportprotokollen benutzt werden.

Das Token Verfahren

Bei diesem Verfahren wird ein Token von einem System zum nächsten weitergereicht. Das Token kann entweder ‚frei‘ oder ‚belegt‘ signalisieren. Erhält ein sendewilliges System ein Frei-Token, wird es von diesem System in ein Belegt-Token umgewandelt und die zu übermittelnde Nachricht angehängt. Das Datenpaket wird nun solange von System zu System weitergereicht, bis es beim Empfänger angelangt ist. Dieser nimmt die Nachricht auf und schickt das Token zurück zum Sender, der dann den fehlerfreien Empfang erkennen kann. Jetzt erst wird das Token wieder frei. Die Geschwindigkeit ist beim Token-Verfahren immer konstant. Auch wenn keine Sendekonflikte entstehen können, ist dieses Verfahren relativ langsam.

Das CSMA/CD Verfahren (Carrier Sense Multiple Access with Collision Detection)

Im Gegensatz zum Token-Verfahren wird hierbei zunächst die Leitung abgehört. Ist diese frei, beginnt ein System mit dem Versand einer Nachricht. Senden aber im gleichen Augenblick (im Bereich von Millisekunden) weitere Systeme so entsteht ein Sendechaos. In diesem Fall stellen alle System das Senden ein, und jedes System fängt nach einer jeweils für sich zufällig ermittelten Wartezeit wieder an zu senden. Alle Systeme müssen den Versand solange wiederholen, bis ein fehlerfreier, konfliktfreier Versand vom empfangenen System bestätigt worden ist. Ein auf CSMA/CD basierendes Transportprotokoll ist daher vor allem bei Netzwerken mit wenig sendewilligen Systemen sehr schnell. Mit zunehmender Anzahl sendewilliger Systeme steigt jedoch die Häufigkeit von Sendekonflikten und damit auch Wiederholungsversuchen, so dass die Übertragungsgeschwindigkeit absinken kann. Ethernet ist ein Vertreter dieses Verfahrens.

Für unterschiedliche Anwendungsbereiche sind verschiedenartige Protokollalternativen durch die von ISO vorgegebene Kommunikationsarchitektur möglich (Bild 39). Für jede Schicht des ISO/OSI Referenzmodells stehen also mehrere, verschiedenartige Protokolle (Schichtenprotokolle) zur Verfügung. Z.B. existieren für Schicht zwei des UCA OSI-Stack die Protokolle LLC, HDLC LAPB und ISDN LAPD.

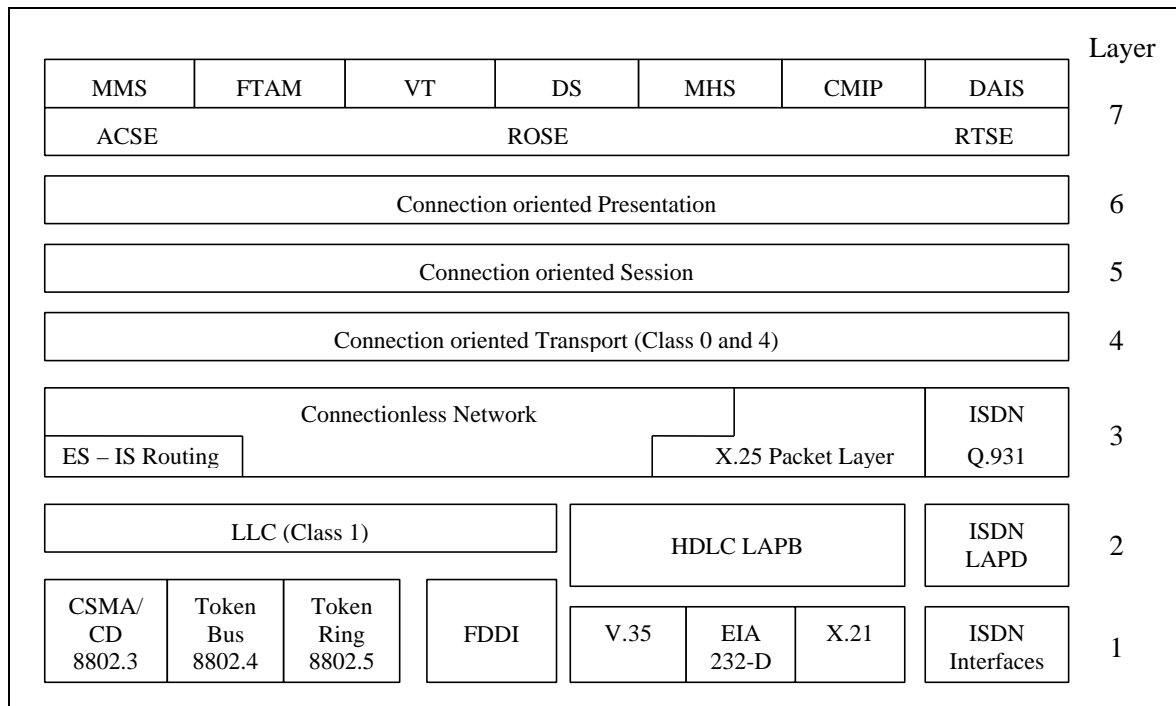


Bild 39 UCA Schichtenprotokolle (7-Layer UCA Modell)

Liegt z.B. für zwei Systeme I und II eine konkrete gemeinsame Kommunikationsaufgabe vor, so wird es erforderlich, aus der allgemeingültigen Kommunikationsarchitektur eine auf den konkreten Anwendungsfall und damit verbunden auf die konkreten Kommunikationsanforderungen speziell zugeschnittene Kommunikations- bzw. Protokollarchitektur zu erstellen. Eine solche zugeschnittene Architektur wird Profil genannt (Bild 40). Ein Profil legt also ein Protokoll pro Schicht des ISO/OSI Referenzmodells fest. Im folgenden Bild (Bild 40) ist ein Beispiel für ein Profil grau hinterlegt.

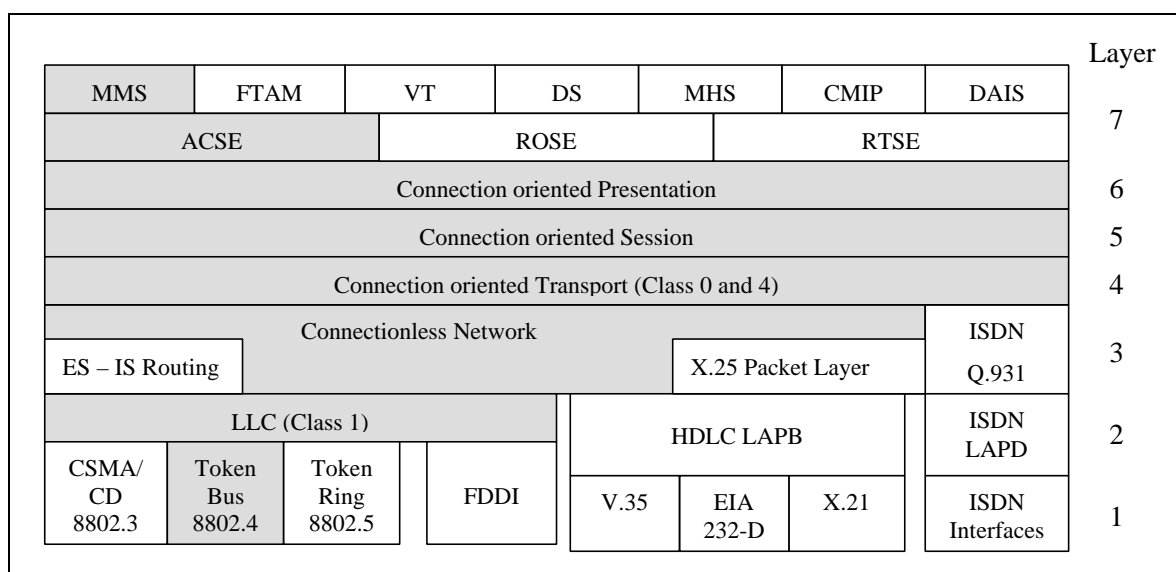


Bild 40 Profil eines UCA 7-Layer Stack

Die Schicht sieben besteht aus zwei Instanzen. Aus dieser Schicht wurden für das Profil die Protokolle MMS und ASCE ausgewählt. Die Schichten sechs, fünf und vier bieten für dieses Beispiel nur ein Protokoll an. In der Schicht drei soll ein verbindungsloses Protokoll gefahren werden, daher wurde das Protokoll ‚Connectionless Network‘ ausgewählt. Verbindungslos bedeutet, dass keine Empfangsbestätigung, nach dem Senden einer Nachricht, vom empfangendem System an das sendende System zurückgegeben wird.

Profile ändern nicht die Standards, sondern zeigen das Zusammenwirken von Standards, die zusammen verwendet werden müssen, um eine spezielle Funktion zu ermöglichen. Des Weiteren beschreiben sie gegebenenfalls einschränkende Maßnahmen hinsichtlich der Parameter, Zeitkriterien und weiterer Optionen für den ausgewählten Anwendungsfall [26, 21]. Grundsätzlich wird ein System mit zugeschnittener Kommunikationsarchitektur nach ISO/OSI in ein Anwender- und Transportsubsystem mit den entsprechenden Anwender- und Transportprofilen unterteilt (Bild 37).

23 Anhang H: MMS

Für die Fabrikautomatisierung hat General Motors (GM) die Realisierung eines einheitlichen industriellen Kommunikationsnetzes mit dem Projekt Manufacturing Automation Protocols (MAP) seit der achtziger Jahre verfolgt. Mit dem von GM initiierten MAP Projekt wurde ein fertigungstechnisch orientierter Kommunikationsstandard angestrebt, mit dem die Maschinen in der Fertigung modelliert werden sollten. Durch das Festschreiben der seinerzeit bekannten Funktionen in einem Standard wäre die Anwendung bei genau solchen Systemen ermöglicht worden, die dem damaligen Entwicklungsstand entsprochen hätten. Im Rahmen der international verfolgten Aktivitäten wurde seinerzeit entschieden, nicht eine bestehende Technologie zu standardisieren, sondern vielmehr die Modelle, Methoden und Verfahren, die geeignet sind, auch mit der Zeit sich ändernde Anforderungen zu erfüllen. So entstand der ISO/IEC Standard Messaging Manufacturing Specific (MMS, ISO 9506). Der MMS Standard definiert allgemeine informationstechnische Funktionen bzw. Objekte für verteilte Automatisierungssysteme [21]. Der Begriff Manufacturing, der für das erste M in MMS steht, war unglücklich gewählt worden, denn der Standard enthält prinzipiell keine Manufacturing-spezifischen Definitionen. Der MMS Standard ist ein allgemein gültiger Industriestandard, und er bildet eine Plattform für eine Vielzahl von Anwendungen.

MMS kann auf allen gängigen Kommunikationssystemen realisiert werden, die einen gesicherten Transport von Daten unterstützen. Das können Netze wie TCP/IP oder ISO/OSI auf Ethernet, ein Feldbus oder einfache Punkt zu Punkt Verbindungen wie HDLC, RS 485 oder RS 232 sein. MMS ist unabhängig von einem sieben Layer Stack (Kap. 21), d.h. unabhängig davon, wie die Kommunikation unterhalb von MMS realisiert wird. Die MMS Basisdokumente wurden 1990 vom ISO TC 184 (Industrial Automation) und IEC TC 65 (Process Control Systems) in der ersten Version gemeinsam veröffentlicht. Die ergänzenden Standards für spezielle Anwendungen folgten später. Die sechs Teile sind:

- Part 1: ISO/IEC 9506-1 Services [27]
- Part 2: ISO/IEC 9506-2 Protocol [28]
- Part 3: ISO/IEC 9506-3 Companion Standard for Robots [29]
- Part 4: ISO/IEC 9506-4 Companion Standards for Numeric Control [30]
- Part 5: ISO/IEC 9506-5 Companion Standards for Programmable Logic Controller [31]
- Part 6: ISO/IEC 9506-6 Companion Standards for Process Control [32]

24 Literatur Anhang

- [17] ISO 7498; OSI Basic Reference Model, 1984.
- [18] Brill,M.: Die anwendernahen Schichten im ISO/OSI-Modell. 1.Teil, Elektronik 5/4.3.1988.
- [19] Link, V.; März W.: etz-Report 27: Normung der Prozeßdatenkommunikation – Einführung. Berlin; Offenbach: VDE-Verlag GmbH, 1992.
- [20] Walter, W.: Heterogene Netze. Addison–Wesley, 1993.
- [21] etz-Report 28: Offene Kommunikationsplattformen für die Leittechnik nach IEC 60870-6 am Beispiel der Netzleittechnik. 2. Aufl. – Berlin; Offenbach: VDE-Verlag GmbH, 1998.
- [22] Henshall, J.; Shaw, S.: OSI praxisnah erklärt. München, Wien: Hanser Verlag; London: Prentice-Hall Internat., 1992.
- [23] Kerner, H.: Rechnernetze nach OSI. Addison-Wesley 2.Aufl., 1993.
- [24] Brill,M.: Die anwendernahen Schichten im ISO/OSI-Modell. 3.Teil, Elektronik 7/31.3.1988.
- [25] D. Steedman: ASN.1 The Tutorial and Reference, Technology Appraisals, ISBN 1 871802 06 7.
- [26] Beyer, W.: Das ISO/OSI Referenzmodell und seine Anwendung in der Fernwirk- und Prozeßdatenkommunikation. ÖZE, Jg. 43, Heft 4, April 1990.
- [27] Manufacturing Messages Specification (MMS) – Part 1 Service Definition ISO International Standard ISO/IEC 9506-1, 1990.
- [28] Manufacturing Messages Specification (MMS) – Part 2 Protocol Definition ISO International Standard ISO/IEC 9506-2, 1990.
- [29] Manufacturing Messages Specification (MMS) – Part 3 Companion Standard for robotics ISO/IEC 9506-3, 1991.
- [30] Manufacturing Messages Specification (MMS) – Part 4 Companion Standard for numerical control ISO/IEC 9506-4, 1992.
- [31] Manufacturing Messages Specification (MMS) – Part 5 Companion Standard for programmable controllers ISO/IEC CD 9506-5, 1993.
- [32] Manufacturing Messages Specification (MMS) – Part 6 Companion Standard for process control ISO/IEC 9506-6, 1993.