

DIN EN 61968-11

ICS 33.040.20; 35.240.99

Ersatz für
DIN EN 61968-11:2011-04
Siehe Anwendungsbeginn

**Integration von Anwendungen in Anlagen der Elektrizitätsversorgung –
Systemschnittstellen für Netzführung –
Teil 11: Erweiterungen des allgemeinen Informationsmodells (CIM) für
die Verteilung
(IEC 61968-11:2013);
Englische Fassung EN 61968-11:2013**

Application integration at electric utilities –
System interfaces for distribution management –
Part 11: Common information model (CIM) extensions for distribution
(IEC 61968-11:2013);
English version EN 61968-11:2013

Intégration d'applications pour les services électriques –
Interfaces système pour la gestion de distribution –
Partie 11: Extensions du modèle d'information commun (CIM) pour la distribution
(CEI 61968-11:2013);
Version anglaise EN 61968-11:2013

Gesamtumfang 271 Seiten

DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik im DIN und VDE

Anwendungsbeginn

Anwendungsbeginn für die von CENELEC am 2013-04-10 angenommene Europäische Norm als DIN-Norm ist 2014-02-01.

Für DIN EN 61968-11:2011-04 besteht eine Übergangsfrist bis 2016-04-10.

Nationales Vorwort

Vorausgegangener Norm-Entwurf: E DIN EN 61968-11:2013-04.

Für dieses Dokument ist das nationale Arbeitsgremium K 952 „Netzleittechnik“ der DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik im DIN und VDE (www.dke.de) zuständig.

Die enthaltene IEC-Publikation wurde vom TC 57 „Power systems management and associated information exchange“ erarbeitet.

Das IEC-Komitee hat entschieden, dass der Inhalt dieser Publikation bis zu dem Datum (stability date) unverändert bleiben soll, das auf der IEC-Website unter „<http://webstore.iec.ch>“ zu dieser Publikation angegeben ist. Zu diesem Zeitpunkt wird entsprechend der Entscheidung des Komitees die Publikation

- bestätigt,
- zurückgezogen,
- durch eine Folgeausgabe ersetzt oder
- geändert.

Für den Fall einer undatierten Verweisung im normativen Text (Verweisung auf eine Norm ohne Angabe des Ausgabedatums und ohne Hinweis auf eine Abschnittsnummer, eine Tabelle, ein Bild usw.) bezieht sich die Verweisung auf die jeweils neueste gültige Ausgabe der in Bezug genommenen Norm.

Für den Fall einer datierten Verweisung im normativen Text bezieht sich die Verweisung immer auf die in Bezug genommene Ausgabe der Norm.

Der Zusammenhang der zitierten Normen mit den entsprechenden Deutschen Normen ergibt sich, soweit ein Zusammenhang besteht, grundsätzlich über die Nummer der entsprechenden IEC-Publikation. Beispiel: IEC 60068 ist als EN 60068 als Europäische Norm durch CENELEC übernommen und als DIN EN 60068 ins Deutsche Normenwerk aufgenommen.

Das Präsidium des DIN hat mit Präsidialbeschluss 1/2004 festgelegt, dass DIN-Normen, deren Inhalt sich auf internationale Arbeitsergebnisse der Informationsverarbeitung gründet, unter bestimmten Bedingungen allein in englischer Sprache veröffentlicht werden dürfen. Diese Bedingungen sind für die vorliegende Norm erfüllt.

Da sich die Benutzer der vorliegenden Norm der englischen Sprache als Fachsprache bedienen, wird die Englische Fassung der EN 61968-11 veröffentlicht. Zu deren Abschnitten 1 bis 5 wurde eine Übersetzung angefertigt und als informativer Nationaler Anhang NA der vorliegenden Norm hinzugefügt. Für viele der verwendeten Begriffe in Abschnitt 3 existieren keine gebräuchlichen deutschen Benennungen, da sich die deutschen Anwender in der Regel ebenfalls der englischen Benennungen bedienen. Diese Norm steht nicht in unmittelbarem Zusammenhang mit Rechtsvorschriften und ist nicht als Sicherheitsnorm anzusehen.

Änderungen

Gegenüber DIN EN 61968-11:2011-04 wurden folgende Änderungen vorgenommen:

- a) Einführung neuer Klassen, um die flexible Namensgebung von identifizierten Objekten zu unterstützen (neue Klassen in der CIM-Grundversion, IEC 61970-301, erhältlich),
- b) Einführung neuer Klassen, um den Austausch von Einlinienschaltbildern zu unterstützen (neue Klassen in der CIM-Grundversion, IEC 61970-301, erhältlich),
- c) konsolidierte Übertragungs- und Verteilungsmodelle für Leitungen, Transformatoren, Schalter-, Sensor- und andere Zusatzgeräte (einige Klassen der Ausgabe 1 wurden leicht verändert und von der DCIM-Version zur CIM-Grundversion, IEC 61970-301, übertragen; weitere, neue Klassen sind in der CIM-Grundversion, IEC 61970-301, erhältlich),

- d) Unterstützung von Definitionen der separaten Phasen, die typischerweise für die Modellierung von unbalancierten Netzwerke benötigt werden (neue Klassen in der CIM-Grundversion, IEC 61970-301, erhältlich),
- e) Unterstützung zeitweiser Änderungen im Netzwerk durch Modelle von Abschaltungen, Brücken (Jumper) und Klemmen (neue Klassen in der CIM-Grundversion, IEC 61970-301, erhältlich),
- f) flexibles Modell für Organisationen und ihre Rollen,
- g) Unterstützung von Koordinatensystemen bei der Beschreibung geographischer Positionen,
- h) Unterstützung der Konfiguration bei Nachverfolgung von Ereignissen,
- i) einfaches Modell für Anlagen (Assets) und Anlagenlisten (Asset Catalogues),
- j) Unterstützung von Verbindungen zwischen netzwerkorientierten Modellen und liegenschaftsorientierten (Metering) Modellen,
- k) Unterstützung von Netzbetriebsmitteln in Wohnumfeld.

Frühere Ausgaben

DIN EN 61968-11: 2011-04

Nationaler Anhang NA (informativ)

1 Anwendungsbereich

Dieser Teil von IEC 61968 spezifiziert die Erweiterungen des allgemeinen Informationsmodells (CIM) für die Verteilung, welche in IEC 61970-301 spezifiziert sind. Er definiert eine grundlegende Reihe von Erweiterungen des allgemeinen Informationsmodells (CIM), welche die Nachrichtendefinition nach IEC 61968-3 bis IEC 61968-9, IEC 61968-13 und IEC 61968-14¹⁾ unterstützen. Der Anwendungsbereich der vorliegenden Norm erstreckt sich auf das Informationsmodell, welches das Basis-CIM für die Bedürfnisse von Verteilnetzen erweitert, und auch auf die Integration in unternehmensweiten Informationssystemen, welche üblicherweise innerhalb von Anlagen zur Elektrizitätsversorgung angewendet werden. Das Informationsmodell ist in UML definiert, welche eine plattformunabhängige und elektronisch verarbeitbare Sprache ist, welche wiederum verwendet wird, um Definitionen des Nachrichtengehalts (en: message payload) in verschiedenen geforderten Formaten zu erzeugen. Dadurch wird die vorliegende Norm nicht von Spezifikationen, Entwicklung und/oder Anwendung von Infrastrukturen der nächsten Generation betroffen, weder durch die Anwendung von Normen noch durch urheberrechtliche Mittel.

Für den Zweck des vorliegenden Teils von IEC 61968 bezieht sich das Verteilungs-CIM (DCIM; en: distribution CIM) auf das IEC-CIM-Modell, wie es in IEC 61970-301 und diesem Teil von IEC 61968 beschrieben ist.

Das allgemeine Informationsmodell (CIM) ist ein abstraktes Modell der wichtigsten Objekte in einem Elektrizitätsversorgungsunternehmen, welches üblicherweise an Versorgungsaktivitäten beteiligt ist. Durch die Bereitstellung einer genormten Weise der Darstellung von Ressourcen von Stromversorgungssystemen als Objektklassen und -attribute zusammen mit ihren Beziehungen unterstützt das CIM die Integration von Softwareanwendungen, welche unabhängig von verschiedenen Anbietern entwickelt wurden. Das CIM unterstützt Integration durch die Definition einer allgemeinen Sprache (d. h. Semantik und Syntax) auf der Grundlage des CIM, um es diesen Anlagen oder Systemen zu ermöglichen, auf öffentliche Daten zuzugreifen und Informationen unabhängig davon auszutauschen, wie solche Informationen intern dargestellt werden.

IEC 61970-301 definiert ein Basis-CIM für Anwendungen von Energiemanagementsystemen (EMS), einschließlich vieler Klassen, welche in einem breiteren Spektrum von Anwendungen zweckdienlich wären. Aufgrund ihrer Größe sind die CIM-Klassen in logische Pakete gruppiert, und Sammlungen dieser Pakete

¹⁾ IEC 61968-5, IEC 61968-6, IEC 61968-7, IEC 61968-8 und IEC 61968-14 sind in Beratung.

werden als getrennte Internationale Normen bearbeitet. Das vorliegende Dokument erweitert das Basis-CIM mit Paketen, welche sich auf Netzführungssysteme (DMS; en: distribution management systems) konzentrieren, einschließlich Vermögenswerten, Arbeitsaufwand, Kunden, Laststeuerung, Verbrauchszählung und anderem. IEC 62325-301²⁾ erweitert das CIM durch Pakete, welche sich auf Markttätigkeiten und Marktmanagementanwendungen konzentrieren. Andere CIM-Erweiterungen dürfen als Internationale Normen veröffentlicht werden, wobei jede durch eine getrennte Gruppe von Fachexperten bearbeitet wird. Abhängig von den Bedürfnissen eines Projekts kann die Integration von Anwendungen Klassen und Pakete aus einer oder mehreren dieser CIM-Normen erfordern.

2 Normative Verweisungen

Die folgenden Dokumente, die in diesem Dokument teilweise oder als Ganzes zitiert werden, sind für die Anwendung dieses Dokuments erforderlich. Bei datierten Verweisungen gilt nur die in Bezug genommene Ausgabe. Bei undatierten Verweisungen gilt die letzte Ausgabe des in Bezug genommenen Dokuments (einschließlich aller Änderungen).

IEC 60076-1, *Power transformers – Part-1: General*

IEC 61968-1, *Application integration at electric utilities – System interfaces for distribution management – Part 1: Interface architecture and general requirements*

IEC 61968-2, *Application integration at electric utilities – System interfaces for distribution management – Part 2: Glossary*

IEC 61970-301, *Energy management system application program interface (EMS-API) – Part 301: Common information model (CIM) base*³⁾

IEC 61970-501, *Energy management system application program interface (EMS-API) – Part 501: Common Information Model Resource Description Framework (CIM RDF) schema*

IEC 62361-100, *Naming and Design Rules for CIM Profiles to XML Schema Mapping*⁴⁾

IEEE 802.3, *IEEE Standard for Information technology-Specific requirements – Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications*

3 Begriffe

Für die Anwendung dieses Dokuments gelten die Begriffe nach IEC 61968-2 und die folgenden Begriffe.

ANMERKUNG Für allgemeine Glossardefinitionen siehe Internationales Elektrotechnisches Wörterbuch IEC 60050.

3.1

Energiemanagementsystem

(en: **energy management system, EMS**)

Computersystem, das eine Softwareplattform für die grundlegenden unterstützenden Dienstleistungen und eine Reihe von Anwendungen zur Sicherstellung der Funktionalität der elektrischen Energieerzeugung und der Übertragungseinrichtungen beinhaltet, um eine angemessene Sicherheit der Energieversorgung bei minimalen Kosten zu sichern

²⁾ In Beratung

³⁾ Fünfte Ausgabe. In Beratung.

⁴⁾ In Beratung.

3.2**Verteilnetzmanagementsystem****(en: distribution management system, DMS)**

Computersystem, das eine Softwareplattform für die grundlegenden unterstützenden Dienstleistungen und eine Reihe von Anwendungen zur Sicherstellung der Funktionalität elektrischer Verteilnetze beinhaltet, um eine angemessene Sicherheit der Energieversorgung bei minimalen Kosten zu sichern

3.3**einheitliche Modellierungssprache****(en: unified modeling language, UML)**

formal und umfassend beschreibende Sprache mit Diagrammtechniken zur Darstellung von Software-Systemen von der Anforderungsanalyse über Auslegung und Implementierung bis zur Dokumentation

Anmerkung 1 zum Begriff: UML hat sich aus einer Sammlung von Methoden, die von verschiedenen Fachleuten zusammengetragen wurden, zu einem Internationalen Standard entwickelt. CIM basiert auf UML für die Modelldefinition. Automatisierte Werkzeuge generieren die Dokumentation, Schemata und andere Objekte direkt aus UML. Ein Grundverständnis von UML ist erforderlich, um CIM zu verstehen.

3.4**gemeinsames Informationsmodell mit Erweiterungen für Verteilnetze****(en: common information model with distributions extensions, DCIM)**

die Zusammenfassung des Basis-CIM in IEC 61970-301 und zusätzlich in diesem Dokument IEC 61968-11, definierter Pakete

Anmerkung 1 zum Begriff: Das DCIM ist vorgesehen, um die meisten Modellierungsanforderungen aus dem Anwendungsbereich DMS zu adressieren, jedoch kann ein spezielles Projekt auch andere CIM-Pakete oder Erweiterungen erfordern.

3.5**Profil****(en: profile)**

Teilmenge von DCIM-Klassen, -Beziehungen und -Attributen, die benötigt wird, um einen speziellen Schnittstellentyp zu definieren

Anmerkung 1 zum Begriff: Es kann in XSD-, RDF- und/oder OWL-Dateien beschrieben werden. Ein Profil kann zwischen den Anwendungen geprüft werden. Ein Profil ist notwendig, um DCIM „verwenden“ zu können. Weitere Profile sind in anderen Teilen der Normenreihe IEC 61968 definiert.

3.6**XML-Schema****(en: XML schema)**

Schema zur Definition von Struktur, Inhalt und Bedeutung von XML-Dateien (erweiterbare Auszeichnungssprache; en: extensible markup language)

Anmerkung 1 zum Begriff: XML-Schemata sind in der Regel in Dateien mit einer „xsd“-Erweiterung zu finden. Das DCIM verwendet XSD-Dateien zur Definition von Nachrichten zwischen Anwendungsprogrammen für die meisten Anwendungsbereiche, außer für den Austausch von Modellen in Stromversorgungssystemen.

3.7**Ressourcen-Beschreibungssystem****(en: resource description framework, RDF)**

Webstandard (W3C), das verwendet wird, um Informationsmodelle darzustellen

Anmerkung 1 zum Begriff: RDF ist leistungsfähiger als XSD, da es Datenmodelle beschreiben kann und nicht nur eine XML-Datei. Das DCIM benutzt eine Teilmenge von RDF, um den Austausch von Modellen von Stromversorgungssystemen zu unterstützen.

3.8

Web-Ontologiesprache

(en: web ontology language, OWL)

weiterer Webstandard (W3C), welcher RDF enthält und dieses erweitert

Amerkung 1 zum Begriff: OWL ist leistungstärker als RDF bei der Unterstützung von Datentypen, Aufzählungen; mehr Details zu Klassenbeziehungen und Assoziationen usw. Zukünftige DCIM-Profile können OWL benutzen.

4 CIM-Spezifikation

4.1 CIM-Modellierungsnotation

Das CIM wird durch Verwendung objektorientierter Modellierungstechniken definiert. Insbesondere die CIM-Spezifikation verwendet die Notation mit der einheitlichen Modellierungssprache (UML), welche das CIM als eine Gruppe von Paketen definiert.

Jedes Paket im CIM enthält ein oder mehrere Klassendiagramme, welche alle Klassen in diesem Paket und ihre Beziehungen graphisch darstellen. Jede Klasse ist wiederum im Text durch ihre Attribute und Beziehungen zu anderen Klassen definiert.

Die UML-Notation ist in Dokumenten der „Object Management Group“ (OMG) und mehreren veröffentlichten Fachbüchern beschrieben.

4.2 CIM-Pakete

4.2.1 Allgemeines

Das CIM ist in eine Reihe von Paketen unterteilt. Ein Paket ist ein Mittel aus gruppierten Modellelementen für allgemeine Zwecke. Die Pakete wurde gewählt, um das Modell leichter entwerfen, verstehen und überarbeiten zu können. Das allgemeine Informationsmodell besteht aus mehreren Reihen von Paketen. Entitäten können Assoziationen haben, welche mehrere Klassengrenzen überschreiten. Jede Anwendung wird Informationen verwenden, welche in mehreren Paketen dargestellt sind.

4.2.2 CIM-Pakete

Das umfassende CIM ist aus Gründen der Zweckmäßigkeit in Gruppen von Paketen unterteilt. Die Gruppen, welche DCIM enthalten, sind:

- IEC 61970-301 (Basis-CIM, definiert Datentypen und Ressourcen von Stromversorgungssystemen, wie sie durch typische EMS- und DMS-Steuerungen in Anwendungen in Schaltzentralen gefordert werden;
- dieser Teil von IEC 61968.

Bild 1 zeigt die maßgeblichen⁵⁾ derzeit definierten normativen CIM-Pakete und ihre Abhängigkeitsverhältnisse. Die gestrichelte Linie zeigt ein Abhängigkeitsverhältnis an, wobei der Pfeil vom abhängigen Paket zu dem Paket gerichtet ist, von welchem es abhängt.

⁵⁾ CIM-Erweiterungen für Märkte, IEC 62325-301, sind auch normative CIM-Pakete, werden aber nicht im DCIM verwendet und deshalb nicht angezeigt.

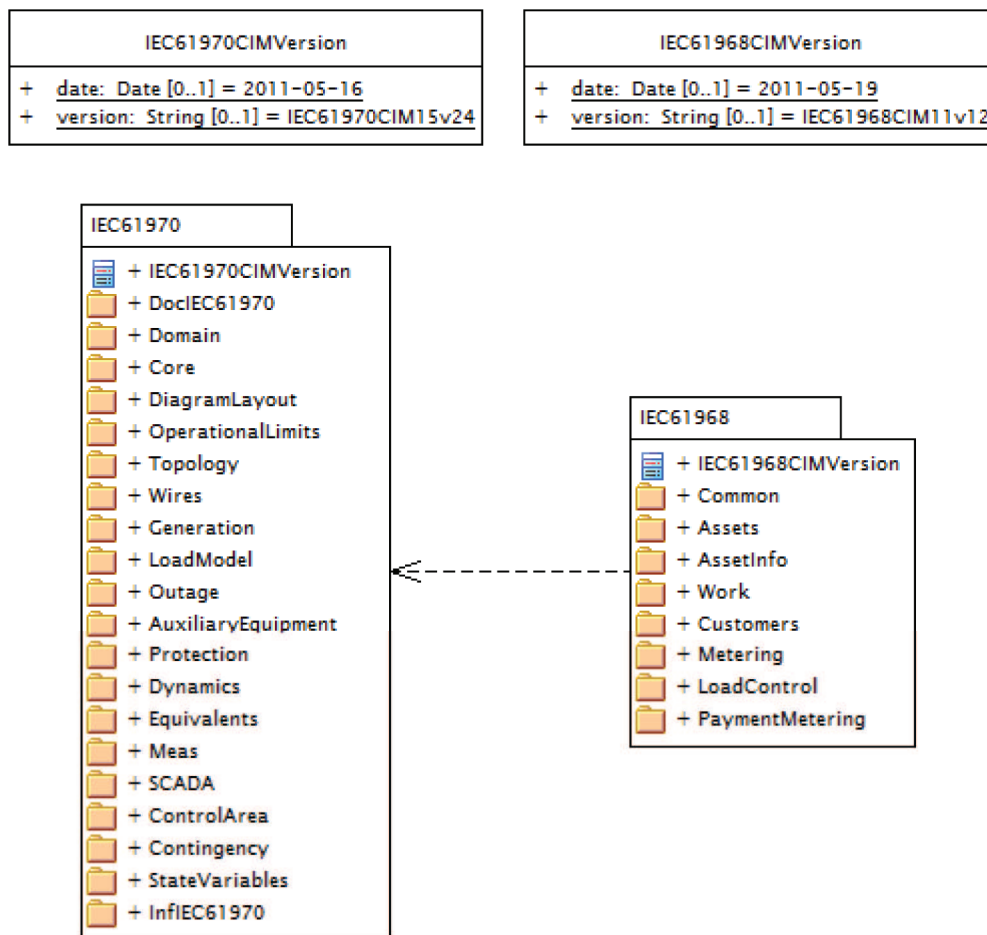


Bild 1 – CIM-Pakete

ANMERKUNG Die Inhalte des Basis-CIM, auf welche aus dem vorliegenden Teil von IEC 61968 verwiesen wird, wurden automatisch aus der Version des elektronischen UML-Modells IEC61970CIM15v31 des Basis-CIM generiert.

4.2.3 CIM-Erweiterungen für Verteilungspakete (dieser Teil von IEC 61968)

Das Basis-CIM, definiert wie in IEC 61970-301, definiert eine Reihe von Teilpaketen, welche sowohl **Domain** (de: Domäne), **Wires** (de: Leitungen), **AuxiliaryEquipment** (de: Hilfseinrichtungen), **Topology** (de: Topologie), **Measurement** (de: Messung), **Equivalents** (de: Äquivalente) und **Core** (de: Kern) einschließt, als auch mehrere andere. IEC 61968-3 bis IEC 61968-9 und IEC 61968-13 erfordern Erweiterungen des CIM, wie sie in IEC 61970-301 spezifiziert sind, um die Objekte und zugehörige Eigenschaften zu beschreiben, welche für Verteilung, Modellierung und Informationsaustausch maßgeblich sind, die nicht nur für typische Leitzentralensysteme anwendbar sind, sondern auch für Unternehmens- und Partnersysteme, und auch für Verbrauchszählungssysteme und -geräte. Folglich dürfen sie, ebenso wie Anwendungen im Verteilungsbereich Klassen aus dem Basis-CIM verwenden, auch Anwendungen außerhalb des Verteilungsbereichs Klassen verwenden, welche im vorliegenden Dokument definiert werden (zum Beispiel Markererweiterungen in IEC 62325-301).

Bild 2 zeigt die Pakete, welche für die CIM-Erweiterungen nach IEC 61968-11 für Verteilung definiert sind. Anmerkungen an der linken Seite des Bildes zeigen den Teil von IEC 61968 an, welcher die Definition von Klassen im betreffenden Paket vorangetrieben hat. Zu beachten ist jedoch, dass verschiedene Dokumente in der Normenreihe IEC 61968 und auch verschiedene Anwendungen, welche CIM zum Informationsaustausch verwenden, üblicherweise Nachrichtengehalte unter Verwendung von Klassen aus mehreren Paketen, einschließlich einigen außerhalb des Dokuments definierten, definieren werden.

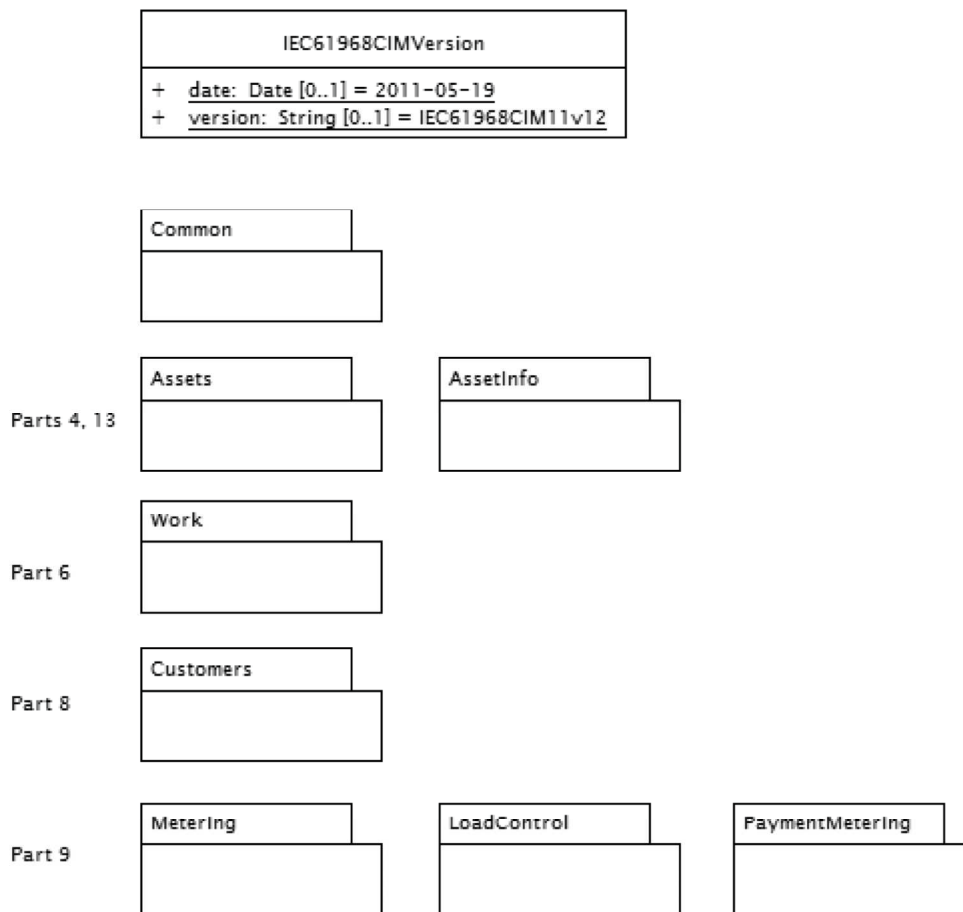


Bild 2 – CIM-Erweiterungen für Verteilungspakete (DCIM) der höchsten Ebene

Abschnitt 6 enthält die Spezifikation für jedes der CIM-Verteilungspakete.

ANMERKUNG Die Inhalte des Basis-CIM, auf welche aus dem vorliegenden Teil von IEC 61968 verwiesen wird, wurden automatisch aus der Version des elektronischen UML-Modells IEC61970CIM11v31 des Basis-CIM generiert.

4.3 UML-Modellierung des CIM

4.3.1 Allgemeines

Das CIM wird durch die Anwendung von UML definiert und gewartet. Die Quelle und Stelle für die Instandhaltung für das CIM ist derzeit eine Enterprise Architect (EA)-Datei. Dies ermöglicht es dem Modell, graphisch betrachtet und gepflegt zu werden. Das gleiche Werkzeug wird verwendet, um Webseiten zu erstellen, welche im Internet angesehen werden können. Aus der EA-Datei wird ebenfalls eine XML-Datei generiert, die in verschiedenen Werkzeugen angewendet werden kann, welche die Erzeugung von kontextspezifischem Nachrichtengehalt im XSD-, RDF- oder OWL-Format für IEC 61968-3 bis IRC 61968-9 und IEC 61968-13 unterstützen.

Der Zweck von 4.3 ist die Definition einiger Prinzipien in Bezug auf das Informationsmodell nach IEC 61968 und den zugehörigen Informationsaustausch. Siehe IEC 61970-301:—, *CIM classes and Relationships*, 4.3 für die Beschreibung verschiedener UML-Konstruktionen, welche im CIM-Modell verwendet werden.

4.3.2 Anwendungsbereich des UML-Modells

Es ist nicht die Absicht des vorliegenden Teils von IEC 61968 und zugehörigen Modellen, Modelle zu definieren, welche alle Anforderungen erfüllen, da dies eine unmöglich zu lösende Aufgabe wäre. Das Standardmodell ist ein anerkanntes Datenmodell für die Integration von Systemen in Unternehmen und muss folglich die Anforderungen für den Informationsaustausch zwischen verschiedenen Systemen erfüllen, z. B.

Nachrichtengehalt, definiert nach IEC 61968-3 bis IEC 61968-9 und IEC 61968-13. Kundenspezifische Erweiterungen sind Gegenstand von nicht normgerechten Projekten und Produkten. Sie können verwendet werden, um CIM für Informationsmodelle von spezifischen Anwendungen oder Systemen oder für nicht normgerechte Integrationsbedürfnisse zu unterstützen. Kundenspezifische Erweiterungen werden jedoch nicht durch die IEC behandelt.

Das gesamte DCIM wurde innerhalb vieler Jahre entwickelt und erst im Jahr 2010 als eine IEC-Norm veröffentlicht. Seit dieser ersten Ausgabe wurde das UML-Modell in normative und informative Klassen und ihre Beziehungen unterteilt. Nur normative Klassen mit ihren Attributen und normativen Beziehungen sind in Abschnitt 6 vollständig dokumentiert. Zur Zeit der Bearbeitung sind die normativen Klassen und Beziehungen jene, welche nach IEC 61968-3 bis IEC 61968-9 und IEC 61968-13 gefordert sind⁶⁾. Sie werden als stabil betrachtet, und nur geringfügige Änderungen werden angenommen.

Im Gegensatz dazu sind informative Klassen und ihre informativen Beziehungen nicht in dieser Spezifikation dokumentiert. Sie sind im elektronischen UML-Modell vorhanden und werden mit den neuen Ausgaben von IEC 61968-3 bis IEC 61968-9 und IEC 61968-13 schrittweise in normative Klassen überführt. Einige dieser Klassen werden solange informativ bleiben, wie angenommen wird, dass sie instabil sind und wahrscheinlich Veränderungen unterliegen, und andere werden möglicherweise entfernt, da sie nicht am normierten Informationsaustausch beteiligt sind.

ANMERKUNG Die nächste erwartete Reihe von Klassen, welche in DCIM12 für die nächste Ausgabe von IEC 61968-11 von informativ nach normativ zu überführen ist, sind jene, welche zur Unterstützung des Überarbeitungszyklus von IEC 61968-3, IEC 61968-4, IEC 61968-9 und IEC 61968-13 und IEC 61968-6 (erste Ausgabe) und IEC 61968-8 (erste Ausgabe) erforderlich sind.

4.3.3 Erweiterbarkeit

Es ist die Absicht, Erweiterungen für das Modell des Informationsaustauschs zuzulassen. Erweiterungen sollten einen lokalen Namensraum verwenden. Der Namensraum muss auch dazu dienen, die Herkunft der Klasse oder Eigenschaft zu identifizieren.

Eine Erweiterung kann in einem UML-Modell und/oder einem physikalischen Modell, wie z.B. XML-Schema, definiert werden:

- Wenn eine Erweiterung in einem UML-Modell erfolgt, sollte eine solche Erweiterung eher in verschiedenen Paketen als eine getrennte Modellschicht oder ein getrennter Kontext gemacht werden, als direkt zum CIM-Modell hinzugefügt zu werden oder dieses direkt abzuändern. Ein Eigenschaftswert „targetNamespace“ (de: Zielnamensraum:) kann zu diesem Zweck verwendet werden, wenn solche Erweiterungen für eine XSD-Generierung vorgesehen sind.
- Wenn eine Erweiterung nur für XSD gemacht wird, muss sich der Ziel-Namensraum vom XSD-Namensraum eines CIM unterscheiden, um die Herkunft der Erweiterung zu identifizieren.

Der Erweiterungs-Namensraum darf einer Benennungskonvention folgen:

`http://<organization>/<versioncontrol>/<profile>`

4.3.4 Definition von Nachrichtengehalt (Profil)

4.3.4.1 Allgemeines

Ein Profil ist beschrieben als Teilmenge von DCIM-Klassen, Assoziationen und Attributen, welche zum Schaffen eines spezifischen Schnittstellentyps erforderlich sind. Ein Profil definiert den Nachrichtengehalt des CIM-Modells (IEC 61968-11 und IEC 61970-301), das Kopfzeilenformat (von Ausführungsprofilen IEC 61968-100 oder IEC 61970-552) und auszutauschende Informationen (spezifisch für Profildokument in IEC 61968-3 bis IEC 61968-9 und IEC 61968-13).

⁶⁾ Das einzige Profil im zweiten Entwurf von IEC 61968-3 wird nach derzeitigem Stand vollständig durch Klassen des Basis-CIM unterstützt.

Eine Möglichkeit, Standard-Nachrichtengehalt (Profile) zu definieren, ist die Verwendung von Anwendungen wie beispielsweise verschiedene Open-Source-Werkzeuge, welche eine Unterstützung für die Erstellung von CIM-basierten Profilen bereitstellen. Andere Methoden und Werkzeuge dürfen verwendet werden, einschließlich manuelle Programmierung.

Derzeit werden in der Normenreihe IEC 61968 zwei Grammatiken von XML für die Definition von DCIM-basierten Profilen verwendet.

4.3.4.2 Syntax des XML-Schemas

IEC 61968-3 bis IEC 61968-9 definieren Profile in XML-Syntax entsprechend IEC 62361-100. Diese Profile spezifizieren das Format und den Inhalt des DCIM-basierten Gehaltteils der Nachricht, welche in verschiedenen Integrationsszenarien ausgetauscht wird, wie in IEC 61968-100 definiert. Nachrichtenkopfzeilen verwenden Kombinationen von Substantiven und Verben. Die Substantive verweisen auf die im DCIM (Gehalt) definierten Konzepte, und die Verben dienen der Unterstützung der Transporttechnik.

Das Verb zeigt üblicherweise an, welche Attribute gefordert sind. Im Fall von **create/created**-Verben (de: erstellen/erstellt:) sind üblicherweise alle in einem Profil definierten Attribute gefordert, während bei den Verben **get** (de: bekommen), **cancel/cancelled** (de: stornieren/storniert), **delete/deleted** (de: löschen/gelöscht) und **close/closed** (de: schließen/geschlossen) üblicherweise nur Objektidentifikatoren gefordert sind. Das Verb **change/changed** (de: ändern/geändert) würde einen Objektidentifikator und die Änderung der Werte von Attributen erfordern.

Die Empfehlungen für die Definition DCIM-basierter Profile für den Datenaustausch sind in IEC 61968-1 gegeben.

Sowohl die für die Verwendung mit IEC 61968 entsprechenden Schnittstellen definierten Verben als auch die Nachrichtenhülle sind in IEC 61968-100 spezifiziert.

Die für die Verwendung mit IEC 61968 entsprechenden Schnittstellen definierten Substantive sind in IEC 61968-3 bis IEC 61968-9 spezifiziert.

4.3.4.3 Syntax des RDF-Schemas

IEC 61970-501 definiert eine Teilmenge des RDF-Schemas, welche für den Massenaustausch in Netzmodellen verwendet wird. Das Profilformat und die Kopfzeileninformation sind in IEC 61970-552⁷⁾ spezifiziert und werden derzeit durch die nach IEC 61968-4 und IEC 61968-13 definierten Profile verwendet.

4.3.4.4 Kontexte von DCIM-gestütztem Datenaustausch

In IEC 61968 definierte DCIM-basierte Profile decken zwei wesentliche Kontexte von Datenaustausch von Unternehmenssystemen ab:

- Massendatenaustausch zum Zweck der Konfiguration von Systemen. Diese Art von Geschäftsprozess ist auch bekannt als Modellverwaltung, Stammdatenverwaltung oder Datenverwaltung. Sie erfordert Nachrichtengehalte, welche die Schaffung und Pflege der Beziehungen zwischen Instanzen verschiedener Geschäftseinheiten zulassen, die zwei oder mehreren Unternehmenssystemen gemein sind.
- Betriebsbedingte (dynamische) Datenaustausche zwischen in Betrieb befindlichen Systemen. Bei Daten, welche über diese Art von Schnittstelle ausgetauscht werden, wird vorausgesetzt, dass eine Konfiguration von Geschäftseinheiten in verschiedenen Systemen ihrer gemeinsamen, DCIM-basierten Darstellung an der Schnittstelle, aber nicht notwendigerweise im System vorausgeht.

Zum derzeitigen Stand werden Profile in RDFS- und OWL-Syntax überwiegend für den Austausch großer Mengen dateibasierter Daten von großen Datensätzen verwendet, während XSD-Profile sowohl in Massen-Nachrichtengehalten als auch in operativen Nachrichtengehalten verwendet werden.

⁷⁾ In Beratung.

4.4 Konzepte und Beispiele von DCIM

4.4.1 Allgemeines

4.4 beschreibt einige Beispiele der Modellierung in der Verteilungsdomäne und in allgemeinen Unternehmen der Elektrizitätsversorgung mit DCIM. Einige Konzepte sind ergänzend zu den Beispielen des Basis-CIM, welche in IEC 61970-301 dargestellt sind (wie beispielsweise für Verteilnetze übliche Netzmodellierung). Andere Konzepte sind übergreifend für Unternehmen der Elektrizitätsversorgung (wie beispielsweise Standorte, Dokumente, Organisationen) oder spezifisch für Verteilungsdomänen (z. B. Verbrauchszählung und Kunden).

4.4.2 Gemeinsame Konzepte

4.4.2.1 DCIM-Schlüsselklassen

Basis-CIM definiert in IEC 61970-301 im Wesentlichen die Funktion von elektrischen Netzwerkelementen durch eine Klasse von **PowerSystemResource** (de: Stromversorgungssystemressource) und durch ihre Spezialisierung für die Bedürfnisse des Informationsaustauschs im Kontext von Anwendungen und Systemen von Leitzentralen. Im vorliegenden Teil von IEC 61968 ergänzt DCIM einige Schlüsselklassen zur Unterstützung von (a) der physikalischen Beschreibung dieser Netzwerkelemente und (b) des gesamten Modells für die Domäne der Verbrauchszählung und auch (c) zum Informationsaustausch in Bezug auf Netzbetrieb und Planung und Verbrauchszählungsmanagement und Steuerung im Kontext des gesamten Versorgungsunternehmens (einschließlich Ausfall-, Arbeits- und Kundenmanagement).

Bild 3 zeigt im DCIM definierte Schlüsselklassen. Beziehungen zwischen ihnen, wenn vorhanden, sind nicht in diesem Bild dargestellt, sondern eher in den folgenden Unterabschnitten. Tatsächlich gibt es zwischen diesen DCIM-Schlüsselklassen zugunsten von spezifischeren Beziehungen zwischen ihren Spezialisierungen relativ wenige direkte Beziehungen. Das UML-Diagramm in Bild 3 zeigt, dass alle Schlüsselklassen identifizierte Objekte sind, d. h. der Name ihrer übergeordneten Klasse, **IdentifiedObject** (de: identifiziertes Objekt), wird in der oberen rechten Ecke einer Klasse angezeigt.

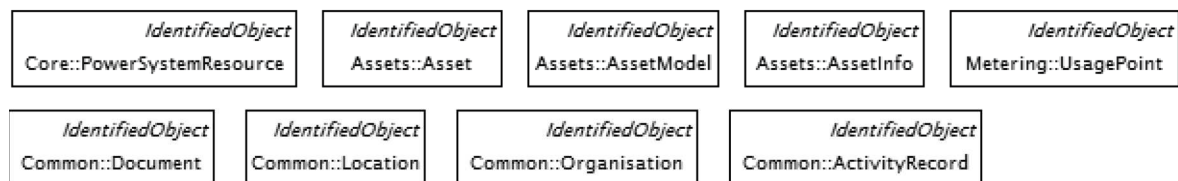


Bild 3 – DCIM-Schlüsselklassen

Asset (de: Vermögenswert) ist eine materielle Ressource der Anlage, einschließlich Stromversorgungseinrichtungen, Endgeräte, Fahrzeuge, Werkzeuge, Schaltschränke, Gebäude, usw. Für elektrische Netzeinrichtungen wird die Funktion des Assets durch die Hierarchie der **PowerSystemResource** modelliert, welche vollständig⁸⁾ im Verdrahtungsmodell (siehe IEC 61970-301) und Modellpaket **IEC61970::Wires** definiert ist. Die Beschreibung von Assets konzentriert sich auf die physikalischen Attribute und die Lebensdauer der Einrichtung, welche diese Funktion ausübt, und basiert auf der Hierarchie von **PowerSystemResource** für elektrische Verbindungsfähigkeit.

AssetModel (de: Assetmodell) liefert Details bezüglich der **Asset** als ein Produkt eines gegebenen Herstellers, so dass ein **AssetModell** (de: Assetmodell) und seine zugehörige **AssetInfo** (de: Assetinformation) viele Instanzen von **Asset** beschreiben können.

⁸⁾ In diesem Teil von IEC 61968 (zweite Ausgabe) wurden alle im Modellpaket **IEC 61968::WiresExt** nach IEC 61968-11:2010 (erste Ausgabe) definierten DCIM-Klassen in das Basis-CIM, IEC 61970-301 (fünfte Ausgabe), und in das Modellpaket **IEC 61970::Wires** verschoben, um die funktionelle Integrität des gesamten CIM-Netzwerks sowohl für Übertragungs- als auch für Verteilnetze sicherzustellen.

AssetInfo ist der Container für Datenblattinformationen zu **Asset** und **AssetModel**, und es können auch mehrere Instanzen von **Asset** und **PowerSystemResource** (d. h. gemeinsam) auf sie verweisen.

Eine **Location** (de: Standort) ist die Lage, der Ort oder die Stelle von etwas, wo sich etwas oder jemand befand, befindet und/oder sich zu einem gegebenen Zeitpunkt befinden wird. Er ist durch eine oder mehrere Positionspunkte (Koordinaten) in einem gegebenen Koordinatensystem definiert.

Ein **UsagePoint** (de: Nutzungspunkt) ist der logische oder physikalische Punkt in einem Netzwerk, welchem Ablesewerte oder Ereignisse zugewiesen werden. Er wird an der Stelle verwendet, an welcher ein physikalischer oder virtueller Zähler angeordnet sein kann. Diese Klasse liefert die Verbindung zwischen dem netzorientierten Modell von elektrisch verbundenen Einrichtungen (Spezialisierungen von **PowerSystemResource**) und dem asset- und gebäudeorientierten Modell der Domäne der Verbrauchszählung.

Ein **Document** (de: Dokument) ist eine Gruppierung von Informationen, welche häufig als Teil eines Geschäftsvorgangs verwaltet wird. Es wird häufig Verweisungen auf andere Objekte enthalten, wie z. B. Assets, Personen und Ressourcen des Stromversorgungssystems. **Organisations** (de: Organisationen) sind Geschäftseinheiten, welche die Rollen von Anlagen, Unternehmern, Lieferanten, Herstellern, Kunden, Steuerbehörden, usw. einnehmen können.

ActivityRecord (de: Aktivitätsprotokoll) zeichnet Aktivitäten einer Einheit zu einem Zeitpunkt auf. Aktivität kann sich auf ein Ereignis beziehen, welches bereits stattgefunden hat, oder auf eine geplante Aktivität.

Die folgenden Unterabschnitte behandeln die Schlüssel-DCIM-Klassen, ihre Spezialisierungen und Beziehungen detaillierter, und verweisen auf ihre vorgesehene Verwendung.

4.4.2.2 Standorte und geographische Darstellungen

Um verschiedene Einheiten und Aktivitäten im Raum lokalisieren zu können, stellt DCIM die Klasse **Location** bereit. Einheiten und Aktivitäten können sowohl durch ihre verschiedenen Arten von Adressen lokalisiert werden, als auch durch Positionskoordinaten, **PositionPoint** (de: Positionspunkt) in einem gegebenen **CoordinateSystem** (de: Koordinatensystem). Änderungen der Konfiguration einer **Location** können mit Instanzen eines speziellen **ActivityRecord**, **ConfigurationEvent** (de: Konfigurationsereignis) nachverfolgt werden.

Die Klasse **Location** ist nicht für die Verwendung für den Austausch graphischer Daten vorgesehen, obwohl ihre **PositionPoints** verwendet werden können, um Anfangslayouts für graphische Anwendungen abzuleiten. Der Austausch von Übersichtsschaltbildern, welche die **PowerSystemResource** darstellen, wird durch das im Basis-CIM nach IEC 61970-301 definierte Modellpaket **IEC61970::DiagramLayout** unterstützt.

ANMERKUNG Die Unterstützung komplexer räumlicher Datenaustausche liegt nicht im Anwendungsbereich des vorliegenden Teils von IEC 61968, da es weithin verwendete und angenommene Standards (z. B. GML) für andere Domänen gibt, welche auch für elektrische Netze angewendet werden dürfen.

Bild 4 zeigt, wie die Klasse **Location** in Bezug auf **Assets**, **PowerSystemResources** und **UsagePoints** verwendet wird.

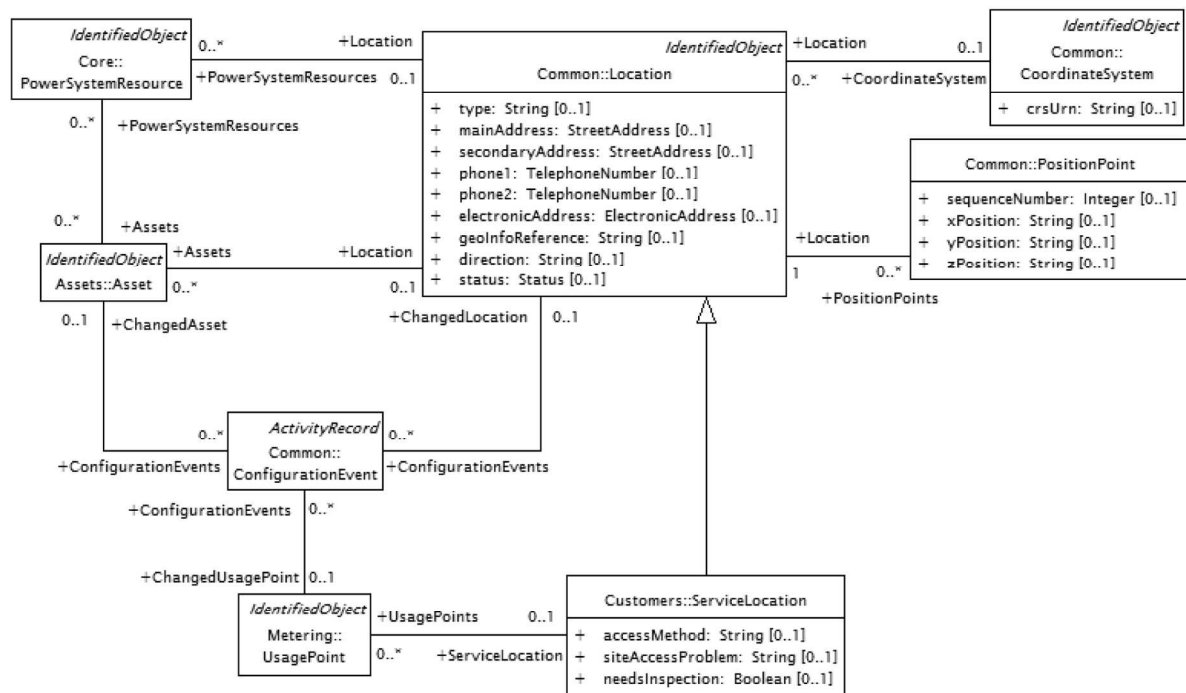


Bild 4 –Stromversorgungssystemressource und Asset-Ort von DCIM

Asset oder **PowerSystemResource** gehören direkt zu einer **Location**, weil die „allgemeine“ **Location** alle erforderlichen Informationen (wie z. B. **mainAddress** (de: Hauptadresse), **electronicAddress** (de: elektronische Adresse oder zugehörige **PositionPoints**) bereitstellt. Im Gegensatz dazu verweist ein **UsagePoint** auf eine spezielle Art eines Ortes, **ServiceLocation** (de: Dienstleistungsort), mit zusätzlichen Attributen, welche nur für Instandhaltung maßgeblich sind, wie z. B. **needsInspection** (de: erfordert Überprüfung). Dies ist ein in DCIM allgemein verwendetes Muster, um dort, wo es zutreffend ist, Beziehungen zwischen Spezialisierungen bereitzustellen und dort auf Klassen höherer Ebene zu verweisen, wo die generische Beziehung auf alle Spezialisierungen anwendbar ist.

4.4.2.3 Organisationen

Bild 5 zeigt zwei Hauptklassen, welche für die Modellierung von Organisationen verwendet werden, **Organisation** und **OrganisationRole**, und ihre vorgesehene Verwendung.

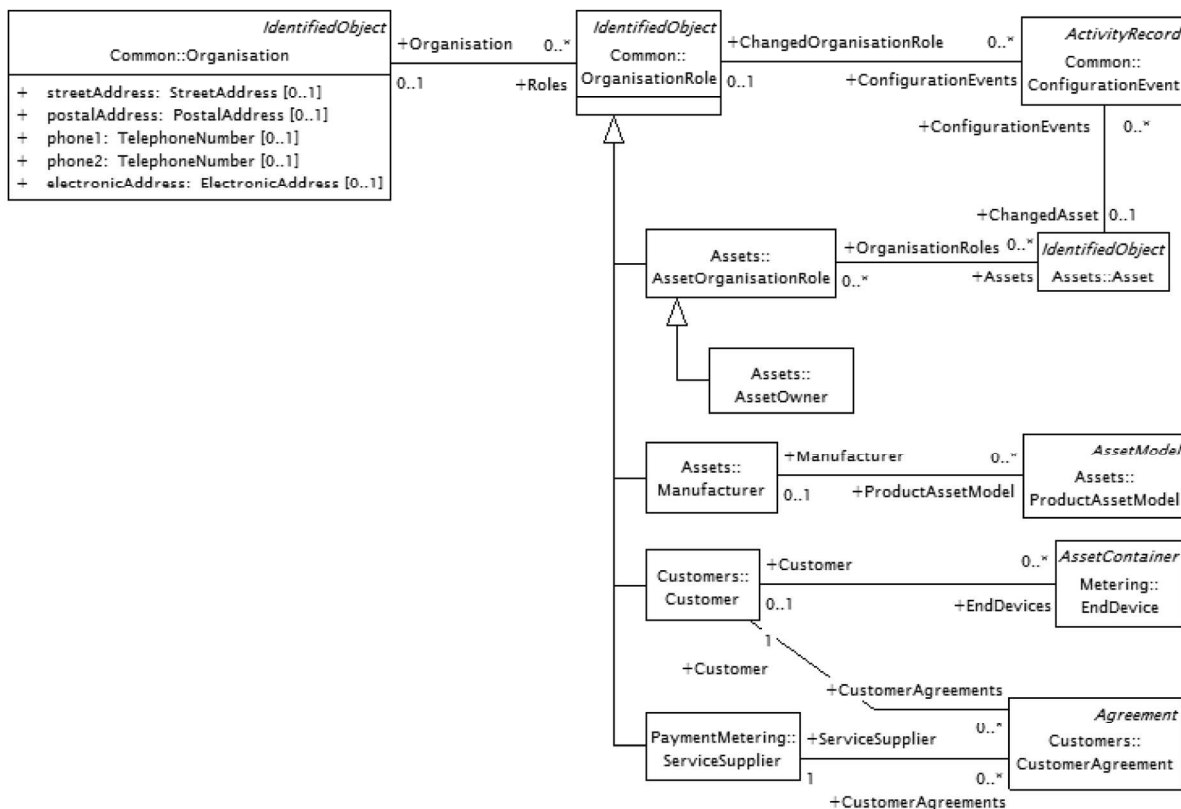


Bild 5 – DCIM-Organisationsmodell

Geschäftseinheiten, welche mit DCIM modelliert werden, werden häufig in vielen verschiedenen Kontexten verwendet. Zum Beispiel kann die gleiche Instanz von **Organisation** die Rolle eines **Customer** in einem Kontext einnehmen, in welchem es um Instandhaltungsanfragen für **EndDevices** (de: Endgeräte) geht, und die Rolle des **Manufacturer** eines **ProductAssetModel** (de: Assetmodell eines Produkts) in einem Kontext, in welchem es um die Erhaltung von Assets geht, oder die Rolle eines **AssetOwner** in einem Kontext, in welchem es um die Verwaltung von Assets geht, und die Rolle eines **ServiceSupplier** in einem Kontext, in welchem es um Instandhaltungsanfragen geht, und entsprechend **CustomerAgreement** (de: Kundenvereinbarung). Dies wird in einem DCIM mit der Beziehung einer **Organisation** zu mehreren **OrganisationRoles** modelliert, wobei jedoch jede **OrganisationRole** auf eine einzige **Organisation** als eine Geschäftseinheit verweisen kann.

Bild 5 zeigt die **OrganisationRole** als den allgemeinen Supertyp verschiedener Rollen. Sie werden nach Bedarf eingeführt, wenn spezifische Attribute (nicht im Bild dargestellt) und/oder Beziehungen, welche nur für eine bestimmte Anwendungsdomäne anwendbar sind, vorliegen. Ein solches Beispiel ist **Manufacturer**: Die Rolle ist für ein gegebenes **ProductAssetModel** anwendbar und nicht für das Asset im Allgemeinen, und folglich gibt es eine explizite Beziehung dieser spezifischen Spezialisierung von **OrganisationRole** zum **ProductAssetModel**, während es andere spezifische Rollen in Bezug auf **Assets** gibt (z. B. **AssetOwner**).

Im DCIM ist **Organisation** als eine Geschäftseinheit nie spezialisiert.

Konfigurationsänderungen von **OrganisationRoles** können auch mit Instanzen von **ConfigurationEvent** nachverfolgt werden.

4.4.2.4 Assets

4.4.2.4.1 Hauptklassen im Assetmodell

Während IEC 61968-11:2010 (erste Ausgabe) lediglich Klassen von **Asset** und **AssetModel** im Zusammenhang mit Verbrauchszählung (IEC 61968-9) enthielt, stellt der vorliegende Teil von IEC 61968 (zweite Ausgabe) ein umfangreicheres Assetmodell bereit, welches die Anforderungen für den

Datenaustausch im Kontext des Austauschs von Netzmodellen (IEC 61968-4 und IEC 61968-13) unterstützt. Bild 6 gibt einen Überblick der Hauptklassen, welche im DCIM-Assetmodell eingeschlossen sind.

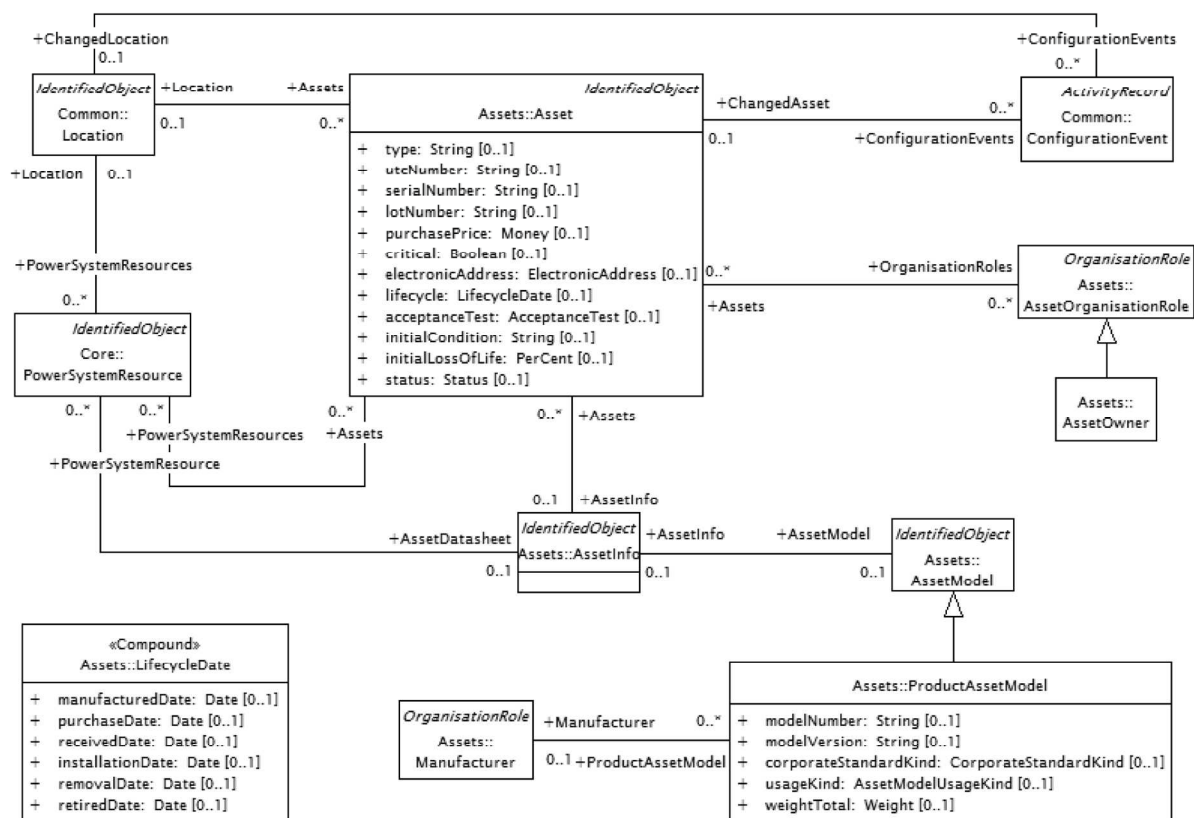


Bild 6 – DCIM-Assetmodell

Asset ist eine materielle Ressource der Anlage, einschließlich der Stromversorgungseinrichtung, von verschiedenen Endgeräten, Schaltschränken, Fahrzeugen, Gebäuden usw. Eine **Asset**-Instanz ist eine verwaltete Ressource, welche in Betrieb oder gelagert sein kann und welche durch ihre **Lebensdauer**attribute gekennzeichnet ist, wobei die möglichen Werte durch die Wortverbindung **LifecycleData** (de: Lebensdauerdaten) definiert sind. Sie kann durch eine **Location** beschrieben werden, und ihre Konfiguration kann durch Instanzen von **ConfigurationEvent** nachverfolgt werden. Ein Asset kann durch mehrere **AssetOrganisationRoles**, wie z. B. **AssetOwner**, auf Organisationen bezogen sein.

ANMERKUNG Im vorliegenden Teil von IEC 61968 (zweite Ausgabe) (dokumentiert DCIM11) gibt es eine einzige Spezialisierung von **AssetOrganisationRole**. Es wird erwartet, dass die nächste Ausgabe (dokumentiert DCIM12) weitere Organisationsrollen mit Bezug zu Assets ergänzen wird, wie z. B. Instandhalter oder Anwender, um den durch IEC 61968-3 und IEC 61968-6 geforderten Datenaustausch zu unterstützen.

Einige Datenaustausche werden ein Modell der Verbindungsfähigkeit elektrischer Einrichtungen erfordern, wobei eine **Asset** auf eine oder mehrere Instanzen von **PowerSystemResource** verweisen kann, welches Modell die elektrische Funktion der Einrichtung durch das physikalische **Asset** dargestellt wird – siehe auch 4.4.2.5. Auch das Gegenteil gilt, d. h. einige Datenaustausche zwischen in Betrieb befindlichen Systemen und anderen Systemen (wie z. B. Planungs- oder Assetverwaltungssysteme) werden im Zusammenhang mit dem elektrischen Modell des Stromversorgungssystems eine physikalische Sichtweise erfordern, und in diesem Fall kann eine **PowerSystemResource** auf die maßgeblichen Instanzen von **Asset** und/oder **AssetInfo** verweisen. Letztendlich werden einige Datenaustausche das Modell der elektrischen Verbindungsfähigkeit vollständig vernachlässigen und ausschließlich mit dem Assetmodell arbeiten.

Eine andere maßgebliche Klasse im Assetmodell ist **AssetInfo**, welche ein identifizierbarer Container für verschiedene physikalische Parameter ist, welche die zugehörigen **Assets** beschreiben. Ein **Asset** ohne die zugehörige konkrete Instanz von **AssetInfo** ist nicht vollständig beschrieben. **AssetInfo**, welche modelliert, was häufig „Katalog“, „Bibliothek“ oder „Code“ genannt wird, ist eine Reihe von Attributen eines Assets,

welche typische Datenblattinformationen eines physikalischen Geräts darstellt, die in verschiedenen Datenaustauschkontexten instanziiert und mitbenutzt werden können:

- als Attribut einer einer Asset-Instanz (installiert oder gelagert);
- als Attribut eines Assetmodells (Produkt eines Herstellers);
- als Attribut eines Assettyps (Gegenstandstyp eines Vermögenswerts, wie er in der Auslegung/Erweiterungsplanung verwendet wird)⁹⁾.

Da Datenblattattribute von physikalischen Geräten vom Typ des Gerätes abhängen, welches sie beschreiben, ist **AssetInfo** die Klasse, die auf konkrete Typen spezialisiert ist, welche die für diese Spezialisierung (siehe 4.4.2.4.2) spezifischen Attribute und Beziehungen aufweisen.

Das Hauptmerkmal einer konkreten Instanz von **AssetInfo** ist der Fakt, dass von mehreren Instanzen von **PowerSystemResource** oder **Asset** darauf verwiesen werden kann. Dies ist eine wesentliche Anforderung bei der Modellierung von Verteilnetzen, wobei die Eigenschaften von Einrichtungen aufgrund der Art und der Menge der verwendeten Einrichtungen selten durch **PowerSystemResource** oder **Asset** definiert werden. Üblicherweise wird durch tausende Instanzen von **PowerSystemResource** oder **Asset** auf eine Instanz einer spezifischen **AssetInfo** verwiesen. Eine ähnliche Anforderung besteht für berechnete elektrische Parameter, wie z. B. Leitungs- oder Transformatorimpedanz, und dies drückt sich in den maßgeblichen elektrischen Katalogklassen aus (4.4.3.3.1).

Die dritte wesentliche auf Assets bezogene Abstraktion ist das Konzept des **AssetModel**. Es repräsentiert das Model eines **Asset**, entweder als Produkt eines spezifischen Herstellers (**ProductAssetModel**) oder als generisches Vermögenswertmodell oder materieller Gegenstand (nicht in Bild 6 dargestellt, weil diese Klasse noch informativ ist). Datenblattkenngrößen des **AssetModel** sind durch die zugehörige Spezialisierung von **AssetInfo** verfügbar und können mit Instanzen von **Asset** oder **PowerSystemResource** gemeinsam verwendet werden.

4.4.2.4.2 Spezialisierungen in Assetmodellen

In den früheren Versionen des elektronischen UML-Modells von DCIM gab es in Ergänzung zur Vererbungshierarchie von **PowerSystemResource** des Basis-CIM, nach IEC 61970-301, üblicherweise drei parallele Vererbungshierarchien in Bezug zu Assets. Der vorliegende Teil von IEC 61968 (zweite Ausgabe) präsentiert ein vereinfachtes Assetmodell, welches gut mit dem Funktionsmodell von **PowerSystemResource** und mit den in 4.4.2.4.1 eingeführten Hauptklassen konsolidiert ist.

4.4.2.4.2 stellt die Richtlinien für die Modellierung bereit, welche in der vorliegenden Ausgabe von IEC 61968-11 eingeführt und angewendet wurden, und diese sollten in zukünftigen Standardversionen von DCIM und auch bei Kundenerweiterungen des Standard-DCIM berücksichtigt werden.

- a) Unter den Klassen **Asset**, **AssetModel**, **ProductAssetModel** und **AssetInfo**: **ProductAssetModel** sollte im Standard-DCIM nie spezialisiert werden, und **AssetModel** wird nur mit **ProductAssetModel**¹⁰⁾ spezialisiert. **AssetInfo** wird immer spezialisiert. **Asset** darf nur entsprechend der Richtlinie d) spezialisiert werden.
- b) Wenn ein Gerät/eine Einrichtung im Modellpaket **IEC61970::Wires** nach IEC 61970-301 existiert (als eine Spezialisierung von oder bezogen auf **PowerSystemResource**), ist es nicht zulässig, sein/ihr Gegenstück als eine Spezialisierung von **Asset** zu definieren. Dagegen muss eine Spezialisierung von **AssetInfo**, wenn zutreffend, mit ihren Datenblatteigenschaften definiert werden.
- c) Eine Beziehung zwischen funktionellen und physikalischen Modellen wird zwischen maßgeblichen Spezialisierungen von **PowerSystemResource** und **AssetInfo** durch eine geerbte Beziehung aufgebaut,

⁹⁾ Unter informativen Klassen im elektronischen UML-Modell von DCIM gibt es eine weitere Spezialisierung von **AssetModel**, welche voraussichtlich in einer der zukünftigen Ausgaben von IEC 61968-11 normativ werden wird.

¹⁰⁾ Unter informativen Klassen im elektronischen UML-Modell von DCIM gibt es eine weitere Spezialisierung von **AssetModel**, welche voraussichtlich in einer der zukünftigen Ausgaben von IEC 61968-11 normativ werden wird.

- entweder **PowerSystemResource-AssetInfo** (wenn die Asset-Instanz nicht existiert) oder **PowerSystemResource-Asset-AssetInfo** (wenn die Asset-Instanz verfügbar ist).¹¹⁾
- d) Wenn eine geräte-/einrichtungs-/netzwerkbezogene Einheit nicht im Modellpaket **IEC61970::Wires** nach IEC 61970-301 modelliert ist, sollte sie von **Asset** oder **AssetInfo** erben, und ihr Name muss nicht das (nun redundante) Wort „Asset“ enthalten.¹²⁾
- e) In Übereinstimmung mit a) sollten die Spezialisierungen von **Asset**, **AssetInfo**, **ProductAssetModel** wie folgt entfernt werden: Die erforderlichen Attributs- und Beziehungsendungen sollten entweder in eine **Asset**-Spezialisierung (wenn sie auf eine Instanz von **Asset** anwendbar sind) oder in eine **AssetInfo**-Spezialisierung (wenn sie Datenblattinformationen repräsentieren, anwendbar auf viele **Asset**-Instanzen) verschoben werden.¹³⁾

Bild 7 zeigt einige DCIM-Beispiele, welche die vorgenannten Richtlinien widerspiegeln, und veranschaulicht außerdem die Anwendung einiger weiterer Klassen des Assetmodells.

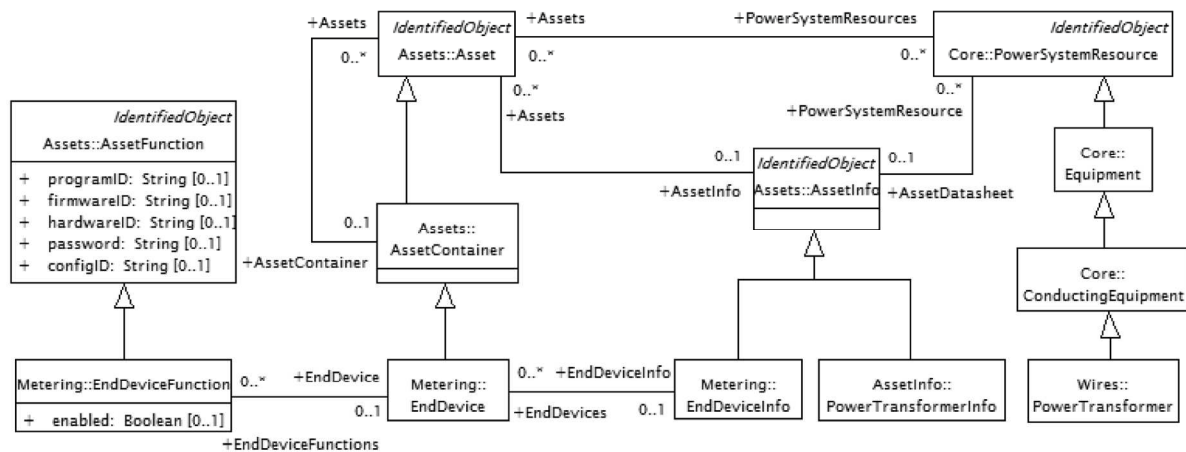


Bild 7 – DCIM-Assets – Spezialisierungsrichtlinien

AssetContainer ist die Spezialisierung von **Asset** und auch der Container für viele **Assets**. Als eine Spezialisierung übernimmt er die Attribute und Beziehungen mit anderen Klassen. Als ein Container läßt er die Modellierung zusammengesetzter **Assets** zu. Ein Beispiel dafür ist das **EndDevice**, welches andere Einheiten enthalten kann, die einen unabhängigen Lebensdauerzyklus haben (wie z. B. Kommunikationsmodule, nicht im Bild dargestellt). Andere Beispiele von **AssetContainer** würden Türme und Tresore (derzeit nicht im normativen DCIM) sein. Keines dieser realen Objekte wird im funktionellen **PowerSystemResource**-Modell (Richtlinie d)) definiert.

Bild 7 zeigt nur zwei von vielen Spezialisierungen von **AssetInfo**: **EndDeviceInfo** (de: Endgerätinformation) und **PowerTransformerInfo** (de: Leistungstransformatorinformation). Letztere übernimmt eine Beziehung zu **PowerTransformers** entweder durch **AssetInfo-PowerSystemResource** oder **AssetInfo-Asset-PowerSystemResource** (Richtlinie c); siehe 4.4.3.4 für Details). Es gibt keine zu **PowerTransformer** (de: Leistungstransformator) zugehörige **Asset**-Spezialisierung (Richtlinie b)).

Für **EndDeviceInfo** gilt, dass sie direkt mehreren Instanzen von **EndDevice** zugeordnet ist und sie auch die Beziehung **AssetInfo-Asset** erbt. In einem solchen Fall sollte die spezifischste Beziehung verwendet werden. Derzeit gibt es wenige Spezialisierungen von Asset, und es ist akzeptabel, solche konkreten Beziehungen

¹¹⁾ Aufgrund der Anwendung dieser Richtlinie wurden einige direkte Beziehungen zwischen Spezialisierungen von **PowerSystemResource** und **AssetInfo** zugunsten der Verwendung der neuen vererbten Beziehung, welche zwischen **PowerSystemResource** und **AssetInfo** definiert ist, entfernt.

¹²⁾ Aufgrund der Anwendung dieser Richtlinie wurden einige Klassen des Modellpakets von **Metering** (de: Verbrauchszählung) umbenannt, um nicht die Vorsilbe „Asset“ zu verwenden.

¹³⁾ Aufgrund der Anwendung dieser Richtlinie wurden einige Klassen des Modellpakets von **Metering** ihre Instanzattribute „verloren“, welche zur zugehörigen **AssetInfo**-Spezialisierung verschoben wurden.

zwischen Spezialisierungen zu definieren, auch wenn die vererbte Beziehung existiert. Die Fälle von Beziehungen zwischen **AssetInfo** und **PowerSystemResource** wurden als zu zahlreich für eine Definition aller expliziten Beziehungen zwischen ihnen angesehen.

Abschließend demonstrieren die Klasse **AssetFunction** (de: Assetfunktion) und ihre Spezialisierung **EndDeviceFunction** (de: Endgerätfunktion) die bevorzugte Auslegung, wenn ausführbar, um nur mit direkten Beziehungen zwischen den maßgeblichen Spezialisierungen, wie z. B. **EndDevice-EndDeviceFunction** in Bild 7, zu beginnen. In zukünftigen DCIM-Ausgaben würden die konkreten Beziehungen möglicherweise für den Fall durch eine generische ersetzt, dass es mehrere Klassen gibt, für welche der Bedarf einer Beziehung zwischen den Spezialisierungen von **AssetFunction** und **Asset** oder **AssetContainer** identifiziert wird.

4.4.2.5 Elektrisches Modell gegenüber physikalischem Modell

Das Verteilungs-CIM deckt sowohl die elektrische als auch die physikalische Darstellung eines Objekts ab. Die Klasse **PowerSystemResource** modelliert die elektrische Darstellung und wird häufig für Netzbetrieb, Überwachung, Ausfallmanagement und Betriebsplanung verwendet, während die Klasse **Asset** die physikalische Darstellung eines Objekts modelliert und im Wesentlichen für die Asset- und Arbeitsverwaltung, aber auch für Netzerweiterungsplanung, Ausfallmanagement und Netzstudien verwendet wird. Zum Beispiel kann die physikalische Darstellung in der Ableitung von Attributen für die elektrische Darstellung bestehen, auch wenn keine explizite Klasse **Asset** verwendet wird. 4.4.3.3.3 erläutert beispielsweise, wie das Assetmodell bei der Berechnung der elektrischen Kenngrößen einer Verteilungsleitung helfen kann. Die Beziehung zwischen den zwei Aspekten der elektrischen Einrichtung liefert auch Schlüsselinformationen für Erweiterungsplanung, Arbeitsverwaltung und Ausfallmanagement.

Die Beziehung zwischen **Asset** und **PowerSystemResource** ermöglicht die Navigation zwischen den zwei verschiedenen Darstellungen (Modellen) eines realen Objekts. Aspekte der Verbindungsfähigkeit und des Betriebs (wie z. B. betriebsbedingte Grenzwerte oder Einschaltzustand) werden immer durch **PowerSystemResource** und ihre Spezialisierungen modelliert, während physikalische Aspekte (wie z. B. Datenblattbemessungsdaten, Abmessungen oder Asset-Lebensdauerdaten und finanztechnische Daten) immer durch **Asset** und ihre zugehörigen Klassen modelliert werden.

Es darf angenommen werden, dass sowohl **Asset** als auch **AssetInfo** Beziehungen zu **PowerSystemResource** haben. Die Beziehung zwischen **PowerSystemResource** und **Asset** wird üblicherweise für den Datenaustausch verwendet, wenn das **Asset** in Betrieb ist, um die physikalische Sichtweise auf betriebene elektrische Netzeinrichtungen zu erhalten oder umgekehrt, um die Sichtweise der elektrischen Verbindungsfähigkeit auf das System zu erhalten, welches die physikalische Darstellung verwendet. Die Beziehung zwischen **PowerSystemResource** und **AssetInfo** kann auch dann verwendet werden, wenn keine tatsächliche **Asset**-Instanz verfügbar oder erforderlich ist; typische Beispiele sind Planungsanwendungen oder Netzberechnungsanwendungen, welche in betriebenen Systemen (wie z. B. Verteilungsstromflüssen) verwendet werden.

4.4.3 Netzmodellierungskonzepte und Beispiele

4.4.3.1 Modellierung von Verbindungsfähigkeit und Phasen

Das Modell der Verbindungsfähigkeit der Verteil-DCIM ist identisch mit dem Modell der Verbindungsfähigkeit der Basis-CIM, d. h. es basiert auf Klassen von **ConductingEquipment** (de: leitende Einrichtungen), **Terminal** (de: Anschluss) und **ConnectivityNode** (de: Verbindungsknoten) (siehe IEC 61970-301:—, 4.4.4, *Connectivity model* und 4.4.8, *Phase wire modelling*). Mit dem Attribut **Terminal.phases** kann die Phaseninformation jedes **ConductingEquipment** für ein Mehrphasen-Verteilnetz ausgedrückt werden. Es ermöglicht auch die Darstellung der Fälle der Nichtübereinstimmung von normalen Netzphasen. In Verteilnetzen gibt es einige seltene Fälle, in welchen ein normaler Öffnungsschalter an einer Seite mit der Phase A verbunden ist und an der anderen Seite mit Phase B. **Jumper** (de: Überbrückungskabel) könnten zu Nichtübereinstimmung von Normalphasen führen, solange eine Seite ausgeschaltet ist, bevor der **Jumper** verbunden wird. Zum Beispiel könnte ein **Jumper** eine stromführende Normalphase A mit einer Normalphase B verbinden, welche zu diesem Zeitpunkt stromlos ist. Dieser Fall ist nur zulässig, wenn alle nachgeschalteten Lasten auf der/denselben Phase/n liegen.

Siehe IEC 61970-301:—, 4.4.4.2, *Connectivity and containment example*, für ein umfangreiches Beispiel eines symmetrischen dreiphasigen Beispielnetzes, und für ein Beispiel zur Veranschaulichung, wie das

Modell der Verbindungsfähigkeit des CIM verwendet werden kann, um ein Mehrphasennetz darzustellen, siehe IEC 61970-301:—, 4.4.8, *Phase wire modelling*.

Bild 8 gibt einen Überblick über die phasenbezogenen Klassen, welche üblicherweise für die Modellierung von Verteilnetzen und für Anwendungen verwendet werden.

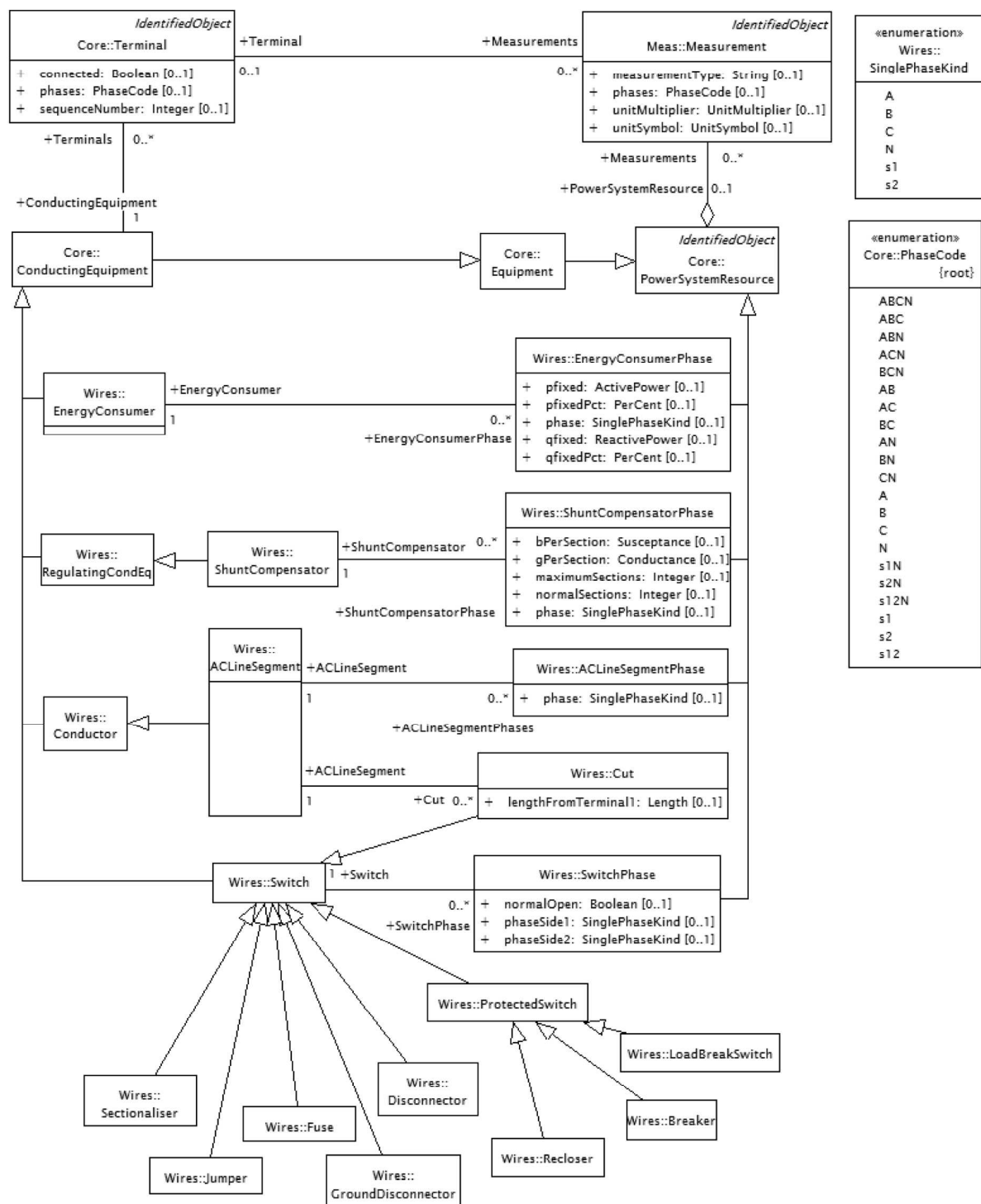


Bild 8 – DCIM-Phasenmodellierung

In Verteilnetzen könnten einige der Geräte unterschiedliche elektrische und betriebsbedingte Kenngrößen auf jeder Phase aufweisen. Zum Beispiel kann ein dreiphasiges **Switch** (de: Schaltgerät) (unabhängig davon, ob es ein **Breaker** (de: Leistungsschalter), eine **Fuse** (de: Sicherung) oder irgendeine andere Spezialisierung von

Switch ist) aus drei getrennten physikalischen Schaltgeräten bestehen, eines für jede Phase. In der Mehrzahl der Fälle haben die drei einzelnen Schaltgeräte die gleichen physikalischen Eigenschaften. Wenn ein solches Schaltgerät gruppenbetätigt wird (d. h. alle physikalischen Schaltgeräte haben den gleichen Normal- und/oder Echtzeit-Betriebszustand), kann es durch eine einzige Instanz von **Switch** modelliert werden. Es kann jedoch vorkommen, dass die einzelnen Phasen unterschiedliche physikalische Eigenschaften aufweisen. Im Fall einer **Fuse** können zum Beispiel, zumindest theoretisch, an jeder Phase verschiedene Sicherungsbemessungswerte verwendet werden. Außerdem kann es bei nicht gruppierten Schaltgeräten Fälle geben, in welchen der Normal- und/oder Echtzeit-Betriebszustand jeder Phase verschieden sein können. Dieser Fall wird dann als eine Instanz von **Switch** modelliert, welche drei Instanzen von **SwitchPhase** (de: Schaltgerätphase), eine für jede Phase, umfasst. Ein wichtiger Aspekt ist, dass es in jedem Fall eine Instanz von **Switch** gibt und dass aus verschiedenen Kontexten (d. h. Konfiguration, Betriebsabläufen) auf eine Instanz verwiesen werden kann.

ANMERKUNG Der vorliegende Teil von IEC 61968 unterstützt nicht mehrphasige Schaltgeräte (modelliert durch **CompositeSwitch** (de: Verbundschalter)).

Aus dem vorliegenden Teil von IEC 61968 hervorgehend, hat die Klasse **Measurement** (de: Messung), vergleichbar der Klasse **Terminal**, die Attribute von **Phase** definiert, um eine detaillierte Phasenbeschreibung für eine Messung bereitzustellen. In den überwiegenden Fällen ist das Attribut wahlfrei, da die Messung an einem **Terminal** mit seiner Phase übereinstimmt. Bei **ConductingEquipment**, welches getrennt auf jeder Phase betrieben werden könnte, stellt jedoch das Attribut **Measurement.phases** die Fähigkeit zur Spezifizierung von Messungen auf jeder Phase bereit. Auf der Grundlage des CIM-Steuerungsmodells, definiert im Basis-CIM, IEC 61970-310:—, 4.4.10, *Measurement and controls*, können Steuerungen auch an getrennten Phasen angewendet werden.

Aus dem vorliegenden Teil von IEC 61968 hervorgehend, stellt das CIM-Verteilungsmodell eine Unterstützung für die Darstellung vorübergehender Änderungen des Netzes bereit. Siehe IEC 61970-301:—, 4.4.9, *Cuts, clamps and jumpers model*, für eine detaillierte Beschreibung und ein Beispiel.

4.4.3.2 Einphasige und unsymmetrische Lasten

Bild 9 zeigt die für die Modellierung von Verteillasten verfügbaren Klassen, welche an einem Ort häufig unter den drei Phasen unsymmetrisch sind. In einigen Fällen können einphasige und zweiphasige Lasten vorkommen.

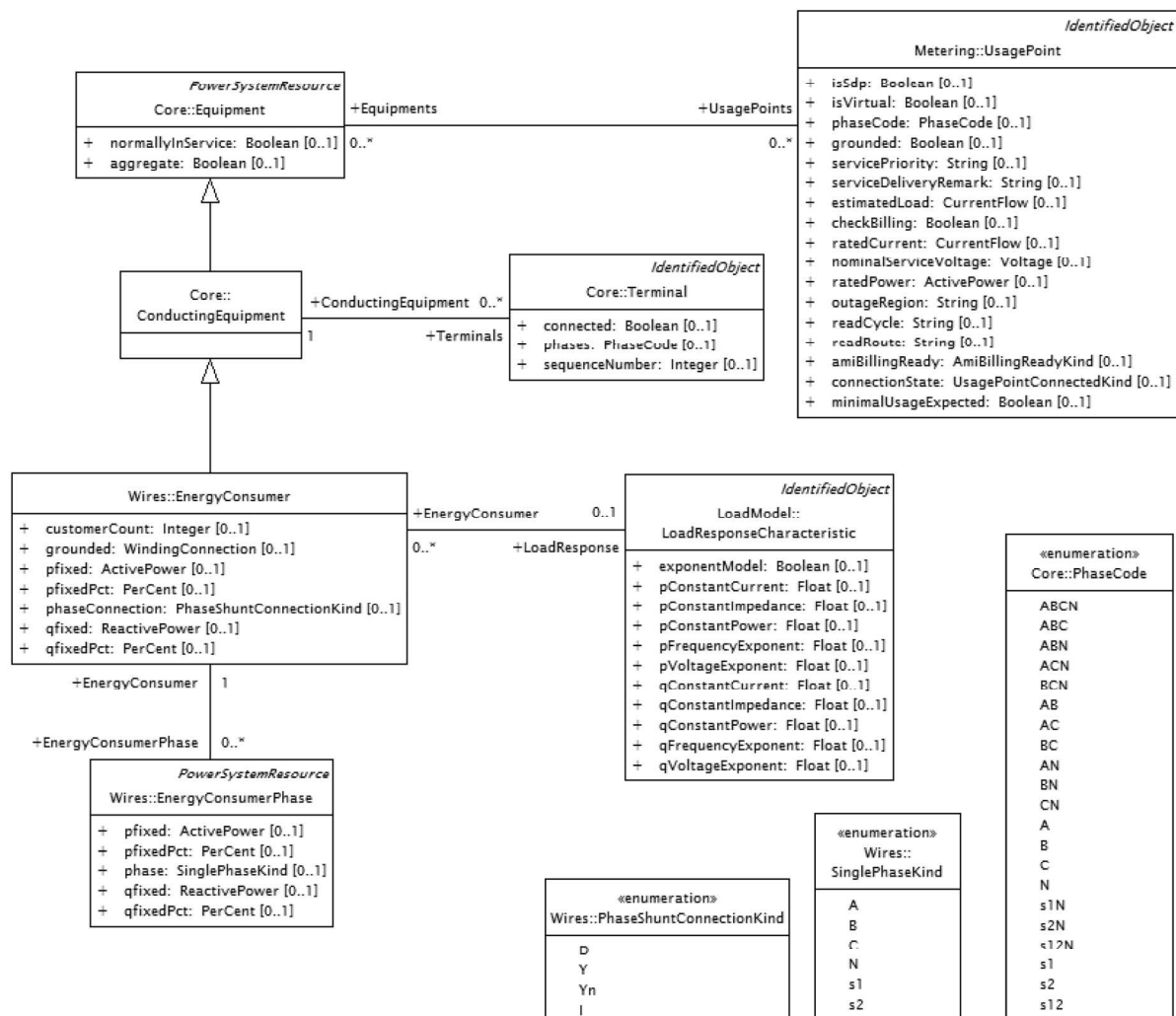


Bild 9 – DCIM-Lastmodell

Die Klasse **EnergyConsumer** (de: Energieverbraucher) sollte verwendet werden, um das Lastmodell zu instanziierten, welches durch einen **UsagePoint** wahlfrei einem Zähler zugeordnet werden kann. **EnergyConsumer** erbt von **ConductingEquipment**, welches mit **Terminal** in Beziehung steht. **Terminal** besitzt Attribute von **Phase**, welche eine **PhaseCode** (de: Phasencode) als buchstabengeheure Aufzählung zugewiesen sein kann. Zum Beispiel beschreibt **AN** eine einphasige Last von A zu null, **BC** beschreibt eine einphasige Last von B nach C, **ABCN** beschreibt eine dreiphasige als Stern geerdete Last, **ABC** beschreibt eine dreiphasige im Dreieck geschaltete Last, usw.

Das Attribut **phaseConnection** von **EnergyConsumer** sollte für eine als Stern geschaltete Last oder Phase-zu-Null-Last **Y** oder **Yn** sein; es sollte für eine als Dreieck geschaltete Last oder Phase-zu-Phase-Last **D** sein. Bei einer symmetrischen dreiphasigen Last ist die gesamte Wirk- und Blindleistung für die gleichmäßige Verteilung unter den drei Phasen an Attributen von **EnergyConsumer** zu spezifizieren.

Wenn eine dreiphasige Last unsymmetrisch ist, kann sie trotzdem mit der gleichen Instanz von **EnergyConsumer** modelliert werden, wobei sie dann in Beziehung zu drei zusätzlichen Instanzen von **EnergyConsumerPhase** steht, wovon jede eine Phase mit ihrem Attribut **phase** repräsentiert. Dies ermöglicht es, einen **EnergyConsumer** mit seiner Identität auch für unsymmetrische Modellierung zu verwenden. Bei einphasigen oder zweiphasigen Lasten sollte das Modell auch genau aufeinander abgestimmte Leiter oder Transformatoren einschließen, welche die Verbindungsfähigkeit zurück zur Quelle herstellen. Die gesamte Wirk- und Blindleistung muss nicht an Attributen von **EnergyConsumer** spezifiziert werden, da sie von der an Attributen von **EnergyConsumer** spezifizierten Verteilung unter den Phasen abgeleitet werden kann. Jede **phase** kann **A**, **B**, **C**, **S1** oder **S2** (**N** gilt in diesem Fall nicht) sein. Für eine als Dreieck geschaltete Last oder eine Phase-zu-Phase geschaltete Last, welche mit **D** für **phaseConnection** gekennzeichnet ist, ist:

- **A** für die zwischen Phasen **A** und **B** geschaltete Last zu verwenden;
- **B** für die zwischen Phasen **B** und **C** geschaltete Last zu verwenden;
- **C** für die zwischen Phasen **C** und **A** geschaltete Last zu verwenden;
- **S1** für die zwischen Phasen **S1** und **S2** geschaltete Last zu verwenden.

Im Fall, dass die Last eine Kombination von konstantem Strom, konstanter Leistung oder konstanter Impedanz ist, sollte eine Instanz von **LoadResponseCharacteristic** (de: Kenngröße des Lastverhaltens) der Instanz **EnergyConsumer** zugeordnet werden.

ANMERKUNG Der vorliegende Teil von IEC 61968 stellt nicht die Mittel für die Definition verschiedener Lastkenngrößen je Phase bereit.

4.4.3.3 Abschnitte der Verteilungsleitung

Bild 10 zeigt die für die Modellierung von Abschnitten von Wechselstromleitungen (d. h. Leiter) verfügbaren Klassen. Es gibt drei Möglichkeiten, um die Impedanzparameter eines **ACLineSegment** (de: Wechselstromleitungsabschnitt) zu beschreiben, wenn nur das Paket **Wire** verwendet wird, und eine vierte Möglichkeit, wenn das Paket **AssetInfo** verwendet wird.

ANMERKUNG Der vorliegende Teil von IEC 61968 (dokumentiert DCIM11) und die zugehörige IEC 61970-301 (dokumentiert Basis-CIM15) spiegeln anerkannte Leitungsabschnitt-Modelle für die Übertragung und Verteilung (T&D; en: transmission and distribution) wider. Folglich wurden die Klassen nach IEC 61968-11:2010 (erste Ausgabe) geringfügig abgeändert und vom Modellpaket **IEC61968::WiresExt** in das Modellpaket **IEC61970::Wires** des Basis-CIM, IEC 61970-301 (fünfte Ausgabe) verschoben.

Um eine einphasige, zweiphasige oder unsymmetrische dreiphasige Leitung mit Verwendung von unterschiedlichen Drähten zu repräsentieren, sollte jeder im Modell festgehaltene Phasen- oder Neutralleiterdraht zu einer Instanz von **ACLineSegmentPhase** (de: Phase des Wechselstromleitungsabschnitts) in Beziehung gesetzt werden. Die zutreffenden Attributwerte von **phase** sind **A**, **B**, **C** für dreiphasige Leitungen, **S1** und **S2** für einphasige Sekundärkreise, und **N** für Fälle, in welchen der Neutralleiter modelliert wird. Jede **ACLineSegmentPhase** hat eine wahlfreie Beziehung zu **WireInfo** (de: Drahtinformation), welche nachstehend detaillierter beschrieben ist, aber alle anderen Kenngrößen kommen vom mit ihr in Beziehung stehenden **ACLineSegment**.

Bei einer unsymmetrischen Leitung spiegeln die berechneten Impedanzen die Positionen der physikalischen Drähte an Turm oder Leitungsmast wider, und diese Positionen sind üblicherweise auf der Grundlage ihrer horizontalen und vertikalen Koordinaten von links nach rechts und von oben nach unten geordnet. Zum Beispiel könnten bei Betrachtung eines Leitungsmastes mit Traverse, welcher 3 Drahtpositionen hat, welche mit 1 (am weitesten links), 2 (außermittig) und 3 (am weitesten rechts) nummeriert sind, die Impedanzen berechnet werden, mit dem Draht von Phase A in Position 1, dem von Phase B in Position 2 und dem von Phase C in Position 3. Ein anderer **ACLineSegment** darf den gleichen Leitungsmasttyp und Drahttyp verwenden, aber mit dem Draht von Phase C in Position 1, dem von Phase B in Position 2 und dem von Phase A in Position 3. Die berechneten Impedanzparameter werden sich von denen im ersten Fall unterscheiden. Sie sind mathematisch durch Zeilen- und Spaltenoperationen der Impedanzmatrix aufeinander bezogen, aber solche Operationen werden im CIM weder unterstützt, noch gibt es ein Konzept für die Anordnung von Phasen im CIM. Dies bedeutet, dass jede andere Anordnung einer physikalischen Phase eine andere Impedanzbeschreibung im CIM erfordert. Diese Betrachtung wirkt sich auf die Verwendung von **PerLengthPhaseImpedance** (de: Phasenimpedanz je Länge) und **WireSpacingInfo** (de: Drahtabstandsinformation) in Bild 10 aus.

Vertauschte Leitungen können mit einem anderen **ACLineSegment** für jeden Abschnitt modelliert werden und eine andere Impedanzbeschreibung für jeden Abschnitt, wie vorgenannt diskutiert. Die Impedanzbeschreibung sollte entweder **PerLengthPhaseImpedance** oder **WireSpacingInfo** verwenden, weil Sequenzparameter dauerhafte Vertauschung annehmen. Es ist nicht notwendig, Instanzen von **ACLineSegmentPhase** für vertauschte dreiphasige Leitungsabschnitte zu verwenden, wenn alle Phasendrahttypen innerhalb eines Abschnitts identisch sind.

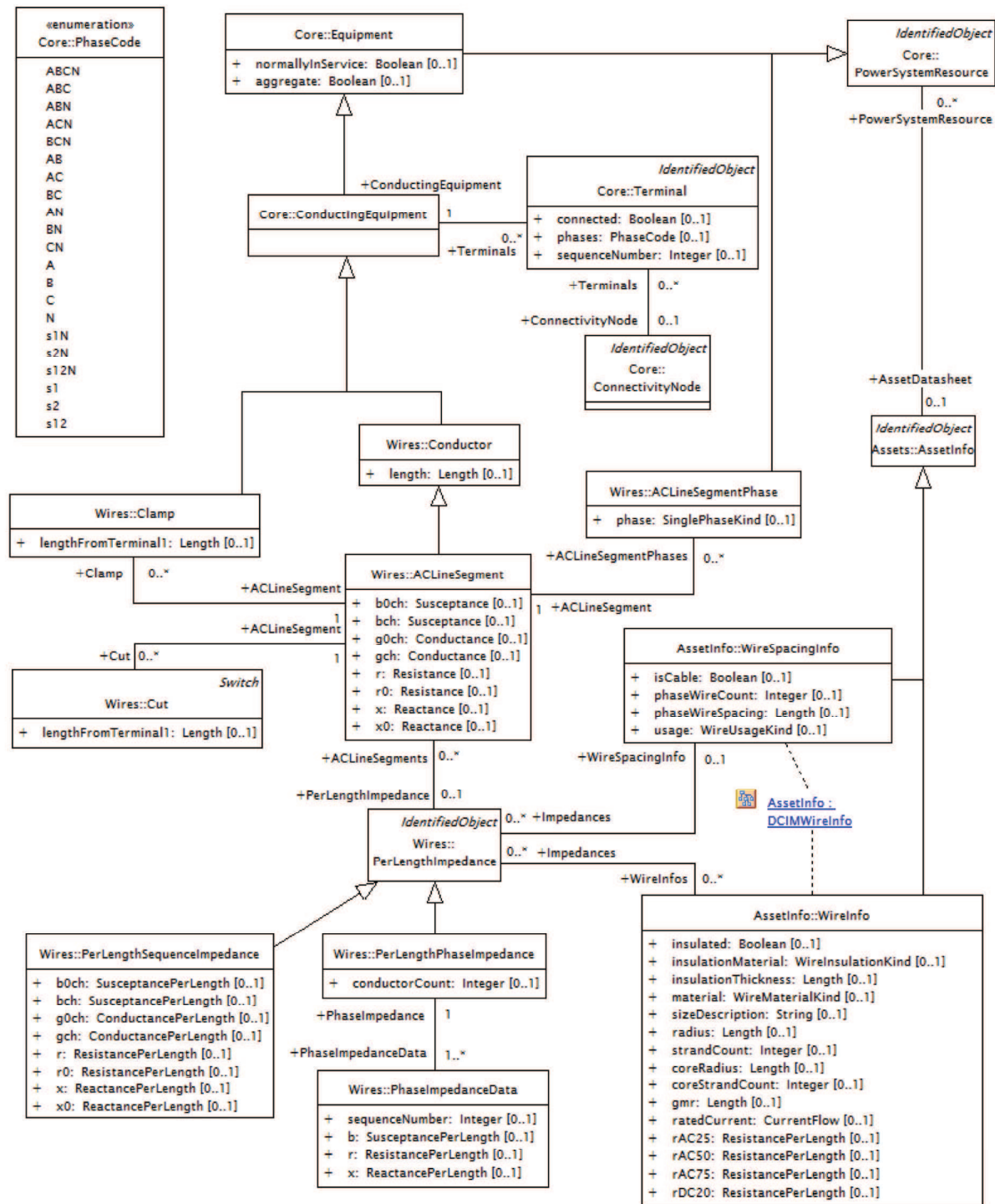


Bild 10 – DCIM- Modell der Leitungsverbindungsfähigkeit

Es gibt drei Möglichkeiten, die Leitungsimpedanz nur unter Verwendung des Modellpakets **IEC61970::Wires** nach IEC 61970-301 zu spezifizieren:

- Die Attribute **r**, **r0**, **x**, **x0**, **bch**, **bch**, **gch** und **gch0** von **ACLineSegment** dürfen für ein symmetrisches Modell verwendet werden. Siehe 4.4.3.3.1 für die Verwendung von einphasigen und zweiphasigen Leitungsabschnitten.
- Eine Beziehung zu **PerLengthSequenceImpedance** (de: Impedanz je Länge) ist bereitzustellen. Diese Klasse modelliert einen elektrischen Katalog oder eine Bibliothek von „Leitungscode“, welche Impedanzen und Leitungsberechnung je Längeneinheit aufweisen. Viele **ACLineSegments** werden sich eine Instanz von **PerLengthSequenceImpedance** teilen und werden Impedanz mit ihrem von **Conductor**

übernommenen Längenattribut berechnen. Siehe 4.4.3.3.1 für die Verwendung von Sequenzparametern mit einphasigen und zweiphasigen Leitungsabschnitten.

- Eine Beziehung zu **PerLengthSequenceImpedance** ist bereitzustellen, welche vorherberechnete symmetrische $N \times N$ -Matrizen von Impedanz und Admittanz je Längeneinheit in Bezug setzt. Diese Klasse modelliert ebenfalls, und das vererbte Attribut von **length** (de: Länge) ist gefordert. Die **conductorCount** (de: Leiteranzahl) muss mindestens gleich der Anzahl der Phasen sein, aber sie könnte größer sein, wenn der Neutralleiter (oder andere Erdungsleiter) in der Matrix gehalten wird. **PhaseImpedanceData** (de: Phasenimpedanzdaten) führt Matrixelemente X und Y ein, die in Spalten abgelegt werden. Die Attribute **r**, **x** und **sequenceNumber** (de: Sequenzanzahl) sind alle gefordert, während **b** wahlfrei ist. Nur die unteren Dreieckselemente werden abgelegt, so dass eine 3×3 -Matrix 6 Elemente (d. h. Instanzen von **PhaseImpedanceData**) hätte. Die Zeilen und Spalten der Matrix müssen der Phasenordnung entsprechen. Ein in Beziehung gesetztes **ACLineSegment** wird Zeile 1 der ersten Phase zuweisen, welche von der geordneten Liste (**A**, **B**, **C**, **S1**, **S2**, **N**) vorhanden ist, Zeile 2 der nächsten vorhandenen Phase und so weiter. Siehe auch nachstehend 4.4.3.3.2.

Zur Spezifizierung von Impedanzen unter Verwendung des Pakets **AssetInfo** ist das Attribut **Conductor.length** (de: Leiterlänge) erforderlich, weil die elektrischen Parameter der Instanz als (Längenparameter je Einheit) $\times$ (**Conductor.length**) definiert werden. Siehe auch nachstehend 4.4.3.3.3.

Eine **WireSpacingInfo** kann unter Verwendung von zwei Methoden zu einer **ACLineSegment** in Beziehung gesetzt werden:

- (**PowerSystemResource-AssetInfo**) Die vererbte Beziehungsendung von **AssetDatasheet** des **ACLineSegment** ist zu verwenden, um auf **WireSpacingInfo** zu verweisen. Diese Methode darf auch verwendet werden, um **WireInfo** ihren Attributen von **ratedCurrent** (de: Bemessungsstrom) zuzuordnen.
- (**PowerSystemResource-Asset-AssetInfo**) Die Klasse **Asset** ist zu instanziiieren, in welcher die Beziehungsendung von **AssetInfo** auf die **WireSpacingInfo** verweist und mehrere **PowerSystemResources** unter Verwendung dieser **WireSpacingInfo** auf jedes **ACLineSegment** verweisen.

Für Impedanzberechnungen ist es außerdem erforderlich, zu jedem/jeder verweisenden **ACLineSegment** oder **ACLineSegmentPhase** **WireInfos** in Beziehung zu setzen. Dies kann entweder durch **AssetDatasheet** (**PowerSystemResource-AssetInfo**) oder durch Navigationsmuster der **Asset**-Klasse (**PowerSystemResource-Asset-AssetInfo**) erreicht werden. Wann immer der **ACLineSegment** in Beziehung gesetzte **ACLineSegmentPhases** hat, sollte jede **ACLineSegmentPhase** ihre eigene **WireInfo** haben. Anderenfalls:

- muss mindestens eine **WireInfo** mit dem **ACLineSegment** in Beziehung gesetzt werden, und es wird angenommen, dass sie für alle Phasendrähte gilt. Sie gilt außerdem für alle Neutralleiter, ausgenommen, eine zweite, identifizierbare **WireInfo** kann zugeordnet werden.
- Eine wahlfreie zweite **WireInfo** darf zum Neutralleiter in Beziehung gesetzt werden, welche als die mit dem geringsten Attributwert von **WireInfo.radius** identifiziert wurde. Wann immer diese Annahme nicht auf die Leitung zutrifft, ist eine Modellierung nach Phase mit **ACLineSegment** anzuwenden.

Bild 10 zeigt Beziehungen zwischen **PerLengthImpedance**, **WireSpacingInfo** und **WireInfo**. Diese dienen jedoch der Verwaltung der Assetbibliothek und nicht der Impedanzberechnung. Entweder ist das **AssetDatasheet** oder das Klassenmuster von **Asset** erforderlich, um **WireSpacingInfo** oder **WireInfo** für Impedanzberechnungen zu verwenden.

Bild 11 zeigt die Klassenhierarchie der **WireInfo**, welche für die Bestimmung von Impedanzen von Leitungsabschnitten aus physikalischen Daten (manchmal als „Leitungscode“ und „Kabelcode“ bezeichnet) verwendet wird. **WireInfo** ist eine Basisklasse, welche nicht direkt instanziiert werden sollte. Ihre Attribute beschreiben die physikalischen Daten für eine Freileitung (es ist zu beachten, dass abgeleitete **OverheadWireInfo** (de: Freileitungsdrahtinformation) derzeit keine eigenen Attribute haben) oder einen Kern-Phasenleiter eines Kabels.

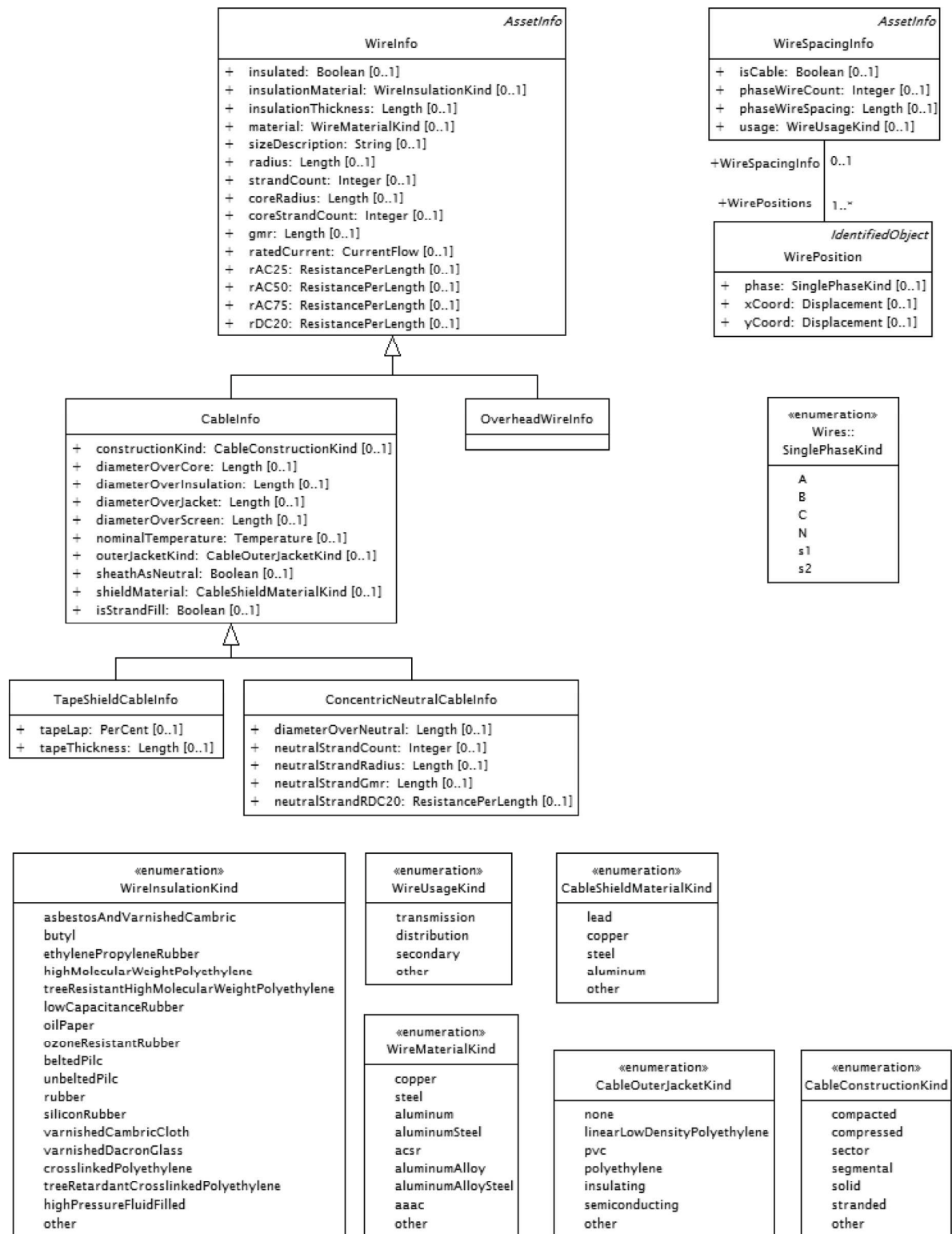


Bild 11 – Modell des DCIM-Leiters (Leitungs- und Kabeldatenblatt)

Die Attribute **material** (de: Werkstoff), **sizeDescription** (de: Größenbeschreibung), **strandCount** (de: Aderanzahl) und **coreStrandCount** (de: Kernaderanzahl) (für stahlverstärkte Aluminiumleiter-Kabel (ACSR; en: aluminium conductor steel reinforced) von **WireInfo** unterstützen die Identifizierung des Drahtes und sind häufig im lokalen Namen der Instanz kodiert. Die mindestens geforderten elektrischen Attribute sind **gmr** (geometrisches Mittel des Radius; en: geometric mean radius), **radius** (de: Radius) und **rAC50** (Widerstand bei Wechselstrom und 50 °C). Eine vollständige Tabelle von Drähten würde üblicherweise Widerstände enthalten, welche bei anderen Temperaturen und Frequenzen bestimmt sind, wie z. B. **rAC25**, **rAC75** und

rDC20. Bei ACSR-Kabeln ist der **coreRadius** (de: Kernradius) wahlfrei für die frequenzabhängige Berechnung des **gmr**. Der **coreRadius** ist für andere als ACSR-Kabel null. Der **ratedCurrent** ist die Strombelastbarkeit bei 50 °C. Der **ratedCurrent** ist erforderlich, um den Ausgangs-Lastfluss auszuwerten, aber nicht für die Lastflusslösung selbst. Die Attribute **insulated** (de: isoliert), **InsulationMaterial** (de: Isolationswerkstoff) und **insulationThickness** (de: Isolierungsdicke) gelten im Wesentlichen für die abgeleiteten Kabelklassen, aber sie unterstützen auch dreifache Sekundärleitungen und Abstandshalter von Freileitungskabeln. Das Attribut **CableInfo** beschreibt zusätzliche Eigenschaften von Schichten über dem leitenden Kern. Der **diameterOverCore** (de: Durchmesser über dem Kern) umfasst sowohl den Leiter als auch den halbleitenden Schirm; er sollte der Innendurchmesser der Isolierung sein. Der **diameterOverInsulation** (de: Durchmesser über der Isolierung) ist der Außendurchmesser der Isolierschicht, wobei keine äußeren Schirme oder halbleitenden Schichten eingeschlossen sind; er sollte der Innendurchmesser des Schirms oder Mantels sein. Der **diameterOverJacket** (de: Durchmesser über dem Mantel) ist der Außendurchmesser des Kabels; er sollte abgesehen von konzentrischen Neutralleitern der größte festgelegte Durchmesserwert sein.

Die Attribute **tapeLap** (de: Bandüberlappung) und **tapeThickness** (de: Banddicke) von **TapeShieldCableInfo** (de: Information zur Bandschirmung des Kabels) beschreiben ein dünnes, üblicherweise aus Kupfer bestehendes Band, welches das Kabel in überlappenden Wicklungen einhüllt. Die **ConcentricNeutralCableInfo** (de: Information zum konzentrischen Neutralleiterkabel) hat Attribute für die äußeren Neutralleiter, welche üblicherweise aus mehreren massiven Kupferdrähten bestehen. Die Anzahl der Drähte ist **neutralStrandCount** (de: Anzahl der Neutralleiterader), der Radius ist **neutralStrandRadius** (de: Radius der Neutralleiterader), der geometrische Mittelwert des Radius ist **neutralStrandGmr** (de: geometrischer Mittelwert des Radius der Neutralleiterader), und der Widerstand je Draht ist **neutralStrandRDC20** (de: Widerstand der Neutralleiterader bei Gleichstrom und 20 °C). Der **diameterOverNeutral** (de: Durchmesser über dem Neutralleiter) ist der Durchmesser über den konzentrischen Neutralleiteradern, welcher gleich dem **diameterOverJacket** sein kann.

ANMERKUNG Das in Bild 11 dargestellte Modell unterstützt derzeit die zwei Typen von Einzeladernkabeln, welche in Verteilungssystemen am häufigsten zu finden sind. Viele andere Kabeltypen, welche in Übertragungs- oder Industriesystemen zu finden sind, werden nicht unterstützt; dies schließt 3-Leiter-Kabel, Rohrkabel, Unterwasserkabel und andere ein.

WireSpacingInfo definiert die geometrischen Daten der Leitung. Die Attribute **phaseWireCount** (de: Anzahl der Phasendrähte) und **phaseWireSpacing** (de: Abstand der Phasendrähte) verweisen auf Teilleiter-bündelung, welche für Verteilungen nicht üblich ist, aber im Modell bei Hochspannungs-Übertragungs-leitungen vorkommen kann.

WirePosition (de: Drahtposition) beschreibt die horizontalen (**xCoord**) (de: x-Koordinate) und vertikalen (**yCoord**) (de: y-Koordinate) Koordinaten jedes Drahtes am Abschnitt des Leitungsmastes oder Turms oder im Kabelgraben/-kanal, und **phase** identifiziert, welcher Phasendraht dort montiert ist. Alle drei Attribute sind erforderlich. Die Höhe der Freileitung über dem Boden ist **yCoord** (de: y-Koordinate), einschließlich jeglicher Druchhangauswirkungen. Für Erdkabel ist die **yCoord** die mittlere Verschüttungstiefe, welche als negative Zahl eingegeben wird. Die horizontale Position des Drahtes, **xCoord**, wird an einer beliebigen, aber durchgehenden Bezugslinie gemessen. Übliche Wahlmöglichkeiten für den horizontalen Bezug sind die Mittellinie des Leitungsmastes und die am weitesten links befindliche Drahtposition.

Jegliche Drähte an einer **WirePosition** mit **phase=N** werden als dauerhaft geerdet angenommen. Die Anwendung darf diese Leiter durch die Reduzierung nach Kronecker aus der Matrix von Impedanz und Admittanz entfernen.

Die Attributwerte von **WirePosition.phase** können abhängig von der Feldinstallation häufig wechseln. Zum Beispiel kann eine einphasige Leitung an Phase **A**, **B** oder **C** verwendet werden. Dies erfordert drei Instanzen von **WireSpacingInfo**, wobei der einzige Unterschied drei verschiedene Werte für einen der in Beziehung stehenden Werte von **WirePosition.phase** sind, und zwar **A**, **B**, **C**. Der Wert von **WirePosition.phase** würde sich für einen Neutralleiterdraht nicht ändern, er sollte immer **N** sein. Zusammengefasst erfordert ein einphasiger Leitungstyp 3 Instanzen von **WireSpacingInfo**, um alle Phasenmöglichkeiten abzudecken. Ein zweiphasiger Leitungstyp erfordert 6 Instanzen und ein dreiphasiger Leitungstyp erfordert ebenfalls 6 Instanzen.

4.4.3.3.1 Verwendung von Sequenzimpedanzen (symmetrischer Fall)

Die Mitimpedanz und die Nullimpedanz dürfen durch die Attribute **r**, **x**, **r0** und **x0** von **PerLengthSequencelImpedance**, in Beziehung gesetzt zu einer Instanz von **ACLineSegment**, transferiert werden. Die Attribute **bch**, **b0ch**, **gch** und **g0ch** sind üblicherweise für Freileitungs-Verteilungen nicht von Bedeutung. Bei drei Phasen beschreibt dies eine symmetrische dreiphasige oder perfekt umgesetzte Leitung. Die Attribute in **PerLengthSequencelImpedance** werden in Einheiten je Länge angegeben, folglich ist es erforderlich, ihre Werte mit den **length**-Attributen von **Conductor** zu multiplizieren.

ANMERKUNG Dies ist gleich den Attributen von **Wires::ACLineSegment**, welche für die gesamte Länge des Abschnitts vorherberechnet und an jeder Instanz des Abschnitts definiert sind. Im Gegensatz kann **PerLengthSequencelImpedance** in Beziehung gesetzt werden und kann als solche (durch Zuordnung) durch mehrere Abschnittsinstanzen verwendet werden, wodurch die Menge von in Datenaustauschen übertragenen Daten verringert wird.

Wenn das **ACLineSegment** nur eine oder zwei Phasen hat, kann ein symmetrisches Modell trotzdem durch die Attribute **r**, **x**, **r0** und **x0** transferiert werden. Dies stellt eine Impedanzmatrix mit entsprechend komplexen Diagonalelementen, Z_s , dar und entsprechend komplexen Nicht-Diagonalelementen, Z_m . Für eine einphasige Leitung sind die zu transferierenden Attribute:

$$Z_1 = Z_0 = Z_s$$

Für eine zweiphasige oder dreiphasige Leitung sind die zu transferierenden Attribute:

$$Z_1 = Z_s - Z_m$$

$$Z_0 = Z_s + (n - 1) Z_m$$

Dabei ist n die Anzahl der Phasen. Nach Erhalt von **r**, **x**, **r0** und **x0** wird die symmetrische zweiphasige oder dreiphasige Impedanzmatrix konstruiert aus:

$$Z_s = (Z_0 + (n - 1) Z_1) / n$$

$$Z_m = (Z_0 - Z_1) / n$$

Das verweisende **ACLineSegment** sollte in Beziehung gesetzte Instanzen von **ACLineSegmentPhase** haben, welchen passende **phase**-Werte zugewiesen sind, um die tatsächlich vorhandenen Phasen zu zeigen, wie z. B. **A**, **B**, **C**, **S1** oder **S2**. Der Neutralleiter, **N**, sollte nicht vorkommen, weil jeder Neutralleiter in die Erdung integriert worden sein muss, wenn Impedanzen verwendet werden.

Für Verteilungs-Erdkabel sind Impedanzen ebenfalls geeignet, einschließlich der Attribute **bch** und **b0ch**.

4.4.3.3.2 Verwendung von Phasenimpedanzen (unsymmetrischer Fall)

Berechnete Matrixparameter können anstelle des in 4.4.3.3.3 beschriebenen physikalischen Modells transferiert werden, indem eine Instanz von **PerLengthPhaseImpedance** in Beziehung gesetzt wird. Ein Matrixmodell ist zweckmäßig, wenn:

- die Zielanwendung keine Mittel zur Berechnung von Parametern aus den physikalischen Daten besitzt;
- die zugrundeliegenden physikalischen Daten nicht ohne Weiteres verfügbar sind;
- es erforderlich ist, die unsymmetrischen Leitungsparameter so genau wie möglich aufeinander abzustimmen.

Für eine zweiphasige Leitung, bei welcher der Neutralleiter bereits verringert wurde, werden die Z- und Y-Matrizen 2x2 sein, aber aufgrund der Symmetrie wird es nur 3 spezifische Matrixelemente geben. Dies führt zu drei in Beziehung gesetzten Instanzen von **PhaseImpedanceData** mit spaltenweiser Ablage:

- **sequenceNumber** = 1 für Zeile 1, Spalte 1 der Matrix;

- **sequenceNumber** = 2 für Zeile 2, Spalte 1 der Matrix;
- **sequenceNumber** = 3 für Zeile 2, Spalte 2 der Matrix.

Die Instanz von **PerLengthPhaseImpedanz** kann von einem **ACLineSegment** in Beziehung gesetzt werden, welches Phasen **AB**, **AC** oder **BC** aufweist. Zeile 1 entspricht immer der ersten vorhandenen Phase und Zeile 2 entspricht immer der zweiten vorhandenen Phase, der Reihenfolge **A**, **B**, **C**, **s1**, **s2**, **N** folgend. Wenn die Phasendrähte in unterschiedlichen Positionen installiert sind, derart, dass C (zum Beispiel) in Zeile 1 sein sollte, ist eine neue Instanz von **PerLengthPhaseImpedanz** erforderlich.

4.4.3.3.3 Verwendung physikalischer Parameter

Bei Freileitungen kann ein physikalisches Modell durch eine Beziehung zu einer Instanz von **WireSpacingInfo** transferiert werden, welche mit weiteren in Bild 11 dargestellten Klassen in Beziehung steht. Dies wird die Berechnung einer Impedanzmatrix einer unsymmetrischen Phase unterstützen, indem die Carson-Gleichung, oder eine vergleichbare Methode zum Umgang mit Erdung, angewendet wird. Als Beispiel werden drei vorhandene Phasendrähte und ein Neutralleiter mit unterschiedlicher Größe an einem Leitungsmast mit horizontaler Traverse angenommen. Dies erfordert eine Instanz von **WireSpacingInfo**, vier Instanzen von **WirePosition**, welche die vier Leiterpositionen beschreiben, und zwei Instanzen von **WireInfo**, welche die Drahttypen von Phase und Neutralleiter beschreiben. Das Attribut **length** von **Conductor** muss verwendet werden, und viele **ACLineSegments** werden üblicherweise auf dieselbe Instanz von **WireSpacingInfo** verweisen.

Die Instanzen von **WirePosition** haben ein **phase**-Attribut, welches auf **ACLineSegmentPhase.phase** abbildet. Das Widerstandsattribut von **WireInfo** sollte für die Netzgrundfrequenz und für die erwünschte Betriebstemperatur des Drahts für Berechnungen bereitgestellt werden.

Ein einphasiger, konzentrischer Neutralleiter erfordert eine Instanz von **ConcentricNeutralCableInfo**, eine Instanz von **WireSpacingInfo** mit Attribut **isCable** = true (de: ist Kabel = wahr) und eine Instanz von **WirePosition**, um die Verschüttungstiefe zu spezifizieren. Ein dreiphasiges bandgeschirmtes Kabel mit blankem Neutralleiter würde eine Instanz von **TapeShieldCableInfo**, eine Instanz von **WireSpacingInfo** mit Attribut **isCable**=true und vier **WirePositions** für die drei Phasen und den Neutralleiter erfordern. Es wird auch eine **WireInfo** für den Neutralleiter geben.

4.4.3.4 Verteiltransformatoren

4.4.3.4.1 Elektrisches Modell

Bild 12 zeigt die Klassen, welche die Instanzen des Leistungstransformators modellieren. Sie können ausschließlich das Paket **Wires** verwenden, um die Impedanzparameter zu definieren, oder verwenden die in Bild 13 im Detail dargestellten Klassen von **AssetInfo**, um eine Bibliothek von Transformatorentypen zu definieren.

ANMERKUNG Der vorliegende Teil von IEC 61968 (dokumentiert DCIM11) und die zugehörige IEC 61970-301 (dokumentiert Basis-CIM15) spiegeln anerkannte Transformatoren-Modelle für die Übertragung und Verteilung (T&D) wider. Folglich wurden die Klassen nach IEC 61968-11:2010 (erste Ausgabe) geringfügig abgeändert und vom Modellpaket **IEC61968::WiresExt** in das Modellpaket **IEC61970::Wires** des Basis-CIM, IEC 61970-301 (fünfte Ausgabe), verschoben.

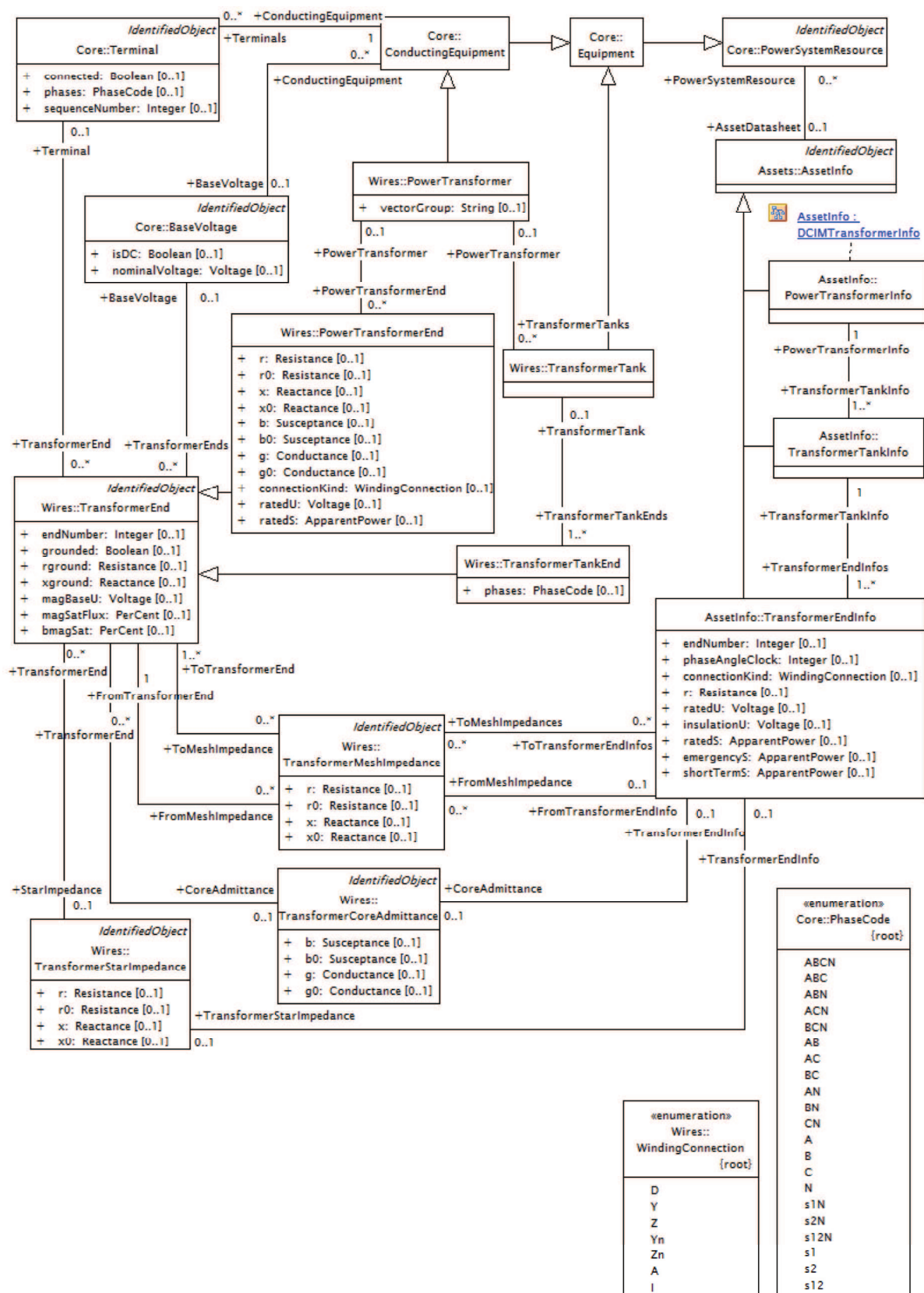


Bild 12 – DCIM-Modell der Verbindungsfähigkeit von Transformatoren

PowerTransformer ist die oberste Instanz für einen Leistungstransformator, ob aus einem dreiphasigen Kessel oder möglicherweise aus unterschiedlichen einphasigen Kesseln aufgebaut. Sie stammt von

ConductingEquipment und hat folglich in Beziehung gesetzte **Terminals** mit **phase**-Attributwerten an jedem Wicklungsverbindungs- und Wicklungsabzweigspunkt. Im CIM wird eine Transformatorwicklung als „end“ (de: Ende) bezeichnet.

Bei einem Aufbau aus verschiedenen Kesseln wurde ein **PowerTransformer** häufig als „Transformatorbank“ bezeichnet, und das CIM unterstützt die Modellierung mit oder ohne Kessel. In Verteilsystemen stellen unabhängige Phasen-Spannungsregler und offene Stern-/offene Dreiecks-Transformatoren zwei Beispiele bereit, welche Modellierung auf Kesselebene erfordern. Auf der Übertragungsebene dürfen Höchstspannungs-Transformatorbanken (EHV) auch einphasige Transformatoren enthalten, welche nicht identisch sein müssen, insbesondere, wenn ein Ersatzteil in Betrieb ist. Das Attribut **vectorGroup** (de: Schaltgruppe) für Schutztechnik ist von den internen Wicklungsverbindungen und Phasenwinkeln abgeleitet; es verwendet die Nomenklatur nach IEC 60076-1, um jede Anzahl von Wicklungen zu beschreiben, welche in der Bank enthalten sein kann.

Wenn **TransformerTank** (de: Transformatorenkessel) verwendet wird, muss es zu seinem **PowerTransformer** in Beziehung gesetzt werden. Es erbt von **Equipment**, nicht von **ConductingEquipment**. Der Kessel darf für die Modellierung von Asset-Datenblättern, detaillierter beschrieben durch Bild 13 und in 4.4.3.4.2, in Beziehung gesetzte **TransformerTankInfo** (de: Transformatorkesselinformation) aus dem Paket **AssetInfo** haben. Da die Prüfung von Transformatoren an Kesseln durchgeführt wird, sind die Datenblätter grundsätzlich zu den Kesseln in Beziehung gesetzt. Werden keine Kessel im Modell verwendet, kann der **PowerTransformer** trotzdem eine Beziehung zu **PowerTransformerInfo** für die Modellierung von Asset-Datenblättern haben. In beiden Fällen sind die Daten tatsächlich in Instanzen von **TransformerEndInfo** (de: Transformatorendinformation) enthalten, in Beziehung gesetzt zu einer **TransformerTankInfo**.

Es gibt zwei Methoden für die Referenzierung von Asset-Datenblättern von Transformatoren, und ein Profil kann die eine oder die andere bevorzugen:

- (**PowerSystemResource-AssetInfo**) Es ist entweder die Beziehungsendung von **AssetDatasheet** von **PowerTransformer** oder von **TransformerTank** zu verwenden, um entweder auf **PowerTransformerInfo** beziehungsweise auf **TransformerTankInfo** zu verweisen, oder
- (**PowerSystemResource-Asset-AssetInfo**) die Klasse **Asset** ist zu instanziiieren, in welcher die Beziehungsendung von **AssetInfo** entweder auf **PowerTransformerInfo** oder auf **TransformerTankInfo** verweist und mehrere **PowerSystemResources** unter Verwendung dieses Datenblatts auf jeden **PowerTransformer** oder **TransformerTank** verweisen.

TransformerEnd (de: Transformatorende), welches in früheren Versionen des CIM auch als **TransformerWinding** (de: Transformatorwicklung) bezeichnet wurde und ein **ConductingEquipment** war, ist kein **ConductingEquipment** mehr, sondern hat ein in Beziehung gesetztes **Terminal** mit Phaseninformationen. Die Attribute **magBaseU**, **magBaseFlux**, und **bmagSat** repräsentieren die Kernsättigung, welche üblicherweise an nicht mehr als zwei der Enden modelliert wird. Die anderen Instanzattribute definieren die Erdungsmöglichkeiten:

- direkte Erdung: **grounded** = true (de: geerdet = wahr), **rground** = 0, **xground** = 0;
- Impedanzerdung: **grounded** = true, **rground** ≥ 0, **xground** ≥ 0;
- keine Erdung: **grounded** = false (de: geerdet = falsch).

TransformerEnd sollte nicht direkt instanziiert werden; es sollte einer ihrer zwei Abkömmlinge instanziiert werden:

- **PowerTransformerEnd**, wenn keine Modellierung auf Kesselebene verwendet wird. Die Attributwerte **ratedU** (de: Bemessungsspannung) und **ratedS** (de: Bemessungsleistung) für Wicklungsbemessungswerte sind festzulegen, und **connectionKind** (de: Verbindungsart) für die Sternschaltung, Dreieckschaltung oder anderen Typ der Verbindung.
- **TransformerTankEnd** (de: Transformatorkesselboden), wenn Modellierung auf Kesselebene verwendet wird. Die Wicklungsschaltung und Bemessungsdaten stammen von einer Instanz von **TransformerTankEndInfo** (de: Information zu Transformatorkesselboden), was bedeutet, dass das Paket **AssetInfo** für die Modellierung auf Kesselebene erforderlich ist. Das **phases**-Attribut unterstützt die Kesselmodellierung je Phase.

Mit **PowerTransformer** und **PowerTransformerTank** gibt es zwei Möglichkeiten für die Spezifizierung von Impedanzparametern, wobei nur das Modellpaket **IEC61970::Wires** des Basis-CIM, IEC 61970-301, verwendet wird:

- Die Attribute **r**, **x**, **r0**, **b**, **b0**, **g**, und **g0** von **PowerTransformerEnd** sind zu verwenden, um Impedanzparameter zu spezifizieren. Dies ist die Möglichkeit, welche mit früheren Fassungen von IEC 61970-301 am besten vereinbar ist, und es gibt einige wichtige in jener Norm beschriebene Konventionen.
- Es ist eine in Beziehung gesetzte **TransformerStarImpedance** (de: Sternimpedanz des Transformators) für jedes **PowerTransformerEnd** zu verwenden, einschließlich einem Stern-Äquivalent (manchmal auch T- oder wye-Äquivalent genannt). Dies kann für bis zu drei Enden (Wicklungen) mathematisch genau sein. Im Fall von drei Enden können jedoch negative Attributwerte auftreten. Wahlweise ist eine zu einem der **PowerTransformerEnds** in Beziehung gesetzte **TransformerCoreAdmittance** (de: Admittanz des Transformator kernels) zu verwenden, wodurch der vorhandene Strom und die Kernverluste repräsentiert werden. Dies kann die Wicklung mit der geringsten Spannung (d. h. am nächsten am Kern) sein oder die Wicklung, welche tatsächlich einer Prüfung ohne Last unterzogen wurde, wenn diese bekannt ist. Die Bezugsspannung für alle Attributwerte, welche Einheiten von Ohm oder Siemens haben, muss für das Ende, mit welchem die Impedanz oder Admittanz verbunden ist, **ratedU** sein.
- Es ist eine **TransformerMeshImpedance** (de: Maschenimpedanz des Transformators) zu verwenden, welche zu jeder Kombination von Paaren von **PowerTransformerEnd** in Beziehung gesetzt ist. Es müssen sein: $(\text{numberEnds}-1) \times \text{numberEnds} / 2$ von diesen; zum Beispiel eine **TransformerMeshImpedance** zwischen zwei Enden, drei von diesen zwischen drei Enden, sechs von diesen zwischen vier Enden, usw. Die Vorteile der Maschenmodells sind: (a) Es ist für mehr als drei Enden mathematisch genau, (b) es hat keine negativen Attributwerte und (c) es entspricht direkter den Daten der Kurzschlussprüfung des Transformators. Die Bezugsspannung muss **ratedU** von **FromTransformerEnd** (de: vom Transformator-ende) sein (es ist zu beachten, dass das andere Ende fast immer eine andere Bemessungsspannung hat). Wahlfrei ist eine wie für das Sternäquivalent beschriebene **TransformerCoreAdmittance** zu verwenden.

4.4.3.4.2 Datenblattmodell

Bild 13 zeigt die Klassen, welche den Austausch von Transformator-Datenblattmodellen zulassen, in Anwendungen manchmal auch „Transformatorencodes“ genannt.

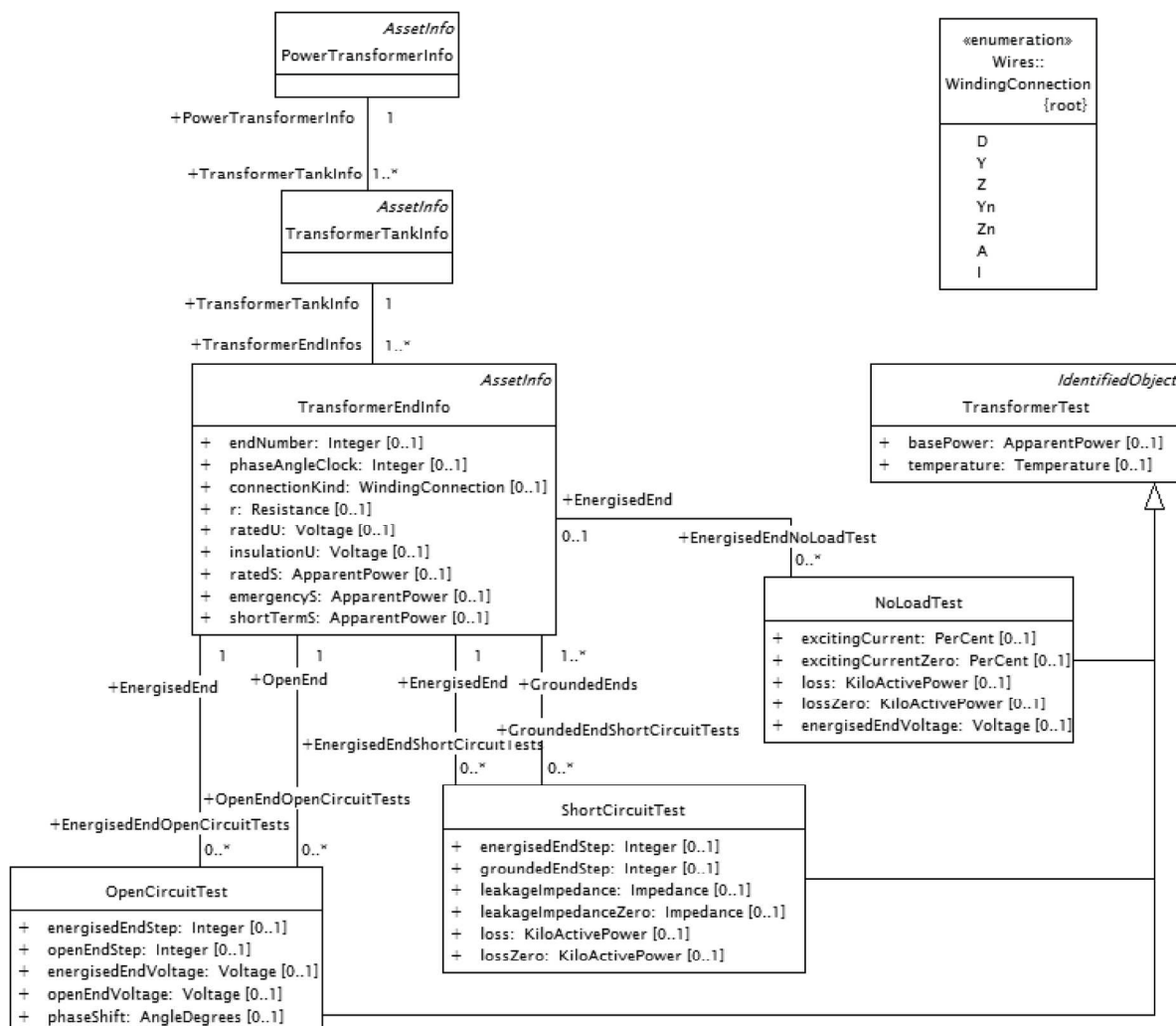


Bild 13 – DCIM-Transformator-Datenblattmodell

Die Hauptklasse ist **TransformerEndInfo**, welche Bemessungs- und Verbindungsdaten für die entsprechenden Transformatorwicklungen enthält. Bemessungsdaten umfassen **ratedU**, **ratedS**, **shortTermS** (de: Kurzzeitleistung) **emergencyS** (de: Notfalleistung) und **insulationU** (de: Isolationsspannung). Das Attribut **r** ist der Gleichspannungswiderstand der Wicklung.

Verbindungsdaten sind in den Attributen **connectionKind**, **phaseAngleClock** (de: Frequenz des Phasenwinkels) und **endNumber** enthalten. Das Attribut **endNumber** ist die Anordnung der Wicklung in der **vectorGroup** des referenzierenden **PowerTransformer** und wird für die Abbildung der Daten auf **TransformerEnd.endNumber** (de: Nummer des Transformerendes) verwendet. Die zugeordnete **Terminal.sequenceNumber** könnte ebenfalls die gleiche **TransformerEnd.endNumber** verwenden, aber dies ist nicht gefordert. Einer Konvention folgend beginnt die **endNumber** üblicherweise mit 1 für die Wicklung mit der höchsten Spannung, und alle anderen Wicklungen werden nach der Reihenfolge der abnehmenden Bemessungsspannung nummeriert. Einige Transformatoren haben mehr als eine Wicklung mit der gleichen **ratedU**, wobei geteilte Sekundärtransformatoren nur ein Beispiel sind, und sie müssen unterschiedliche **endNumbers** haben.

Die Buchstabenaufzählung von **WindingConnection** (de: Wicklungsschaltung), welche als Typ für **connectionKind** verwendet wird, schließt die Standardnomenklatur **D**, **Y**, **Z**, **Yn** und **Zn** zur Beschreibung von Dreieck-, Stern-, Zickzack- und Normalschaltungen in dreiphasigen Transformator-Schaltgruppen ein. **A** wird für die Wicklung eines allgemeinen Spartransformators und **I** für Wicklung eines einphasigen Transformators verwendet.

Das Attribut **endNumber** ermöglicht die Verwendung und Aktualisierung der Datenblatt-Bibliothek mit nur einer Beziehung zu **PowerTransformerInfo** oder **TransformerTankInfo**. Die **TransformerTankInfo** und **PowerTransformerInfo** stellen zwei Navigationspfade nach **TransformerEndInfo** bereit. Mit **PowerTransformerInfo** kann es erforderlich sein, für die Modellübertragung eine künstliche **TransformerTankInfo** zu erstellen, weil sich Datenblätter auf der Kesselebene befinden. Eine Instanz von **PowerTransformerInfo** verweist tatsächlich auf eine Sammlung von einem oder mehreren Kesseln.

Auf **TransformerTankInfo** wird durch **TransformerTank** (vererbt von **PowerSystemResource-AssetInfo**) und **PowerTransformerInfo** (direkt) verwiesen. Sie dient im Wesentlichen der Organisation der Daten in einer Bibliothek mit einem Eingangspunkt. **TransformerTankInfo** sammelt Beziehungen zu **TransformerEndInfo**, wodurch jede Anzahl von Wicklungen verallgemeinert wird.

Es gibt zwei Möglichkeiten der Spezifizierung von Impedanzparametern in einem Datenblattmodell:

- Zuordnen von **TransformerEndInfo** zu den Klassen **TransformerMeshImpedance**, **TransformerStarImpedance** und **TransformerCoreAdmittance** des Modellpakets **IEC 61970::Wires** nach IEC 61970-301. Diese Beziehungen wurden in Bild 12 dargestellt und in 4.4.3.4.1 beschrieben.
- Verwenden von **TransformerTest** (de: Transformatorprüfung) und ihren drei abstammenden Klassen, dargestellt in Bild 13. Jede Anwendung ist für die Umwandlung von Prüfdaten in Maschenequivalent, Sternequivalent oder ein weiteres anderes elektrisches Impedanzmodell zuständig.

TransformerTest ist die Überklasse für alle Transformator-Prüfungsklassen, wobei die Attribute **basePower** (de: Grundspannung) und **temperature** (de: Temperatur) für alle Prüfungen anwendbar sind. **basePower** ist für die Konvertierung von Datenblattwerten in **Impedanz** oder **Admittanz** bei der richtigen Bezugsspannung unerlässlich.

Die Attribute **excitingCurrent** (de: Erregungsstrom), **excitingCurrentZero** (de: Erregungsstrom null), **loss** (de: Verlust) und **lossZero** (de: Verlust null) von **NoLoadTests** (de: Prüfung ohne Last) werden alle am **EnergisedEnd** (de: unter Spannung stehendes Ende) der Wicklung gemessen und stellen die Basisdaten für einen Admittanzzweig bereit. Prüfdaten von Positiv- und Nullsequenzen können in derselben Instanz von **NoLoadTest** wiedergegeben werden. Diese Prüfung wird im Allgemeinen durch Anlegen der Bemessungsspannung an die unter Spannung stehende Wicklung durchgeführt, aber in **energisedEndVoltage** (de: Spannung am unter Spannung stehenden Ende) können abweichende Spannungen festgelegt sein, und mehrere Prüfungen können bereitgestellt werden, um die Parameter der Kernsättigung zu bestimmen.

Eine **ShortCircuitTest** (de: Kurzschlussprüfung) wird durchgeführt, indem der Bemessungsstrom durch die Wicklung des **EnergisedEnd** geleitet wird, wobei eine oder mehrere Wicklungen von **GroundedEnds** (de: geerdete Enden) kurzgeschlossen sind. Sie liefert die Basisdaten für den Stromkreis des Maschen- oder Sternequivalents. Wenn Prüfungen bei unterschiedlichen Stufenschaltereinstellungen durchgeführt werden, werden die Abzweigwerte in **energisedEndStep** (de: Takt am unter Spannung stehenden Ende) und **groundedEndStep** (de: Takt am geerdeten Ende) spezifiziert. Prüfdaten von Positiv- und Nullsequenzen können in derselben Instanz von **ShortCircuitTest** wiedergegeben werden. Es ist wahrscheinlich, dass die von **loss** und **lossZero** abgeleiteten Wechselstromwiderstände von den Gleichstrom-Widerständen der Wicklung, **TransformerEndInfo.r**, abweichen, welche aus getrennten Prüfungen gewonnen werden.

Die Attribute **openEndVoltage** und **phaseShift** von **OpenCircuitTest** werden an einer einzigen offenen Wicklung gemessen, wenn **energisedEndVoltage** an das **EnergisedEnd** angelegt ist. Die Prüfungen werden durchgeführt, um Übersetzungsverhältnis des Transformators, Wicklungsschaltung und Wicklungspolarität zu verifizieren. Sie sind üblicherweise nicht erforderlich, um elektrische Impedanzparameter zu bestimmen.

4.4.3.4.3 Stufenschalter-Model

Bild 14 zeigt die Klassen, welche verwendet werden, um einen möglicherweise unsymmetrischen Verteilungs-Spannungsregler zu modellieren. Ein **RatioTapChanger** (de: Übersetzungs-Stufenschalter) steht zu einem **TransformerEnd** in Beziehung. Jeder Regler verwendet eine unabhängige lokale Steuerung, so dass **RegulationSchedule** (de: Regelungsplan) und **TapSchedule** (de: Stufenschalterplan) (enthalten in IEC 61970-301 und üblicherweise bei Übertragung angewendet) hier nicht verwendet werden. Phasenwinkelregler und Abweichungskurven werden im Allgemeinen auch nicht in Verteilsystemen verwendet.

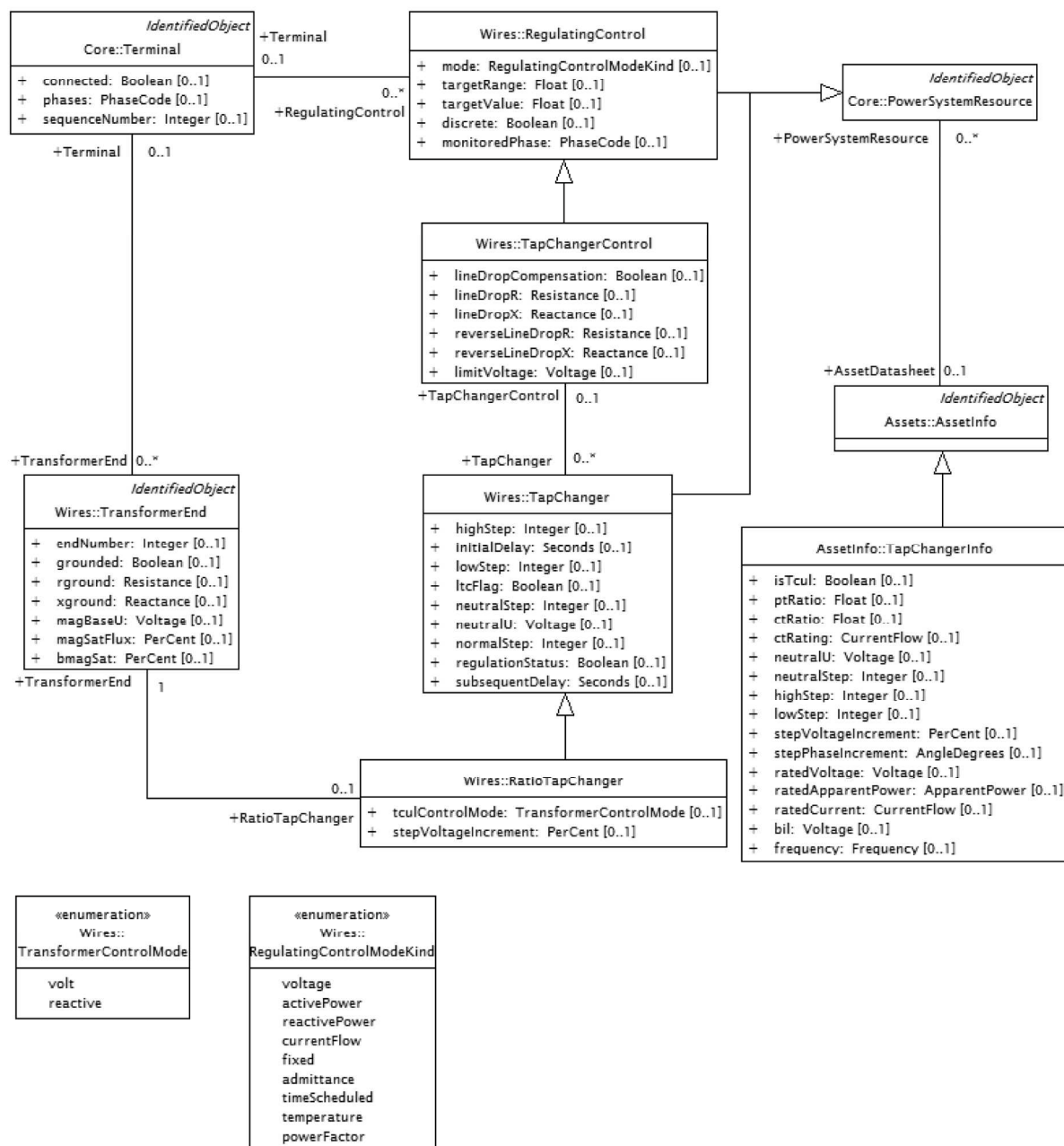


Bild 14 – DCIM Stufenschaltermodell

Ein dreiphasiger Netzspannungsregler hat üblicherweise drei unabhängige Regler, um die Korrektur von Spannungsunsymmetrien zu unterstützen. Die Regler sind üblicherweise als Stern geschaltet. Das Modell beginnt mit einem **PowerTransformer**, welcher drei **TransformerTanks** enthält, und insgesamt sechs **TransformerTankEnds**. Es wird drei Instanzen von **RatioTapChanger** geben, wobei jede zu einem anderen **TransformerTankEnd** in Beziehung gesetzt ist. Das Attribut **RegulatingControl.monitoredPhase** (de: regelnde Steuerung, überwachte Phase) sollte in die **Terminal.phase** eingeschlossen werden, welche zu **PowerTransformer** in Beziehung gesetzt ist. Die Stufenschalterpositionen und manchmal die anderen Attribute werden nicht in jeder Phase des Reglers die gleichen sein. Ein offener Dreieck-Regler ist auch einigermaßen verbreitet; dieser besteht aus zweiphasigen Reglern, welche Leitung zu Leitung in einer Bank geschaltet sind, mit teilweiser Fähigkeit zur Korrektur von Spannungsunsymmetrien.

Ein dreiphasiger Stations-Spannungsregler ändert üblicherweise alle drei Stufenschalter gemeinsam ohne die Fähigkeit zur Korrektur von Spannungsunsymmetrie. In diesem Fall ist Modellierung auf Kesselebene nicht gefordert, und das Modell darf aus einem **PowerTransformer** mit zwei **PowerTransformerEnds** bestehen. Es gäbe nur einen **RatioTapChanger**, der einem **PowerTransformerEnd** zugeordnet ist. Das Attribut **RegulatingControl.monitoredPhase** kann **A**, **B** oder **C** sein, wenn der Spannungswandler Leitung zu Erde

geschaltet ist. Es kann auch **AB**, **AC**, oder **BC** sein für Leitung zu Leitung geschaltete Spannungswandler. Üblicherweise steuert nur ein Spannungswandler diesen Typ von Regler.

4.4.3.4.4 Beispiel für Verteiltransformator

Der Transformator in Bild 15 zeigt eine offene Stern-/offene Dreieck-Bank, welche verwendet wird, um kostengünstige, dreiphasige Dienstleistungen für Kleinabnehmer zu unterstützen.

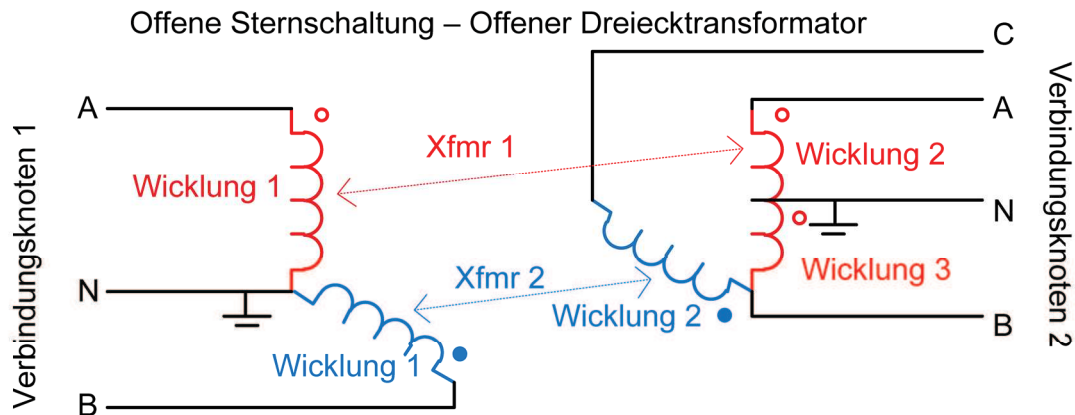


Bild 15 – Beispiel für einen Verteiltransformator, welcher mit DCIM modelliert werden kann

Tabelle 1 zeigt einige der wichtigen Attribute für dieses Beispiel. Es erfordert Modellierung auf Kesselebene.

Tabelle 1 – Offene Stern-/offene Dreieck-Transformatorbankschaltungen

Transformer Tank	Transformer TankEnd	ratedU	ratedS	Schaltungs- typ	phaseAngle Clock	Transformer TankEnd. phases
Xfmr 1	Wicklung1	7 200	100e3	1	0	AN
	Wicklung2	120	50e3	1	0	AN
	Wicklung3	120	50e3	1	6	BN
Xfmr 2	Wicklung1	7 200	50e3	1	0	BN
	Wicklung2	240	50e3	1	0	BC

Ein Wert der Phasenwinkelereignis von „6“ zeigt an, dass Wicklung 3 tatsächlich von N nach B verläuft, anstatt von B nach N. Durch die **Terminals** werden in **ConnectivityNode 1** die Phasen **ABC** vorhanden sein. Andere angeschlossene Einrichtungen, wie z. B. ein Leitungsabschnitt, könnte Phase C hinzufügen. In **ConnectivityNode 2** werden die Phasen **ABCN** vorhanden sein. Der „Beleuchtungsweig“ (Xfmr 1) hat üblicherweise andere Bemessungsdaten als der „Stromversorgungsweig“ (Xfmr 2). Das bedeutet, dass Phasen- und Bemessungszuordnungen zur Bank zweideutig sein können und folglich in **TransformerTankEnd** spezifiziert werden müssen.

4.4.3.4.5 Beispiel Spartransformator

Ein Spartransformator wird geschaffen, indem zwei Transformatorwicklungen in Reihe geschaltet werden, und somit eine metallische Verbindung zwischen den zwei Spannungsebenen besteht. Der wesentliche Vorteil ist eine Kostenersparnis, weil die MVA-Bemessung höher ist, als wenn dieselben beiden Wicklungen als konventioneller Transformator geschaltet wären. Spartransformatoren sind außerdem effizienter und weisen einen geringeren Spannungsabfall auf. Der Hauptvorteil ist möglicherweise ein höherer Kurzschlussstrom in vollständig entwickelten Systemen, weil Spartransformatoren eine niedrigere Impedanz haben. Zwei allgemeine Anwendungen sind:

- Transformierungen zwischen zwei Höchstspannungsebenen (EHV) in einem Unterwerk, wo Kostenersparnis für Wicklungsübersetzungen bis zu ungefähr 2:1 wichtig sind.

- Netzspannungsregler an Verteileinspeisungen, wo die regelnde Wicklung (kompensieren/hochsetzen) auf Sparen geschaltet ist.

Bild 16 zeigt einen Zweiwicklungstransformator auf der linken Seite mit einer Kurzschlussprüfungsschaltung am Anschluss der Wicklung L. Der gesamte Strom wird magnetisch transformiert und das Übersetzungsverhältnis ist $n_1 : n_2$. Auf der rechten Seite sind die gleichen Wicklungen als Spartransformator geschaltet. Der rote Strom fließt direkt zur Kurzschlussprüfungsschaltung am Anschluss der Wicklung L. Zusätzlich trägt der magnetisch transformierte blaue Strom ebenfalls zum Kurzschlussstrom bei, welcher höher als im Fall mit zwei Wicklungen ist. Das Übersetzungsverhältnis des Spartransformators ist $(n_1 + n_2) : n_2$. In einem Spartransformator wird die Wicklung H manchmal auch Reihenwicklung (S; en: series) genannt, und die Wicklung L wird manchmal Parallelwicklung (C; common) genannt.

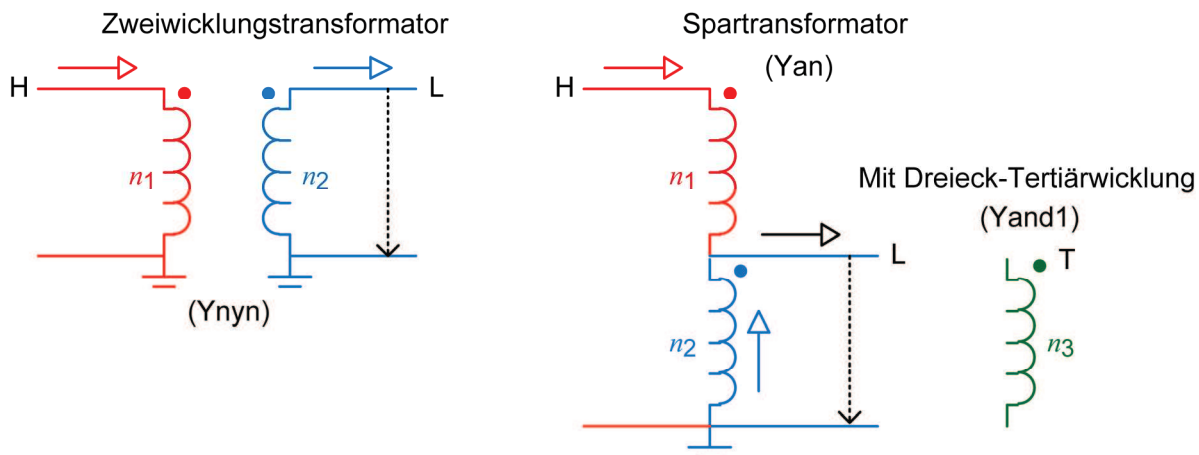


Bild 16 – Beispiel eines als Spartransformator geschalteten Zweiwicklungstransformators

Die Schaltgruppen nach IEC 60076-1 definieren Erdungs- und Phasenwechselkenngrößen jeder Wicklung, welche wichtig sind für Schutztechnik, Anpassung und andere Anwendungen – Bild 16 zeigt dies in Klammern an. „Ynyn“ bezeichnet einen Transformator mit zwei Wicklungen, wobei beide in Sternschaltung geerdet sind. (Eine äußere Neutraleiterimpedanz kann trotzdem hinzugefügt werden, wird aber nicht in der Schaltgruppe angegeben.) Der Spartransformator könnte auch mit „Ynyn“ bezeichnet werden, weil er die gleiche Neutraleiterschaltung und die gleichen Phasenkenngößen hat. Einige Transformatoranbieter verwenden jedoch „A“ und „a“, um eine Spartransformatorwicklung zu bezeichnen. Bild 16 zeigt die „Yan“-Schaltgruppe für einen Spartransformator. Obwohl der Anschluss H durch die Wicklung L einen Strompfad zum Neutraleiter hat, ist die Wicklung H nicht selbst mit dem Neutraleiter verbunden. Entsprechend IEC 60076-1 wird die Wicklung mit der höchsten Spannung in der Schaltgruppe in Großbuchstaben angegeben, während alle anderen Wicklungen in Kleinbuchstaben angegeben werden. In der Praxis bedeutet dies, dass das „a“ für einen Spartransformator immer ein Kleinbuchstabe sein wird.

Viele Spartransformatoren haben eine Dreiecksschaltung mit Tertiärwicklung, dargestellt in Bild 16. Die Schaltgruppe wäre „Yand1“, wobei sich die 1 auf eine Verzögerung von 30° (1 Uhr) gegenüber der Wicklung H bezieht. Der Wert der Maschenimpedanz Z_{HT} wird durch die Sparschaltung beeinflusst, aber nicht der Wert der Maschenimpedanz Z_{LT} . Die Auswirkung auf Z_{HT} ist hier nicht dargestellt, ist aber in mehreren technischen Verweisungen zu finden. Die Umrechnungen von Zweiwicklungsdaten (linke Seite von Bild 16) in Übersetzung des Spartransformators (N), Volt-Ampere-Bemessungswert (S_{auto}) und Maschenimpedanz (Z_{auto}) auf der rechten Seite von Bild 16 sind:

$$N = 1 + n_1 / n_2$$

$$S_{\text{auto}} = S_{2\text{-Wicklungen}} [N / (N - 1)]$$

$$Z_{\text{auto}} = Z_{2\text{-Wicklungen}} [(N - 1) / N]^2$$

Je Einheit (pu; en: per unit) ist die umgerechnete Impedanz des Spartransformators ($Z_{\text{pu-auto}}$):

$$Z_{\text{pu-auto}} = Z_{\text{pu-2-Wicklungen}} / N$$

Es wird zum Beispiel angenommen, ein Zweiwicklungstransformator hat 115/115 kV, bemessen mit 100 MVA, mit 10 % Impedanz auf 100 MVA. Die Übersetzung ist 1:1. Von jeder Wicklung her betrachtet, ist die Kurzschlussimpedanz 13,225 Ω. Als ein Spartransformator, 230/115 kV, geschaltet ist die Übersetzung 2:1 (N = 2) und der Bemessungswert ist 200 MVA. Vom 115-kV-Anschluss her betrachtet ist die Kurzschlussimpedanz 3,306 25 Ω, was bei dem neuen Bemessungswert von 200 MVA 5 % entspricht. Die Kosten von Eisenkern und Kupferwicklung entsprechen der Hälfte der Kosten für einen konventionellen Zweiwicklungstransformator mit den gleichen Bemessungswerten. Die Gesamtkosten liegen bei etwas mehr als 50 %, weil die Leiter der Wicklung L mehr Strom führen müssen und die Wicklung H für höhere Spannung isoliert werden muss. Der Prüfungsbogen für diesen Spartransformator würde eine Spannungsübersetzung von 230/115 kV angeben, einen Bemessungswert von 200 MVA und eine Kurzschlussimpedanz von 5 %.

Es ist üblich, einen Spartransformator als einen konventionellen Zweiwicklungstransformator zu modellieren, wobei Daten aus dem Prüfungsbogen verwendet werden und die Tatsache vernachlässigt wird, dass die Wicklungen tatsächlich in Reihe geschaltet sind. Es gibt jedoch Situationen, in welchen der Unterschied wichtig ist, wie z. B. genauere Kernmodellierung oder genauere Modellierung der Impedanz gegenüber der Stufenschalterkenngröße. Wenn die Reihen- und Parallelwicklungen identifiziert wurden, ist es möglich, das physikalische Spartransformatormodell in Bild 16 abzuleiten.

Ein Spartransformator sollte im CIM mit konventionellen Zweiwicklungsdaten für Impedanzen, Admittanzen und Bemessungswerten modelliert werden, wie es üblicherweise in Prüfberichten von Spartransformatoren zu finden ist. Jede physikalische Wicklung wird in dem CIM ein zugehöriges **PowerTransformerEnd** oder **TransformerTankEnd** haben. Es ist wahlfrei, die Spartransformatorschaltung mit einem Attribut **A** für **connectionKind** am parallel geschalteten Ende und mit „an“, als Teil von **PowerTransformer.vectorGroup** für diese Ende vorkommend, zu spezifizieren. Das in Reihe geschaltete Ende muss dann einen Attributwert **Y** für **connectionKind** haben. Die **endNumbers** für Reihen- und Parallelschaltung sollten 1 beziehungsweise 2 sein. Die Zielanwendung kann dann, wenn erforderlich, das physikalische Spartransformatormodell ableiten. Um Spartransformatorschaltungen im Modell zu vernachlässigen, ist **Yn** für **connectionKind** an den Enden in Reihen- und Parallelschaltung festzulegen, es gibt keine Einschränkungen für **endNumbers**. Das Attribut **connectionKind** erscheint in **TransformerEndInfo**, wenn eine Modellierung auf Kesselebene verwendet wird, und in **PowerTransformerEnd**, wenn keine Modellierung auf Kesselebene verwendet wird. Es ist zu beachten, dass **A**, **Y**, **D** und **Z** in **connectionKind** immer Großbuchstaben sind, aber immer, wenn die **endNumber** größer 1 ist, sollten sie in der **vectorGroup** Kleinbuchstaben sein.

4.4.3.5 Hilfseinrichtungen

Die vorliegende Ausgabe von IEC 61968-11 umfasst die Mittel zur Modellierung von Hilfseinrichtungen mittels des Modellpakets **IEC 61970::Wires** nach IEC 61970-301, dargestellt in Bild 17.

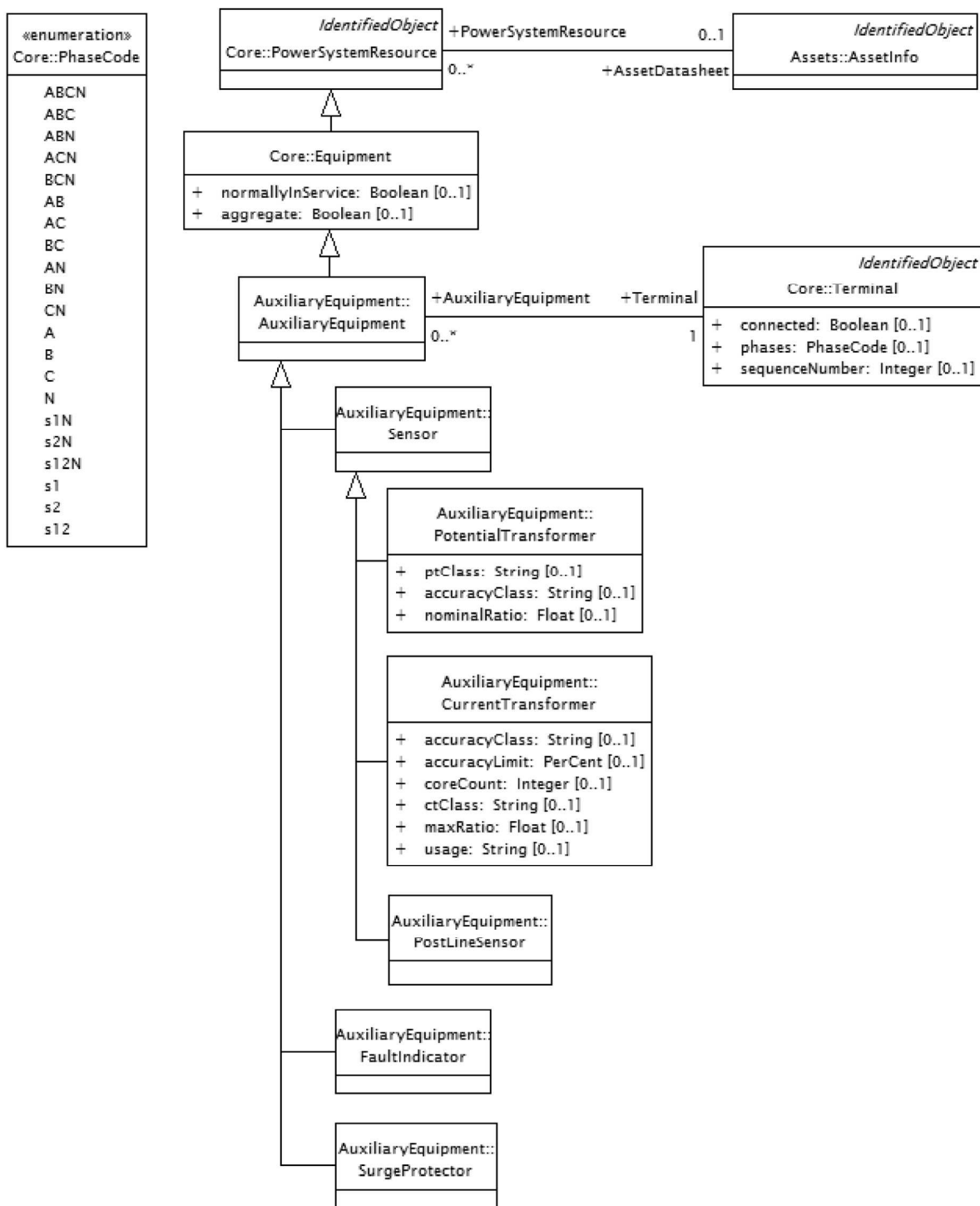


Bild 17 – DCIM Hilfseinrichtungen

Eine wichtige Ergänzung ist es, ein **Terminal** zu jedem **AuxiliaryEquipment** in Beziehung zu setzen, wodurch die Positionierung der Hilfseinrichtungen im Modell der Verbindungsfähigkeit des elektrischen Netzes möglich wird. Zum Beispiel ist es nun möglich, Messwandler explizit zu modellieren, sie in Übersichtsschaltbildern zusammen mit dem üblichen **ConductingEquipment** zu positionieren und die **Measurements** zu bestimmen, welche sie tatsächlich erzeugen. Diese Art von Datenaustausch würde typisch sein, wenn Unterwerksmodelle importiert werden, welche mit anderen Datenmodellen als CIM definiert wurden.

Außerdem ist es nun möglich, die „Platzierung“ im Netz von Instanzen von **FaultIndicator** im Netz auszutauschen, wie es für Zwecke der Fehlereingrenzung, Isolierung und Wiederherstellung angewendet

wird, um die Entsendung von Betriebspersonal in den Teil des Netzes (**ACLineSegment**) zu unterstützen, in dem das Auftreten des Fehlers am wahrscheinlichsten ist.

Die Ergänzung der Klasse **Sensor** (de: Messwertaufnehmer) liefert den Punkt der Erweiterbarkeit für andere Arten von Messeinrichtungen als Messwandler und **PostLineSensor** (de: Mastleitungsmesswertaufnehmer).

ANMERKUNG Im vorliegenden Teil von IEC 61968 (dokumentiert DCIM11) gibt es keine Unterstützung für die Datenblattmodellierung jeglicher dieser Hilfseinrichtungen, d. h. Bild 18 zeigt die Klasse **AssetInfo** ohne Spezialisierungen. Es ist geplant, die zugehörigen Assetmodellklassen in der nächste Ausgabe zu ergänzen, um eine detailliertere Beschreibung dieser Assets zur Unterstützung des nach IEC 61968-3 und IEC 61968-4 geforderten Datenaustauschs zu ermöglichen.

4.4.4 Kundenmodell

Das Paket **Customer** (de: Kunde) führt Klassen ein, dargestellt in Bild 18, welche für die unternehmensseitige Integration von Kundeninformationen und Abrechnungssystemen und die Informationen, die sie mit anderen Unternehmenssystemen austauschen, erforderlich sind.

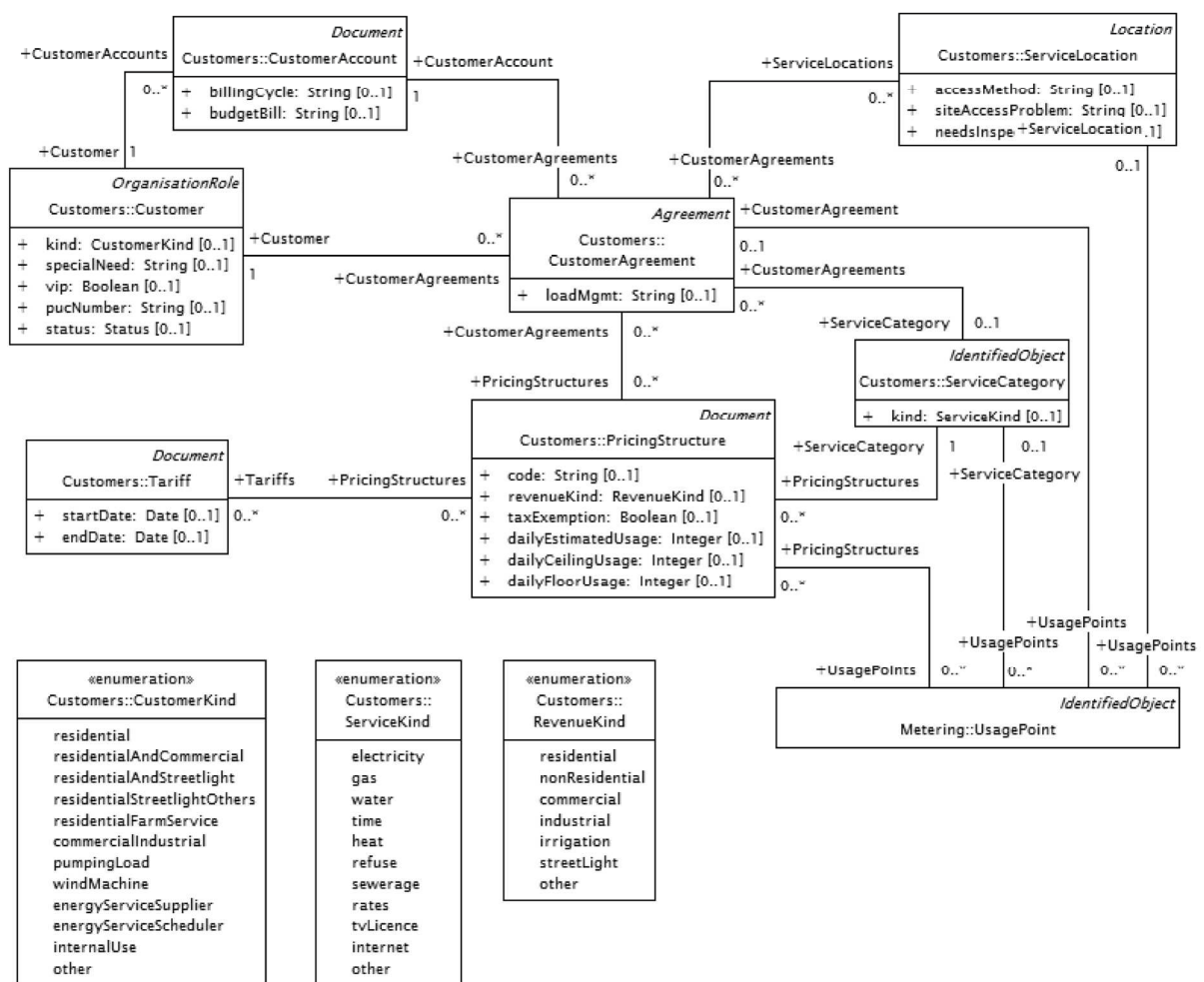


Bild 18 – DCIM-Kundenmodell

Ein **Customer** ist eine Organisation, welche Dienstleistungen von einem oder mehreren **ServiceSuppliers** (de: Dienstleister) empfängt. Ein **Customer** kann viele **CustomerAccounts** (de: Kundenkonten) und/oder **CustomerAgreements** (de: Kundenvereinbarungen) haben.

Ein **CustomerAccount** ist der Mechanismus für Kundenabrechnung und -zahlung. Es wird verwendet, um Abrechnungen (Rechnungen) für **Customer** zu erstellen und Zahlungen zu empfangen. Ein **CustomerAccount** wird üblicherweise in einem definierten Zeitplan/Abrechnungszyklus abgerechnet. Ein **CustomerAccount** kann

die zusammengefasste Abrechnung vieler **CustomerAccounts** unterstützen. Folglich können viele **CustomerAgreements** einem **CustomerAccount** zugewiesen sein.

Ein **CustomerAgreement** ist eine Vereinbarung zwischen dem **Customer** und dem **ServiceSupplier**, an einem spezifischen Punkt der Erbringung einer Dienstleistung (**UsagePoint**, siehe Verbrauchszählungsmodell in 4.4.5.1) für die Dienstleistung zu bezahlen. Es identifiziert bestimmte Abrechnungsinformationen, welche während der Erstellung der **Charge** (de: Forderung) verwendet werden, einschließlich der **PricingStructure** (de: Preisstruktur).

Eine **PricingStructure** ist eine Gruppierung von Preiselementen und Preisen, welche in der Erstellung der **Charges** für Kunden verwendet werden, und den Kriterien für Wahlmöglichkeiten, unter welchen diese Bedingungen einem Kunden angeboten werden. Die Gründe für die Gruppierung schließen Gerichtsbarkeit, Kundenklassifizierung, Standortmerkmale, Klassifizierung (d. h. Preisstruktur der Forderung, Preisstruktur der Vorauszahlung, Preisstruktur der elektrischen Dienstleistungen usw.) und Buchführungsanforderungen ein. Eine **PricingStructure** kann einen oder mehrere **Tariffs** (de: Tarife) umfassen, welche häufig als Gebührensatz bezeichnet werden.

CustomerAccount, **PricingStructure** und **CustomerAgreement** sind alle Spezialisierungen von **Document**, entweder direkt oder durch die Klasse **Agreement**. Dies demonstriert die vorgesehene Verwendung der Klasse **Document** mittels ihrer zweckmäßigen Spezialisierung, und niemals mittels der direkten Verwendung der Klasse **Document** für Beziehungen.

Eine **ServiceCategory** (de: Dienstleistungskategorie) ist die Kategorie (Elektrizität, Wasser, Gas usw.) der bereitgestellten Dienstleistung. Ein **ServiceSupplier** ist die Organisation, welche dem Kunden die Dienstleistung bereitstellt. Es gibt viele Typen von **ServiceSuppliers** (z. B. Versorgungsunternehmen, Händler und andere). Ein gegebener **ServiceSupplier** kann Dienstleistungen an viele Instanzen von **ServiceCategory** bereitstellen.

Eine **ServiceLocation** ist der Standort einer Immobilie, üblicherweise bezeichnet als das Gebäude, zu welchem eine oder mehrere Dienstleistungen geliefert werden können. Er kann mehrere **UsagePoints** haben; jeder **UsagePoint** wird jedoch für eine und nur eine **ServiceCategory** vorhanden sein.

ANMERKUNG Dieser Teil von IEC 61968 (dokumentiert DCIM11) stellt die Kundenmodellierung nur für die Erfordernisse von IEC 61968-9 bereit. Die nächsten Ausgaben werden auch umfassendere Modelle zur Unterstützung von IEC 61968-8 und IEC 61968-6 bereitstellen.

4.4.5 Verbrauchszählungsmodell

4.4.5.1 Allgemeines

Das Paket **Metering** (de: Verbrauchszählung) führt die Klassen ein, welche wesentlich für die Integration von Verbrauchszählungssystemen in Unternehmen sind, und die Informationen und Steuerbefehle, welche sie mit anderen Unternehmenssystemen austauschen.

Bild 19 veranschaulicht Klassen, welche für die Domäne der Verbrauchszählung spezifisch sind¹⁴⁾; siehe auch Bild 18 in 4.4.4 für kundenbezogene Klassen, welche ebenfalls in der Domäne der Verbrauchszählung maßgeblich sind.

¹⁴⁾ Aufgrund der Anzahl enthaltener Klassen sind im Bild nur die Namen und Beziehungen dargestellt; 4.4.5.2 bis 4.4.5.7 gehen genauer auf Teilgruppen und bezogene Klassen ein.

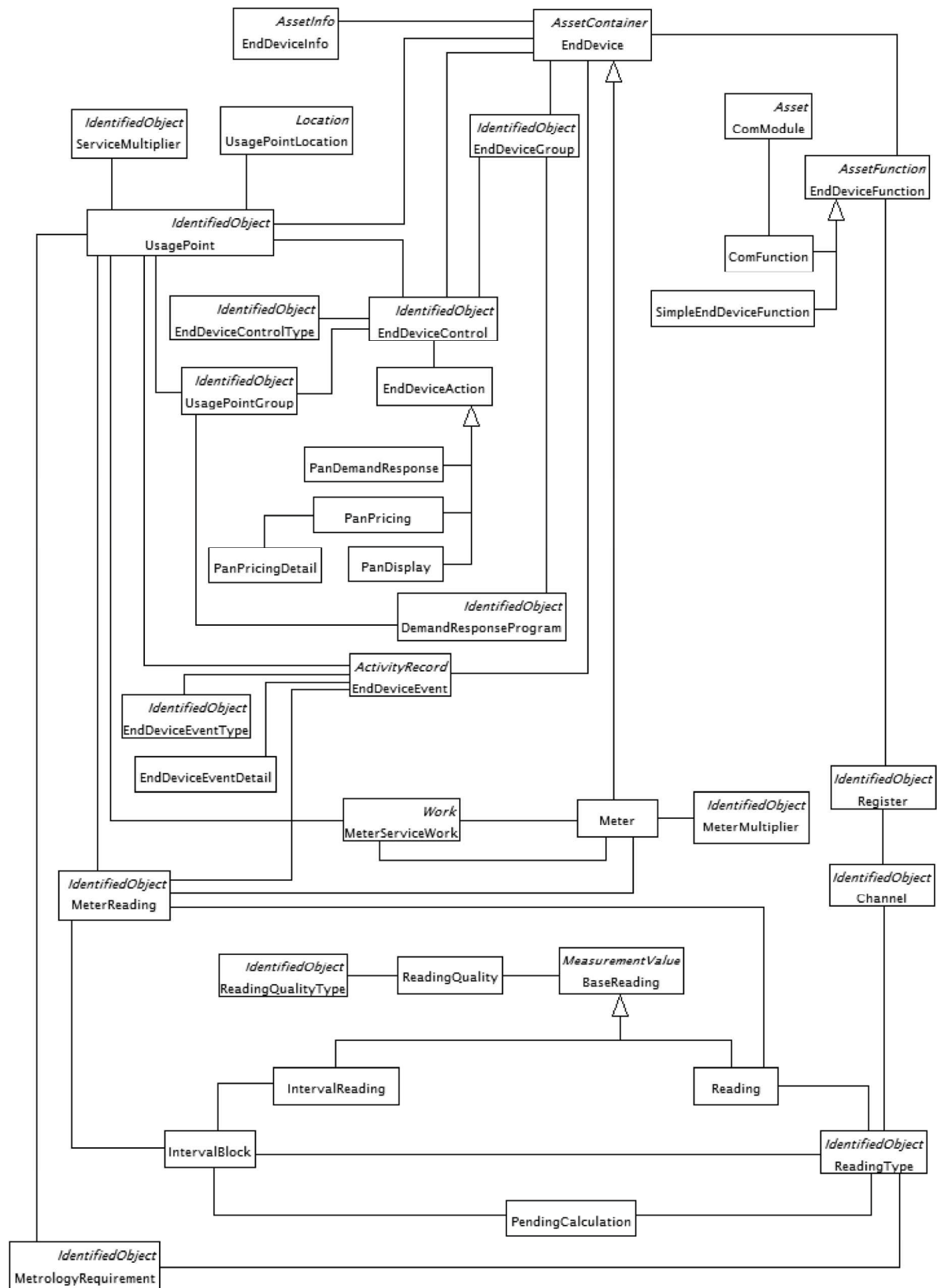


Bild 19 – DCIM-Verbrauchszählungsmodell

Diese Klassen unterstützen sowohl typische Einheiten in der Domäne der Verbrauchszählung als auch Konzepte, welche für betriebsbedingten Datenaustausch erforderlich sind. 4.4.5.2 bis 4.4.5.7 beschreiben bezogene Teilgruppen detaillierter. 4.4.5.2 bis 4.4.5.4 konzentrieren sich im Wesentlichen auf die Einheiten der Domäne, welche üblicherweise konfiguriert werden müssen, um den betriebsbedingten Datenaustausch

zu ermöglichen. Diese Typen von betriebsbedingtem Datenaustausch sind detaillierter in 4.4.5.5 bis 4.4.5.7 beschrieben.

4.4.5.2 Nutzungspunkte

Ein Schlüsselbegriff im Verbrauchszählungsmodell ist der **UsagePoint**, welcher mit seinen Beziehungen in Bild 20 dargestellt ist.

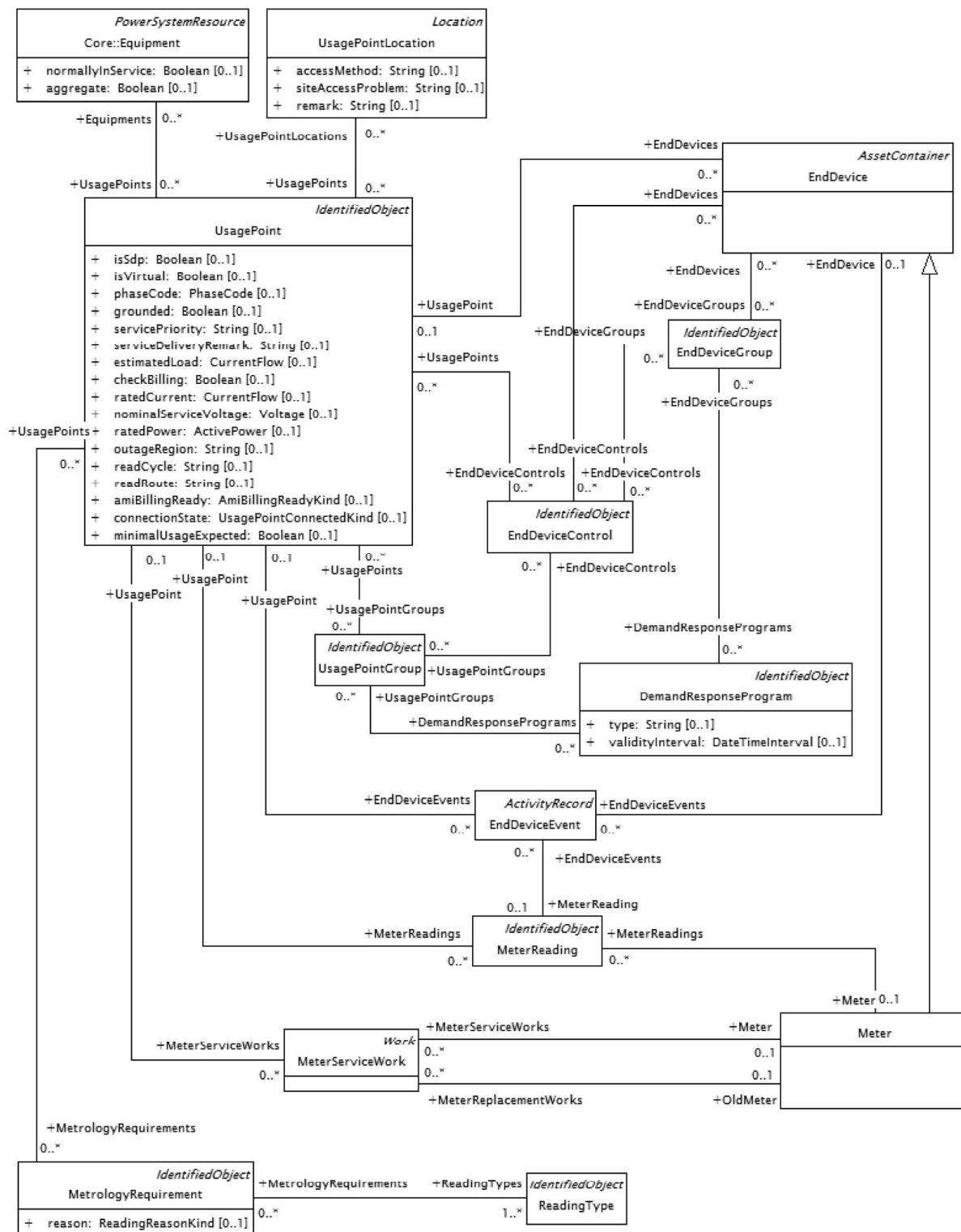


Bild 20 – DCIM-Nutzungspunktmodell

Der **UsagePoint** ist der logische und physikalische Punkt eines Netzes, welchem **MeterReadings** (de: Zählerablesungen) oder **EndDeviceEvents** zugeordnet werden können. Er wird an der Stelle verwendet, wo ein physikalischer/s oder logischer/s Zähler oder **EndDevice** angeordnet sein kann, wobei jedoch nicht erforderlich ist, dass dieses Gerät vorhanden ist.

Ein **UsagePoint** kann der Punkt der Erbringung einer Dienstleistung sein, welcher als Punkt der Abgrenzung in einem Netz definiert ist, an welchem die Eigentümerschaft vom **ServiceSupplier** zum **Customer** übergeht (siehe Bild 18).

Ein **UsagePoint** darf keine oder mehrere **MetrologyRequirements** (de: messtechnische Anforderungen) haben, welche die **ReadingTypes** (de: Ablesearten) definieren, welche für den **UsagePoint** ermittelt oder berechnet werden müssen.

UsagePointLocation (de: Ort des Nutzungspunkt) definiert den physikalischen Ort des **UsagePoint**.

MeterServiceWorks (de: Zählerinstandhaltungsarbeiten) sind jedem **UsagePoint** dynamisch zugeordnet und ermöglichen eine explizite Spezifizierung, wenn die Arbeit im Austausch eines Zählers mittels seiner Beziehungsendung **OldMeter** (de: alter Zähler) besteht.

UsagePoint ist die Klasse, welche die Verbindung zwischen dem netzorientierten Modell elektrisch verbundener Einrichtungen einerseits, und dem asset- und gebäudeorientierten Modell der Domäne der Verbrauchszählung andererseits bereitstellt. Die Verbindung kann durch die Beziehung von **UsagePoint** zu **Equipment** geschaffen werden, jener im Modellpaket **IEC61970::Wires** nach IEC 61970-310 definierten Klasse. Diese Beziehung ermöglicht den Austausch der Konfiguration von Beziehungen zwischen mindestens den folgenden Konzepten in den zwei Domänen:

- Ein **TransformerTank** liefert Strom an einen oder mehrere **UsagePoints**. Die Kenntnis dieser Beziehung wird durch bestimmte Funktionen des Ausfallmanagements und des Verteilnetzmanagements gefordert.
- Ein oder mehrere **UsagePoints** können zusammengefasste **EndConsumer** repräsentieren, wie vom typischen Leitzentralensystem aus gesehen.
- Messwandler (**CurrentTransformer** (de: Stromwandler) oder **PotentialTransformer** (de: Spannungswandler)), welche am **UsagePoint** physikalisch installiert sind, können durch diese Beziehung explizit modelliert werden, und ihre Datenblattinformation kann in einem System definiert und mit dem anderen System gemeinsam verwendet werden.

Usage Points können dynamisch in mehrere **UsagePointGroups** gruppiert werden, welche auf Betriebsabläufe einschließlich **MeterReadings**, **EndDeviceControls** (de: Endgerätesteuerungen) und potentiell andere Verbrauchszählungsfunktionen zu definierten Gruppen von **EndDevices** und/oder **UsagePoints** ausgerichtet sind. Nachrichten zwischen Unternehmenssystemen werden verwendet, um das Verständnis jedes Systems und die Mitgliedschaft jeder Gruppe zu synchronisieren.

4.4.5.3 Endgeräte

Ein weiterer wesentlicher Begriff in der im DCIM modellierten Domäne der Verbrauchszählung ist der der **EndDevices**, dargestellt in Bild 21.

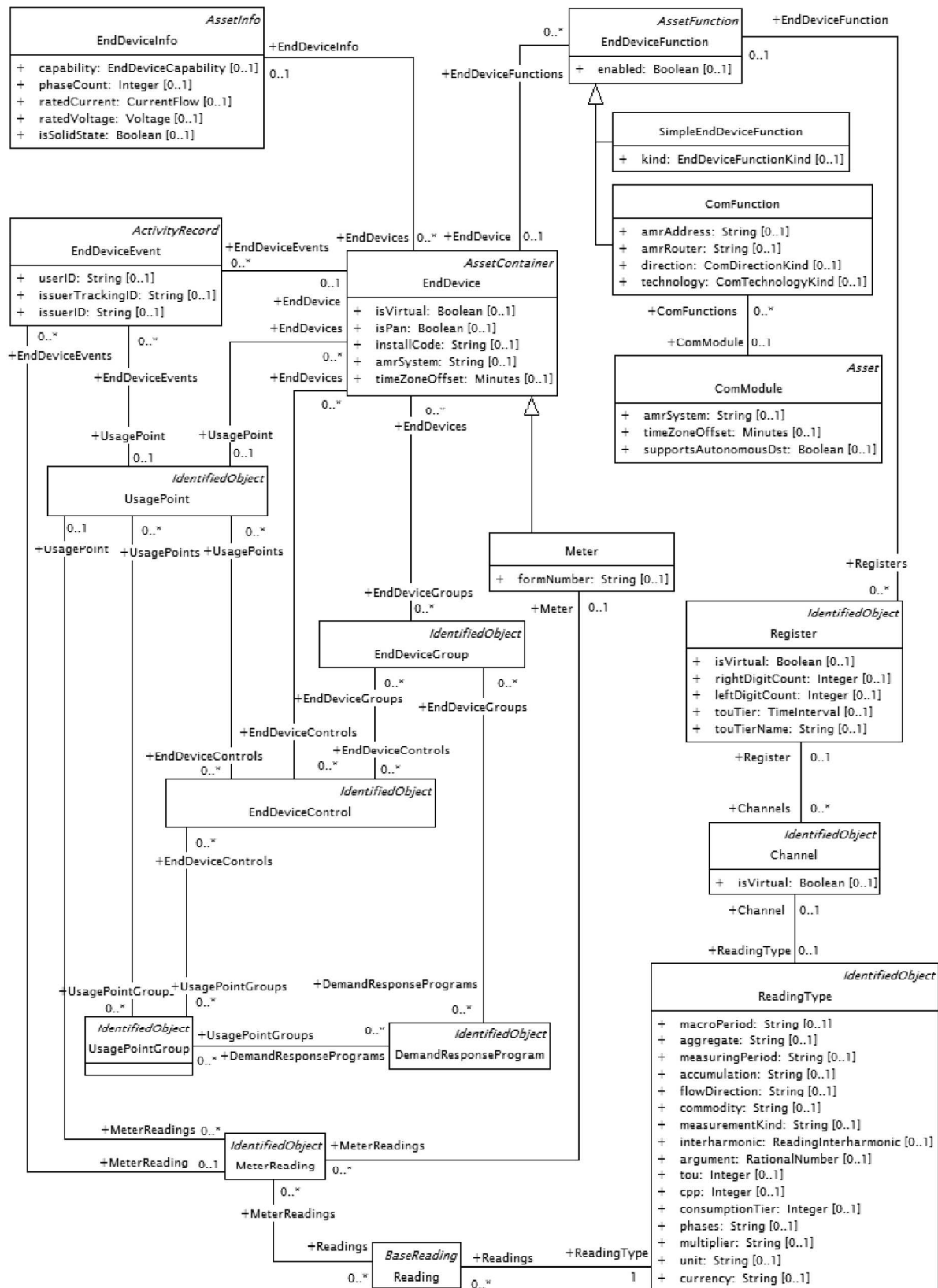


Bild 21 – DCIM-Endgerätemodell

Ein **EndDevice** ist ein **AssetContainer**, welcher eine oder mehrere **EndDeviceFunctions** ausführt. Ein Typ von **EndDevice** ist ein **Meter**, welches Verbrauchszählung, Lastmanagement, Verbinden/Trennen und andere Funktionen ausführen kann. Ein **EndDevice** kann auch verwendet werden, um ein Gebäudenetz (PAN; en: premises area network) zu modellieren, welches Funktionen wie Haushaltsanzeige und -regelung und

Überwachung von gebäudebasierten Einrichtungen, wie z. B. Klimaanlage, Kühlschränke, Schwimmbeckenpumpen usw., ausführen kann. Ein oder mehrere PAN-Geräte können mit einem **Meter** oder einem **UsagePoint** verbunden sein.

Mehrere **EndDevices**, folglich auch mehrere **Meter**, können mit einem gegebenen **UsagePoint** verbunden werden. Es sind jedoch verhältnismäßig häufig Unternehmenssysteme anzutreffen, welche einen einzelnen **UsagePoint** für jeden **Meter** erfordern.

EndDevices dürfen Kommunikationsfähigkeiten haben, welche in das **EndDevice** integriert sind. Außerdem darf die Kommunikationsfähigkeit des **EndDevice** durch ein oder mehrere **ComModules** (de: Kommunikationsmodul) bereitgestellt werden, wobei diese Einschließung von der Beziehung **AssetContainer-Asset** übernommen wurde.

Ein **Meter** kann ein physikalisches oder logisches **EndDevice** sein, welches die Funktion der Verbrauchszählung ausführt. Ein physikalischer Zähler ist vorhanden, wenn es eine reale Hardware gibt, und kann zum Beispiel definiert werden, um Funktionen wie Zusammenfassen des Verbrauchs für zwei oder mehr physikalische Zähler auszuführen.

Meter können eine oder mehrere **Register** (de: Verzeichnis) haben, welche Geräte sind, die Einheiten eines Erzeugnisses oder einer anderen Menge anzeigen oder aufzeichnen. Jedes Register kann seinerseits mehrere **Channels** (de: Kanäle) haben. Jeder **Channel** ist jedoch für nur einen **ReadingType** reserviert.

Ein **Channel** ist ein einzelner Pfad für die Sammlung oder Aufzeichnung der Werte eines **Registers** über eine Dauer. Zum Beispiel kann ein **Register**, welches die gehandelte Energie misst, zwei **Channels** haben: einen zur Bereitstellung der Ablesung der Gesamtmenge und einen zur Bereitstellung von Intervallablesungen einer festen Intervallgröße.

Eine Instanz von **ReadingType** wird als eindeutiger Bezeichner verwendet, welcher die für eine vollständige Beschreibung einer **Reading** (de: Ablesung) erforderlichen Attribute spezifiziert. Eine **Reading** ist ein spezifischer Wert, welcher durch ein **Meter** oder ein System gemessen oder berechnet wurde.

ANMERKUNG Attribute von **ReadingType**, derzeit im UML-Modell als Zeichenkette definiert, werden beim Datenaustausch ganzzahlige Werte enthalten, beschrieben in IEC 61968-9:2009, Anhang C. Es ist beabsichtigt, diese Werte in einer der nächsten Ausgaben von IEC 61969-11 als buchstabengetreue Aufzählung zu modellieren. Für die folgenden Attribute von **ReadingType**: **phases**, **multiplier** (de: Multiplikator), **unit** (de: Einheit) und **currency** (de: Währung), wird eine Harmonisierung mit den Typen der buchstabengetreuen Aufzählung erforderlich sein, welche im Modellpaket **IEC61970::Domain** nach IEC 61970-301 definiert sind.

4.4.5.4 Konfigurationsereignisse

Im Gegensatz zu Modellen von elektrischen Netzen, in welchen es üblicherweise ein System als Stammdatenquelle gibt, können in der Domäne der Verbrauchszählung mehrere Aufzeichnungssysteme vorhanden sein, wobei jedes eine Stammdatenquelle für nur eine Teilmenge von Geschäftseinheiten ist. Folglich erfordert die Nachverfolgung von Konfigurationsänderungen eine hohe Flexibilität und wird deshalb mit Instanzen von **ConfigurationEvent** modelliert, wie in Bild 22 dargestellt.

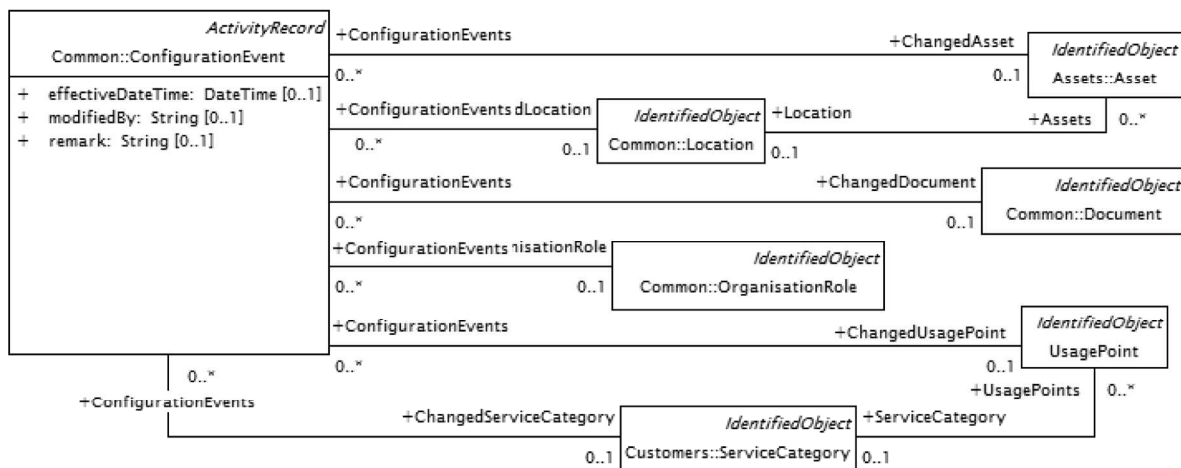


Bild 22 – Konfigurationsereignisse für Zählung

ConfigurationEvent ist eine Spezialisierung von **ActivityRecord**, welche ausschließlich zum Datenaustausch über Konfigurationsänderungen für verschiedene Objekte verwendet wird: **Assets**, **Locations**, **Documents**, **Organisations**, **UsagePoints** und Instanzen von **ServiceCategory**.

4.4.5.5 Zählerablesungen

Zählerablesungen werden zwischen Unternehmenssystemen ausgetauscht, um Funktionen wie Abrechnung, Bedarfsbefriedigung, Lastmanagement, Systemplanung und Einnahmesicherung zu unterstützen. Bild 23 veranschaulicht Klassen, welche diese Art von Datenaustausch unterstützen.

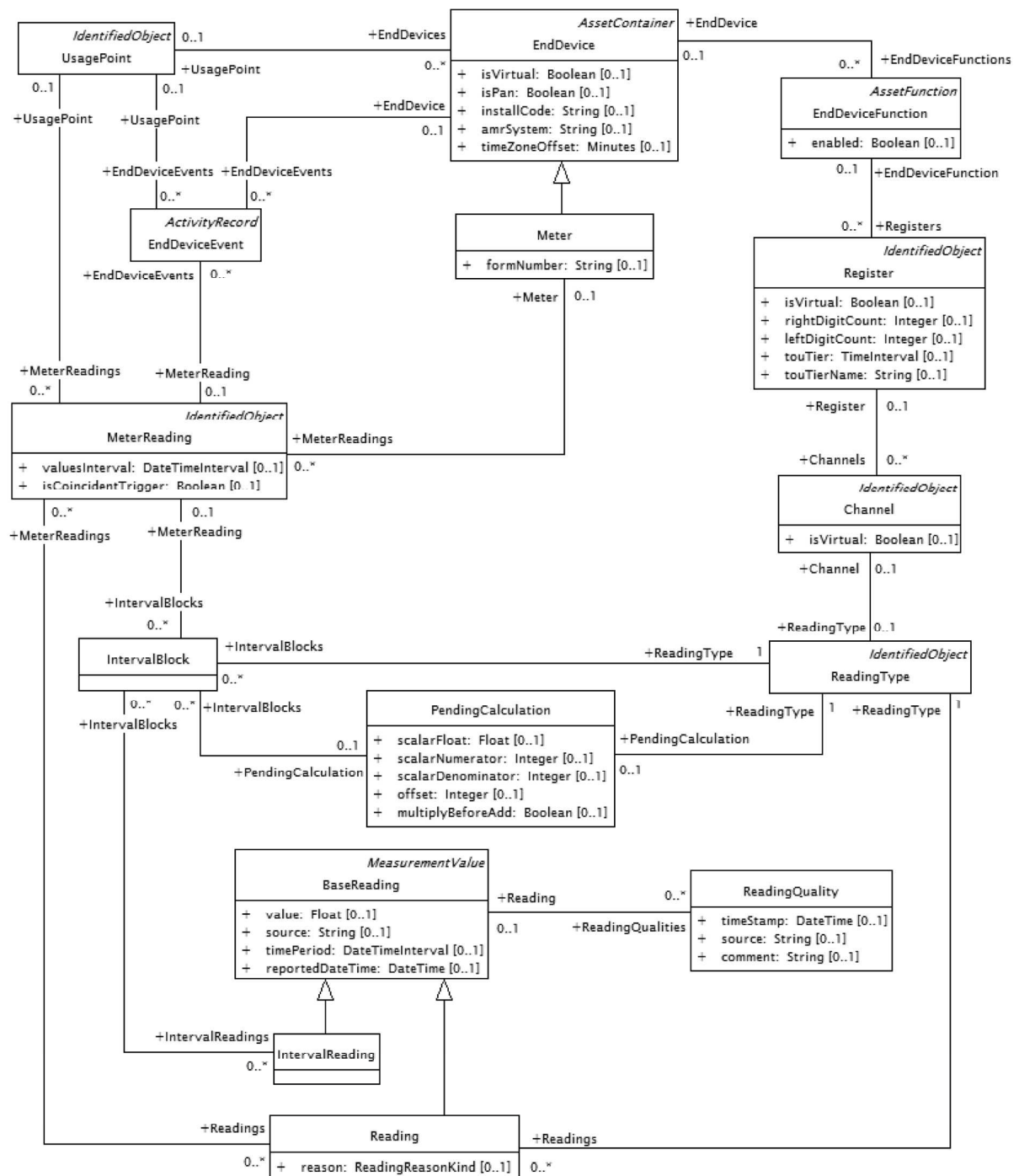


Bild 23 – DCIM-Zählerablesungsmodell

MeterReadings können aus mehreren Kombinationen von **Readings** und **IntervalBlocks** (de: Intervallblöcke) bestehen. **Readings** repräsentieren Werte (entweder gemessen oder berechnet) zu einem spezifischen Zeitpunkt und gekennzeichnet durch einen **ReadingType** und eine oder mehrere Instanzen von **ReadingQuality** (de: Ablesungsqualität). **IntervalBlocks** repräsentieren Sammlungen von **Readings** in gleichmäßig bemessenen Zeitintervallen.

Ein typische Verwendung für **Readings** ist die eines Zählungssystems oder eines Systems des Zählerdatenmanagements, welches ein Abrechnungssystem mit nächtlichen Verzeichnisablesungen bereitstellt. **IntervalBlocks** können verwendet werden, um ein Kundeninformationssystem mit genauer getakteten Verbrauchsablesungen bereitzustellen, zum Beispiel für 15-minütige oder stündliche Perioden.

Letzteres kann verwendet werden für Abrechnung oder Aufteilung des Verbrauchs in Nutzungszeiten, welche mit unterschiedlichen Tarifen abgerechnet werden können.

ANMERKUNG Obwohl sowohl **Reading** als auch **IntervalReading** die Klasse **MeasurementValue** (de: Messwert), definiert im Modellpaket **IEC61970::Meas** nach IEC 61970-301, spezialisieren, gibt es wenige Gemeinsamkeiten zwischen diesen zwei Modellen für Messungen. Das einzige aus **MeasurementValue** verwendete Attribut ist **timeStamp** (de: Zeitstempel); keine seiner Beziehungsendungen wird verwendet. Selbst wenn eine Beziehung einer **Reading** zu einer Instanz von **Measurement** möglich ist, sind Quelle, Wert, Qualität und Typdefinition völlig unterschiedlich. Folglich bietet das Standard-DCIM derzeit für Anwendungen oder Systeme, welche gerne eine **Reading** als einen **MeasurementValue** verarbeiten würden, oder umgekehrt, keine Unterstützung für eine Transformation zwischen den beiden Darstellungen.

4.4.5.6 Steuerungen und Ereignisse von Endgeräten

Bestimmte Endgeräte sind in der Lage, Vorgänge als Antwort auf Steuerungen auszuführen oder das Auftreten von bestimmten Ereignissen zu erkennen und spezifische Details bezüglich dieser Ereignisse aufzuzeichnen oder an andere Systeme zu übermitteln. Klassen des DCIM-Modells, welche diese Art des Datenaustauschs unterstützen, sind in Bild 24 dargestellt.

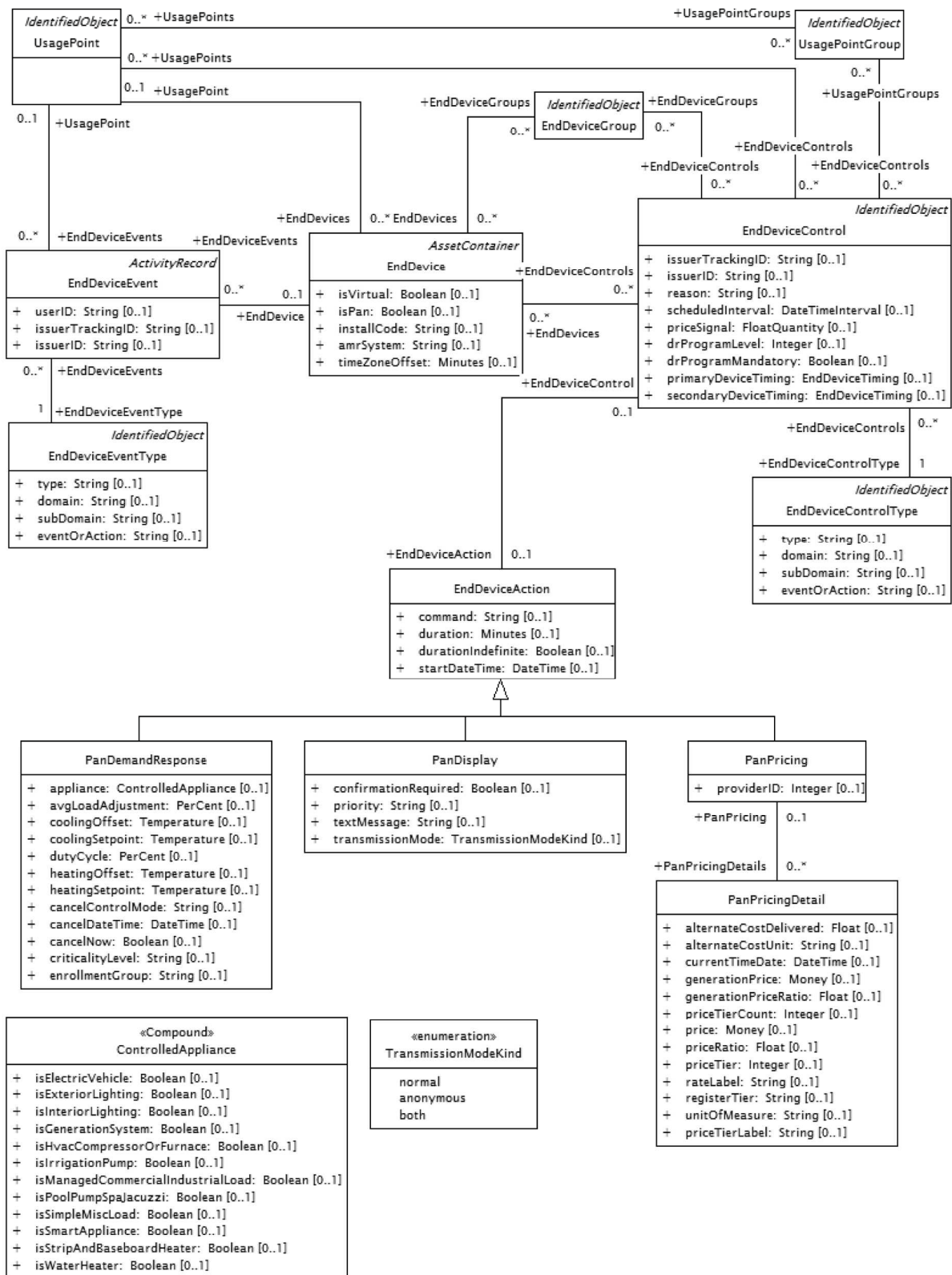


Bild 24 – DCIM-Modell von Steuerungen und Ereignissen von Endgeräten

Ein **EndDevice** kann in der Lage sein, auf Befehle eines **EndDeviceCommand** (de: Endgerätbefehl) hin zu agieren. Beispiele umfassen das automatische Verbinden oder Trennen der Versorgung durch einen **Meter**, Zurücksetzen des Bedarfsverzeichnisses eines **Meter**, Anzeige einer Textnachricht auf einem PAN-Gerät, Kommunizieren von Preisinformationen an ein PAN-Gerät oder ein anderes **EndDevice** usw.

Befehle von **EndDeviceControl** werden üblicherweise von Unternehmenssystemen angefordert, wie z. B. ein Kundeninformationssystem, und werden letztendlich zur Ausführung an das Verbrauchszählungssystem weitergegeben. Eine **EndDeviceControl** spezifiziert die verlangten Steuervorgänge, welche einen **EndDeviceControlType** (de: Steuerungstyp des Endgeräts) verwenden. Das Unternehmenssystem kann eine Steuerung für die unverzügliche Ausführung oder die Ausführung zu einem späteren Datum oder einer späteren Zeit anfordern.

Systeme können **EndDeviceControl** auf einzelne **EndDevices** (oder **Meters**) oder **UsagePoints** ausgerichtet sein. Alternativ können sie auf eine oder mehrere **EndDeviceGroups** oder **UsagePoints** ausgerichtet sein.

Bestimmte **EndDevices** können programmiert werden, um andere Einrichtungen zu steuern, welche nicht direkt den **EndDeviceControls** unterliegen. Zum Beispiel können Details eines Ereignisses des Lastmanagements an ein PAN-Gerät gesendet werden (unter Verwendung von **EndDeviceControl**), woraus letztendlich das Ein- oder Ausschalten von Einrichtungen, wie z. B. Klimaanlage, Schwimmbadenpumpen usw. resultiert. Für diesen Datenaustausch wird die zum „Programmieren“ des PAN-Gerätes erforderliche Information in der **EndDeviceControl** mit zugeordneter **EndDeviceAction** (de: Endgeräteaktion) (welche in **PanDemandResponse** (de: PAN-Gerät-Bedarfsbefriedigung), **PanPricing** (de: PAN-Gerät-Preisermittlung) oder **PanDisplay** (de: PAN-Gerät-Anzeige) weiter spezialisiert wird) kommuniziert.

Bild 24 zeigt auch die Klassen, welche für die Modellierung von **EndDeviceEvents** maßgeblich sind; sie werden durch **EndDevices** erkannt und erfasst und an andere Unternehmenssysteme übertragen. **EndDeviceEvents** werden häufig in einem Verbrauchszählungssystem erzeugt (welches in direkter Kommunikation mit dem **EndDevice** steht) und werden als Auslöseereignisse für die Ausführung von logischen Geschäftshandlungen an andere Systeme übertragen. Beispiele von **EndDeviceEvents** umfassen Verlust oder Wiederherstellung der Stromversorgung, Verlust oder Wiederherstellung der Kommunikationsfunktionen, Alarmierungen bei Zählermanipulation, Signalzeichen für Rückwärtslauf und viele, viele andere. Alarme zum Verlust der Stromversorgung könnten beispielsweise zur Diagnose des Ausmaßes des Ausfalls an ein Ausfallmanagementsystem kommuniziert werden. Oder Rückwärtslauf- und Manipulationsalarme können an ein System übermittelt werden, welches für die Erkennung von Stromdiebstahl oder anderer Einnahmeaspekte verantwortlich sind.

Ein **EndDeviceEvent** spezifiziert das Ereignis unter Verwendung von **EndDeviceEventType**.

Durch **EndDevices** erkannte Ereignisse können spontan (in unaufgeforderter Weise) oder als ein direktes Ergebnis eines **EndDeviceControl**-Befehls auftreten. In letzterem Fall werden sie häufig als „Folgeereignis“ bezeichnet. **EndDeviceEvents** sind üblicherweise zu einem **UsagePoint** in Beziehung gesetzt.

4.4.5.7 Instandhaltungsanforderung von Zählern

Das DCIM-Modell liefert auch eine grundlegende Unterstützung für Datenaustausche im Zusammenhang mit Arbeitsmanagement in Bezug zu Endgeräten. Maßgebliche Klassen wurden in Bild 20 dargestellt.

MeterServiceWork ist eine Spezialisierung von **Work** (de: Arbeit), in welche ein **EndDevice** einbezogen ist.

Zum Beispiel kann es als Folge der Anmeldung neuer **Customer**, dem Löschen eines **Customer** oder dem Wechsel eines **Customer** von einem **ServiceSupplier** zu einem anderen erforderlich sein, **Meter** zu installieren, zu entfernen oder zu konfigurieren. Es kann ebenfalls der Bedarf zum Austausch eines **Meter** vorhanden sein, was das Entfernen des alten **Meter**, die Installation eines neuen **Meter** und die Konfiguration des neuen **Meter**, je nach Bedarf des Verbrauchszählungssystems, einschließt.

MeterServiceWork unterstützt auch die Sammlung von **MeterReadings**, welche im Zusammenhang mit den Instandhaltungsarbeiten ausgeführt wurden.

4.4.6 Zahlungszählung

4.4.6.1 Allgemeines

Zahlungszählung ist eine Erweiterung des Pakets **Metering** und enthält die Informationsklassen, welche spezialisierte Anwendungen wie Vorauszahlungszählung unterstützen. Diese Klassen stehen im Allgemeinen