Transformer modeling synopsis

UML

PowerTransformer

An electrical device consisting of two or more coupled windings, with or without a magnetic core, for introducing mutual coupling between electric circuits. Transformers can be used to control voltage and phase shift (active power flow).

A power transformer may be composed of separate transformer tanks that need not be identical.

A power transformer can be modelled with or without tanks and is intended for use in both balanced and unbalanced representations. A power transformer typically has two terminals, but may have one (grounding), three or more terminals.

The inherited association ConductingEquipment.BaseVoltage should not be used. The association from TransformerEnd to BaseVoltage should be used instead.

TransformerEnd.endNumber

Number for this transformer end, corresponding to the end's order in the power transformer vector group or phase angle clock number. Highest voltage winding should be 1. Each end within a power transformer should have a unique subsequent end number. Note the transformer end number need not match the terminal sequence number.

Power TransformerEnd

A PowerTransformerEnd is associated with each Terminal of a PowerTransformer.

The impedance values r, r0, x, and x0 of a PowerTransformerEnd represents a star equivalent as follows.

1) for a two Terminal PowerTransformer the high voltage (TransformerEnd.endNumber=1) PowerTransformerEnd has non zero values on r, r0, x, and x0 while the low voltage (TransformerEnd.endNumber=2) PowerTransformerEnd has zero values for r, r0, x, and x0. Parameters are always provided, even if the PowerTransformerEnds have the same rated voltage. In this case, the parameters are provided at the PowerTransformerEnd which has TransformerEnd.endNumber equal to 1.

2) for a three Terminal PowerTransformer the three PowerTransformerEnds represent a star equivalent with each leg in the star represented by r, r0, x, and x0 values.

3) For a three Terminal transformer each PowerTransformerEnd shall have g, g0, b and b0 values corresponding to the no load losses distributed on the three PowerTransformerEnds. The total no load loss shunt impedances may also be placed at one of the PowerTransformerEnds, preferably the end numbered 1, having the shunt values on end 1. This is the preferred way.

4) for a PowerTransformer with more than three Terminals the PowerTransformerEnd impedance values cannot be used. Instead use the TransformerMeshImpedance or split the transformer into multiple PowerTransformers.

Each PowerTransformerEnd must be contained by a PowerTransformer. Because a PowerTransformerEnd (or any other object) can not be contained by more than one parent, a PowerTransformerEnd can not have an association to an EquipmentContainer (Substation, VoltageLevel, etc).

4.5.6 Transformer model

Error! Reference source not found. shows a part of the Wires package in a class diagram which models a PowerTransformer device.

The transformer model is applicable for both balanced and unbalanced models. Additionally the PowerTransformer itself is now a ConductingEquipment with potentially multiple terminals to more directly model transformers with one, two, three, or more terminals.

As shown in Figure 1, the PowerTransformer is also able to optionally model tank details, which can be used to describe in detail the transformer internal winding phase connections and imbalances. In all cases a PowerTransformer models a group of physical devices acting together to transform power among terminals and in one physical location. For transmission systems, three physical single phase devices are often represented by one PowerTransformer instance. If detail of the individual single phase devices were required, the TransformerTank objects should additionally be modelled.

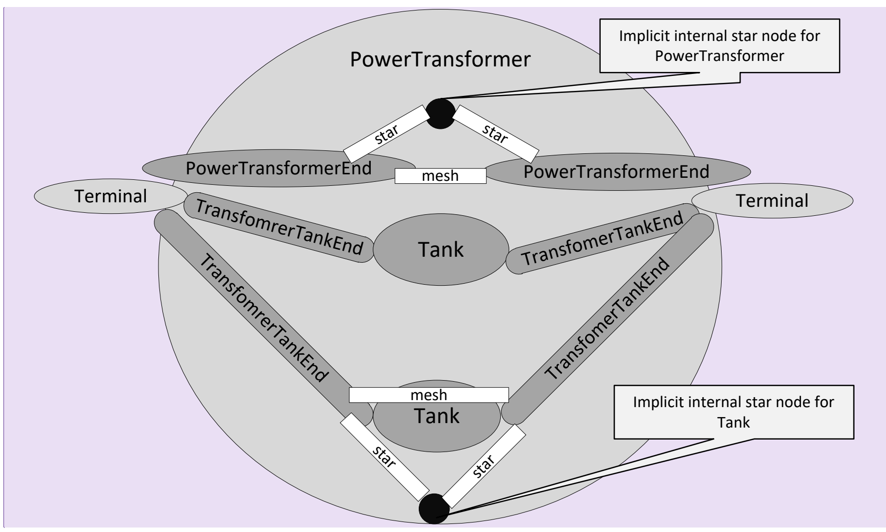


Figure 1 - Transformer and Tank model

Both a PowerTransformer and a TransformerTank can have impedance described in either an assumed star connection with implied centre connection or with mesh impedance form by using the optional TransformerMeshImpedance class. The mesh form of impedance is required to accurately model transformers with more than three terminals. If using the mesh impedance form, you would specify a TransformerMeshImpedance class for each possible terminal to terminal connection, thus one instance for two terminals, three for three terminals, and six for four terminals and so on. If using the star connection, the impedance parameters can be specified directly on the PowerTransformerEnd class, or alternatively shared among transformers using the TransformerStarImpedance class.

As shown in **Error! Reference source not found.**, a PowerTransformer is a specialized class of ConductingEquipment, which is a specialized class of Equipment. This is shown by the use of the generalization-type of relationship, which uses an arrow to point to the general class. The

Commented [DCI1]: CZ

Commented [TK2]: Resized figure to fit on one line inheritance permits the PowerTransformer to inherit attributes from ConductingEquipment, Equipment, and all generalizations of Equipment that are not shown in **Error! Reference source not found.**.

Commented [TK3]: How about starting the sub-clause with this paragraph?

A PowerTransformer also has relationships to PowerTransformerEnd and TransformerTank which are modelled with association type relationships. As shown, a PowerTransformer may associate zero or more PowerTransformerEnd-s, but a PowerTransformerEnd may associate to only zero or one PowerTransformer. Similarly, the PowerTransformer may associate to zero or more TransformerTank objects, but a TransformerTank may associate to only zero or one PowerTransformer object.

The PowerTransformerEnd specializes the TransformerEnd class which has other relationships as well:

- a generalization relationship with IdentifiedObject;
- two association relationships with the TransformerMeshImpedance class, such that a TransformerEnd object may be “From” 0, 1, or more TransformerMeshImpedance objects and “To” 0, 1, or more TransformerMeshImpedance objects;
- aggregation relationships with the PhaseTapChanger and RatioTapChanger classes, such that a TransformerWinding object may have 0 or 1 PhaseTapChanger objects and 0 or 1 RatioTapChanger objects associated with it.

For a transmission network PowerTransfomer that has two or three ends, the equivalent pi-model impedances appear in the PowerTransformerEnd r, r0, x, x0, g, g0, b and b0 attributes. The PowerTransformerEnd and its connected Terminal both have sequence numbers TransformerEnd.endNumber and ACDCTerminal.sequenceNumber. Those numbers may or may not be in synch.

Commented [DCI4]: CZ

4.4 Transformer modelling

A two winding PowerTransformer has two PowerTransformerEnds. This gives the option to specify the impedance values for the equivalent pi-model completely at one end or split them between the two ends. The impedances shall be specified at the primary voltage side as shown in Figure 2 where the left side is the “primary” (high voltage) voltage side.

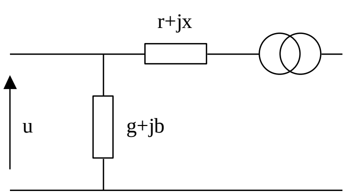


Figure 2 – Two winding transformer impedance

A three winding PowerTransformer has three PowerTransformerEnds. The equivalent pi-model corresponds to three ends connected in wye configuration as shown below. The impedance values for a three winding transformer are specified on each of the three TransformerWindings. Each of the ends has series impedances rn+jxn and shunt gn+jbn where n is: p for primary, s for secondary and t for tertiary as shown in Figure 3.

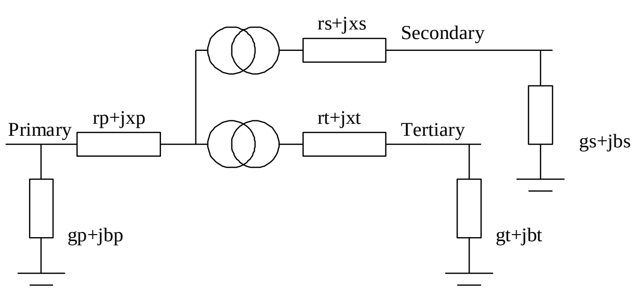


Figure 3 – Three winding transformer impedance

Additional requirements related to transformer modelling are listed below.

- Each PowerTransformer shall have at least two and no more than three PowerTransformerEnds.
- Each PowerTransformerEnd can have at most one tap changer (RatioTapChanger, PhaseTapChangerLinear, PhaseTapChangerSymmetrical, or PhaseTapChangerAsymmetrical). If a PowerTransformerEnd does not have an associated tap changer, the end should be considered to have a fixed tap.

Multiple types of regulating transformers are supported by the CIM model. Depending on the regulation capabilities, the effects of tap movement will be defined using the RatioTapChanger class, PhaseTapChangerLinear class, PhaseTapChangerSymmetrical class, or

PhaseTapChangerAsymmetrical class. Each of these classes are subtypes of the TapChanger class. The use of the various subtypes is explained in IEC 61970-301.

4.3 Requirements and Constraints

.
.
.

- C:452:EQ:PowerTransformerEnd:pu

  The parameters r, x, g and b are specified for each end and are not related to the overall base voltage. These values are specified in engineering units. Any PU calculations are internal to particular tools and are not part of the data exchange.

- C:452:EQ:PowerTransformerEnd.b:valueRange

  PowerTransformerEnd.b shall be negative value or zero. Negative magnetising branch susceptance (PowerTransformerEnd.b) means inductive reactive power losses in no load.

- C:452:EQ:PowerTransformerEnd.g:valueRange

  PowerTransformerEnd.g shall be positive value or zero. Positive magnetising branch conductance (PowerTransformerEnd.g) means positive active power losses in no load.

- C:452:EQ:PowerTransformerEnd.x:value

  Transformers with zero series reactance do not exist. PowerTransformerEnd.x of high voltage end in case of a two winding transformer shall be a positive value. In case of a three winding transformer the PowerTransformerEnd.x shall not be zero, but it can be a negative value.

- C:452:SC:PowerTransformerEnd.grounded:grounding

  If TransformerEnd.grounded is true, then TransformerEnd.rground and TransformerEnd.xground are required.

5.2.118 & 5.4.24 (Description) PowerTransformer

An electrical device consisting of two or more coupled windings, with or without a magnetic core, for introducing mutual coupling between electric circuits. Transformers can be used to control voltage and phase shift (active power flow).

A power transformer may be composed of separate transformer tanks that need not be identical.

A power transformer can be modelled with or without tanks and is intended for use in both balanced and unbalanced representations. A power transformer typically has two terminals, but may have one (grounding), three or more terminals.

The inherited association ConductingEquipment.BaseVoltage should not be used. The association from TransformerEnd to BaseVoltage should be used instead. [same as UML]

5.2.152 & 5.4.32 (abstract) TransformerEnd

A conducting connection point of a power transformer. It corresponds to a physical transformer winding terminal. In earlier CIM versions, the TransformerWinding class served a similar purpose, but this class is more flexible because it associates to terminal but is not a specialization of ConductingEquipment. [same as UML]

TransformerEnd.endNumber

Number for this transformer end, corresponding to the end's order in the power transformer vector group or phase angle clock number. Highest voltage winding should be 1. Each end within a power transformer should have a unique subsequent end number.   Note the transformer end number need not match the terminal sequence number. [same as UML]

5.4.25 & 5.2.119 PowerTransformerEnd

A PowerTransformerEnd is associated with each Terminal of a PowerTransformer.
The impedance values r, r0, x, and x0 of a PowerTransformerEnd represents a star equivalent as follows.
1) for a two Terminal PowerTransformer the high voltage (TransformerEnd.endNumber=1) PowerTransformerEnd has non zero values on r, r0, x, and x0 while the low voltage (TransformerEnd.endNumber=2) PowerTransformerEnd has zero values for r, r0, x, and x0. Parameters are always provided, even if the PowerTransformerEnds have the same rated voltage. In this case, the parameters are provided at the PowerTransformerEnd which has TransformerEnd.endNumber equal to 1.
2) for a three Terminal PowerTransformer the three PowerTransformerEnds represent a star equivalent with each leg in the star represented by r, r0, x, and x0 values.
3) For a three Terminal transformer each PowerTransformerEnd shall have g, g0, b and b0 values corresponding to the no load losses distributed on the three PowerTransformerEnds. The total no load loss shunt impedances may also be placed at one of the PowerTransformerEnds, preferably the end numbered 1, having the shunt values on end 1. This is the preferred way.
4) for a PowerTransformer with more than three Terminals the PowerTransformerEnd impedance values cannot be used. Instead use the TransformerMeshImpedance or split the transformer into multiple PowerTransformers.
Each PowerTransformerEnd must be contained by a PowerTransformer. Because a PowerTransformerEnd (or any other object) can not be contained by more than one parent, a PowerTransformerEnd can not have an association to an EquipmentContainer (Substation, VoltageLevel, etc). [same as UML]

PowerTransformerEnd.ratedS

Normal apparent power rating.
The attribute shall be a positive value. For a two-winding transformer the values for the high and low voltage sides shall be identical. [same as UML]

PowerTransformerEnd.ratedU

Rated voltage: phase-phase for three-phase windings, and either phase-phase or phase-neutral for single-phase windings.
A high voltage side, as given by TransformerEnd.endNumber, shall have a ratedU that is greater than or equal to ratedU for the lower voltage sides.
The attribute shall be a positive value. [same as UML]

Annex F (informative) Common power system model (CPSM) minimum data requirements

Assumptions

- Transformers with more than two windings are assumed to be modeled with an equivalent “star” model consisting of multiple 2 winding transformers connected to a fictitious “center” node.

61968-13

6.3.2.99 PowerTransformer

An electrical device consisting of two or more coupled windings, with or without a magnetic core, for introducing mutual coupling between electric circuits. Transformers can be used to control voltage and phase shift (active power flow).
A power transformer may be composed of separate transformer tanks that need not be identical. [same as UML]

The same power transformer can be modelled in two ways, namely with and without tanks:

1. The power transformer that uses power transformer ends directly (without tanks) is suitable for balanced three-phase models. This is typical for transmission and sub-transmission network modelling. Such a transformer will require one power transformer end for each physical winding. There must be a one-to-one association between PowerTransformerEnd and Core::Terminal.
2. The power transformer that uses transformer tanks is suitable for an unbalanced transformer, a balanced transformer within a single tank, or a balanced transformer made up of three tanks. This is typical for distribution network modelling and the only choice when modelling an unbalanced transformer, or a transformer that has more than three windings. Power transformer modelled with tanks will require for each tank, one transformer tank end per physical winding in the tank. There may be one, two, or three phases in the transformer tank end. Examples: 3 phases for 3-phase delta or wye connected windings, 2 for one phase-to-phase winding, and 1 for a phase-to-neutral or phase-to-ground winding. With 1 or 2 phases, more than one transformer tank end may be associated to the same 3-phase Core::Terminal instance, while with 3 phases there should be a one-to-one association.

This power transformer model is flexible in order to support different kinds of data exchange requirements. There are 5 possible ways to combine available classes and their attributes:

1. Instance parameters - Use the r, x, r0, x0, b, b0, g, and g0 attributes on PowerTransformerEnd and ignore related TransformerStarImpedance, TransformerMeshImpedance, or TransformerCoreAdmittance. This option assumes a star connection of the series impedances. It is suitable for typical transmission, balanced three-phase transformer models, for transformers with 2 or three windings.

2. Star instance parameters by association - Instead of the r, x, r0, x0, b, b0, g, and g0 attributes, use associations to TransformerStarImpedance and TransformerCoreAdmittance. This option is suitable in same scenarios as option 1, but when catalogue data is available for transformers.

3. Mesh instance parameters by association: Instead of the r, x, r0, x0, b, b0, g, and g0 attributes, use associations to TransformerMeshImpedance and TransformerCoreAdmittance. This option supports transformers with more than three windings.

4. Catalog mesh parameters by association - Instead of attributes r, x, r0, x0, b, b0, g, and g0 and associations to TransformerStarImpedance, TransformerMeshImpedance, or TransformerCoreAdmittance, use the association to TransformerEndInfo. The TransformerEnd.endNumber should match the corresponding TransformerEndInfo.endNumber, following the IEC standard convention of numbering from the highest voltage ends to the lowest, starting at 1. This matching supports higher-level use of a catalog, through just one association between TransformerTank and TransformerTankInfo, with simpler exchanges and incremental updates. The associated TransformerEndInfo will have associations to TransformerMeshImpedance and TransformerCoreAdmittance. This option supports unbalanced transformer, with more than three windings and is suitable whenever the transformer test data has been converted to an electrical model.

5. Catalog test data by association - This is the same as option 4, except TransformerEndInfo will have associations to AssetModels::TransformerTest decendents, instead of to TransformerMeshImpedance and TransformerCoreAdmittance. This option is suitable when the test data is available, and the receiving application is able to interpret the test data.

Every profile should specify which one or more of these options are supported.

PowerTransformer.vectorGroup

Vector group of the transformer for protective relaying, e.g., Dyn1. For unbalanced transformers, this may not be simply determined from the constituent winding connections and phase angle dispacements.
The vectorGroup string consists of the following components in the order listed: high voltage winding connection, mid voltage winding connection (for three winding transformers), phase displacement clock number from 0 to 11, low voltage winding connection, phase displacement clock number from 0 to 11. The winding connections are D (delta), Y (wye), YN (wye with neutral), Z (zigzag), ZN (zigzag with neutral), A (auto transformer). Upper case means the high voltage, lower case mid or low. The high voltage winding always has clock postion 0 and is not included in the vector group string. Some examples: YNy0 (two winding wye to wye with no phase displacement), YNd11 (two winding wye to delta with 330 degrees phase displacement), YNyn0d5 (three winding transformer wye with neutral high voltgage, wye with neutral mid voltgage and no phase displacement, delta low voltage with 150 degrees displacement).
Phase displacement is defined as the angular difference between the phasors representing the voltages between the neutral point (real or imaginary) and the corresponding terminals of two windings, a positive sequence voltage system being applied to the high-voltage terminals, following each other in alphabetical sequence if they are lettered, or in numerical sequence if they are numbered: the phasors are assumed to rotate in a counter-clockwise sense. [same as UML]

6.3.2.100 & 6.4.2.37 PowerTransformerEnd

A PowerTransformerEnd is associated with each Terminal of a PowerTransformer.
The impdedance values r, r0, x, and x0 of a PowerTransformerEnd represents a star equivalent as follows

1) for a two Terminal PowerTransformer the high voltage PowerTransformerEnd has non zero values on r, r0, x, and x0 while the low voltage PowerTransformerEnd has zero values for r, r0, x, and x0.
2) for a three Terminal PowerTransformer the three PowerTransformerEnds represents a star equivalent with each leg in the star represented by r, r0, x, and x0 values.
3) for a PowerTransformer with more than three Terminals the PowerTransformerEnd impedance values cannot be used. Instead use the TransformerMeshImpedance or split the transformer into multiple PowerTransformers. [matches UML, but UML has more]

6.4.2.54 & 6.3.2.139 (abstract) TransformerEnd 1890

TransformerEnd is a conducting connection point of a power transformer. It corresponds to a physical transformer winding terminal. In earlier CIM versions, the TransformerWinding class served a similar purpose. [same as UML] This successor TransformerEnd class is more flexible and has important differences with TransformerWinding

TransformerEnd.endNumber

Number for this transformer end, corresponding to the end's order in the PowerTransformer.vectorGroup attribute. Highest voltage winding should be 1. [matches UML, but UML has more]

6.9.2.25 PowerTransformerInfo 3226

Set of power transformer data, from an equipment library. [same as UML]

TransformerEndInfo.endNumber

Number for this transformer end, corresponding to the end's order in the PowerTransformer.vectorGroup attribute. Highest voltage winding should be 1. [same as UML]

TransformerEndInfo.phaseAngleClock

Winding phase angle where 360 degrees are represented with clock hours, so the valid values are {0, ..., 11}. For example, to express winding code 'Dyn11', set attributes as follows: 'connectionKind' = Yn and 'phaseAngleClock' = 11. [same as UML]

Balanced Transmission Transformer Class Model

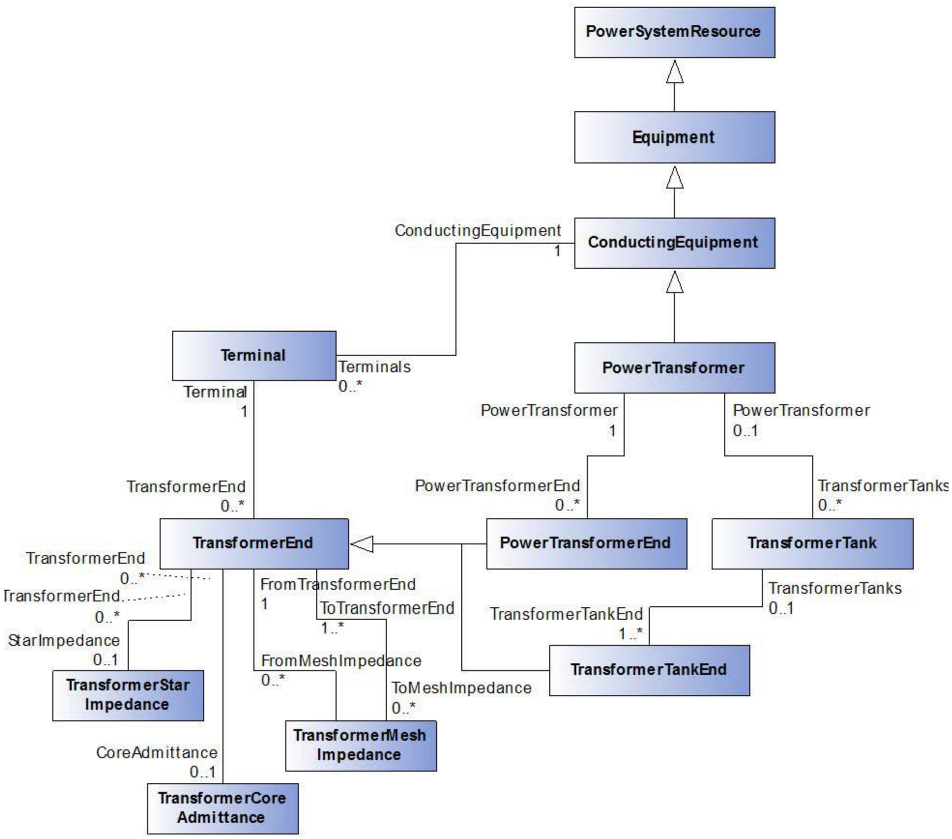


Figure 5-15 Transformer Class Diagram CIM15+

The complete class diagram for the new transformer model is shown in Figure 5-15 and even with the *Tap Changer* model omitted it is significantly more complex than Figure 5-13. This does not mean that representing the same transformer in CIM v14 and CIM v15 requires a corresponding increase in complexity. The diagram reflects the two different potential routes to model transformers depending on whether they are for a balanced transmission system or represent multi-phase transformers (that is, pole-mounted distribution transformers) and when modeling a balanced transmission level power transformer the changes are minor.

21 The concepts of balanced/unbalanced, catalog/de-normalized, and impedance representation as star or mesh form are in theory completely orthogonal in the CIM information model. In practice the balanced and de-normalized star impedance are used together for transmission exchanges and the unbalanced and catalog features are used in distribution model exchanges.

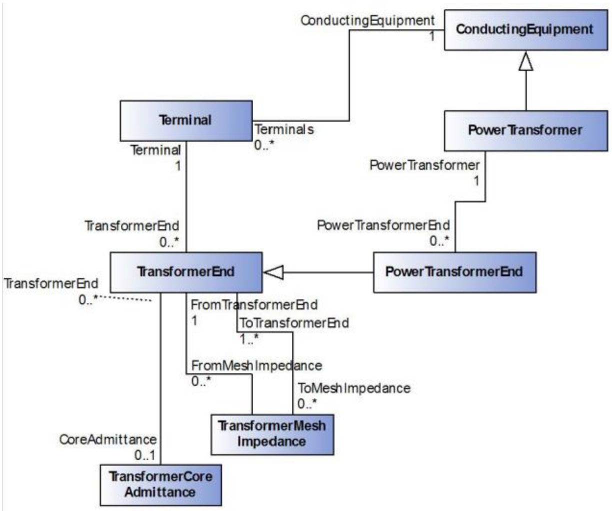


Figure 5-16 Balanced Power Transformer Classes

Figure 5-16 shows the classes required to represent a balanced transmission level power transformer in CIM v15. This includes the new *TransformerMeshImpedance* and *TransformerCoreAdmittance* classes, which could be considered optional as the legacy explicit attributes for impedance and admittance are still included in *TransformerEnd*.

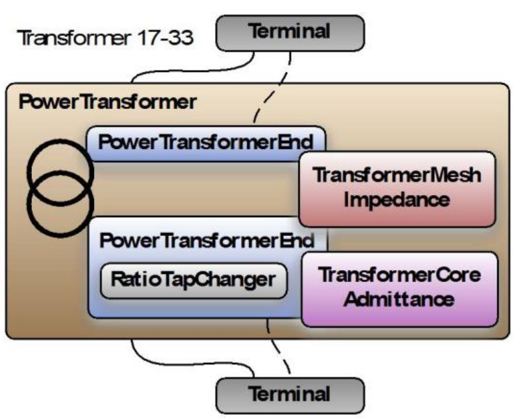


Figure 5-17 CIM v15 Transformer Instance Example

The corresponding instance example for CIM v15 is shown in Figure 5-17 where a *PowerTransformer*, two *PowerTransformerEnds*, a *TransformerMeshImpedance*, and a *TransformerCoreAdmittance* are modeled. In this diagram a *RatioTapChanger* has also been included, although this could also be a

PhaseTapChanger, in CIM v15 they are explicitly defined as separate associations and there are multiple subclasses of *PhaseTapChanger* which are described later in this section.
The *Terminals* in this representation have an association to the *PowerTransformer* since it now inherits from *ConductingEquipment*, but there is also a direct association between *TransformerEnd* and *Terminal* (shown as the dotted line in the diagram).
At first glance the new transformer model appears far more complex than was present before CIM v15. However, the reader should see that representing the same data for a balanced transmission transformer requires only minor refactoring of the data elements and the inclusion of some new classes for more explicitly modeling impedance and admittance. These new classes for impedance and admittance are currently optional as the attributes for representing the impedance and admittance are still present in the new *TransformerEnd* class (however these are not used in the example above as they should be considered legacy or deprecated).

Unbalanced Distribution Transformer Model with Tanks

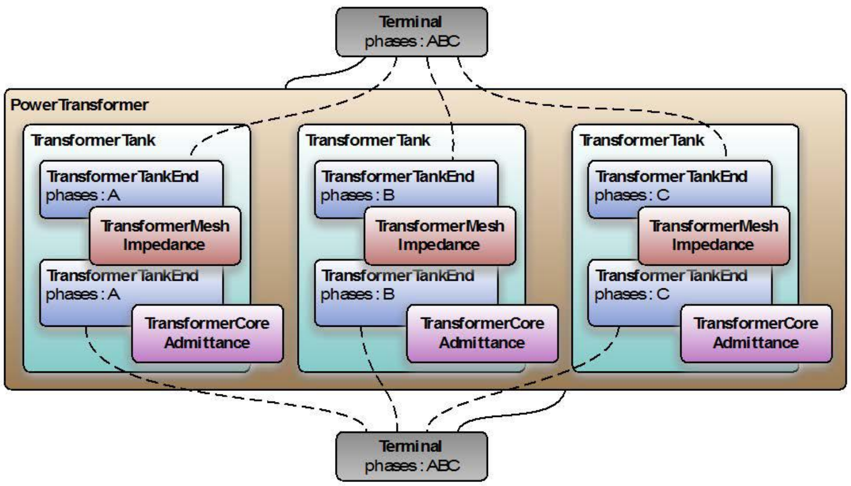


Figure 5-18 CIM v15 Transformer with multiple tanks instance example

When defining an unbalanced distribution transformer, the *PowerTranformer* will contain one or more *tanks*, each with its own phase and two or more *TransformerTankEnds*. An instance example of this is shown in Figure 5-19 with a single *PowerTransformer* with two *Terminals* that is part of a three-phase system, but with three *TransformerTank* instances, each containing two *TransformerTankEnds* of a single phase. Both the *Terminals* have a *phase code* of ABC (They could also be ABCN to denote that the neutral is included) and a *ConductingEquipment* association to the *PowerTransformer*, but each *Terminal* also has three additional associations to the individual *TransformerTanksEnds* on each side of the transformer.
The *TransformerTankEnds* and *PowerTransformerEnds* from the previous example share references to a common *Terminal* instance but specialize the abstract *TranformerEnd* class and thus reuse the same representation of impedance schema. This means that the two models share common representations for the key electrical properties, the major differences being the inclusion of multiple *tanks* within a single *PowerTransformer* instance when modeling the distribution network; and the *TransformerTankEnd* specifies the phases of the tank, which must be a subset of the phases of the associated *Terminal*.
Thus, the *Terminal* represents the external connection of the *PowerTransformer* and the *TransformerTank* and *TransformerTankEnd* represent the internal model of the transformer and how it connects to the external terminals.

<u>Tom's Physical_Device_Subgroup_Reviewed_document</u>

1 Transformer and Regulator Examples

Most of these examples were developed and documented earlier [4, 10], during work on *TransformerTankInfo* and transformer test sheets. As is the case with lines and cables, the CIM necessarily provides different options to model transformers, based on whether a catalog will be used, and whether the transformer is unbalanced. Figure 4 shows UML for an accurate power flow and short circuit model of multi-winding balanced transformers (other models are possible in the CIM). There is currently no catalog support for this model.

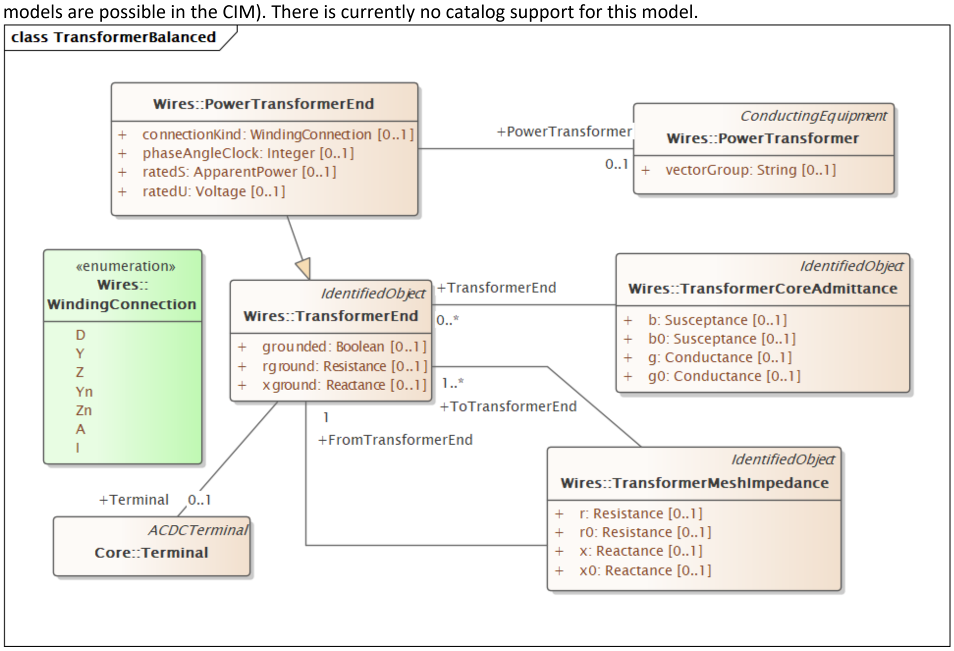


Figure 4: A profile for balanced transformer analysis using mesh impedances

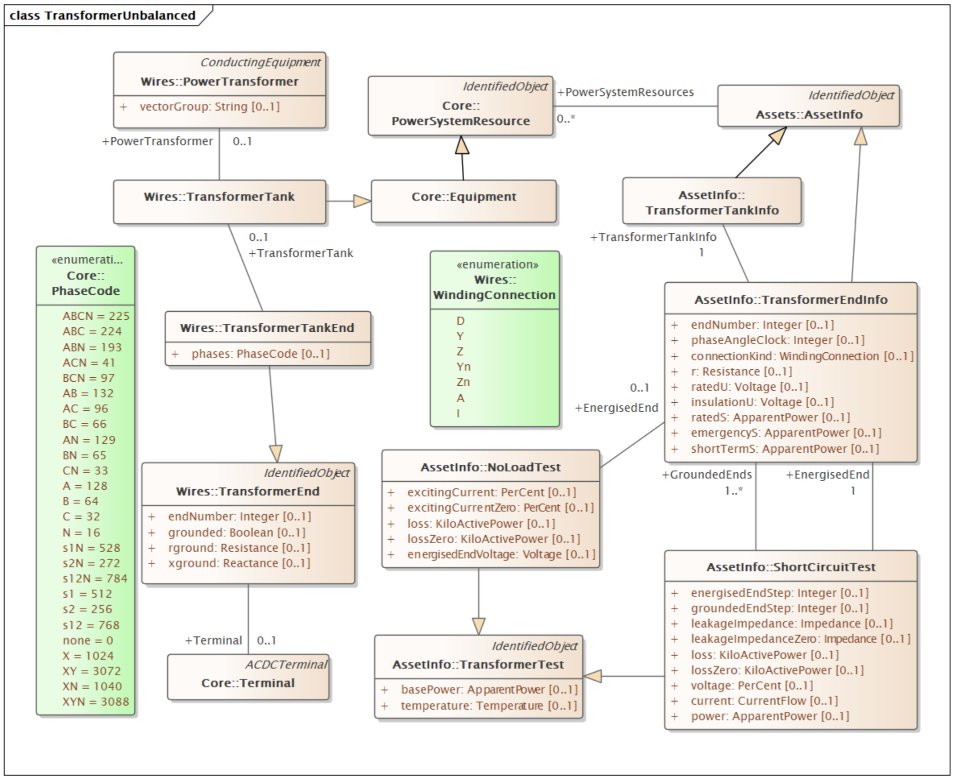


Figure 5: A profile for unbalanced transformer analysis using catalog of transformer tanks

Figure 5 shows UML for an accurate power flow and short circuit model of unbalanced transformers, composed of one or more TransformerTank instances. These use rating and test data maintained in a catalog. If the *DataSheetInfo* proposal is adopted, this UML could be simplified with a direct association between *TransformerTank* and *TransformerTankInfo*. There is an *OpenCircuitTest* not shown, because it serves to confirm rating attributes already included in other CIM classes. There is also a *PowerTransformerInfo* class not shown, because the associations to *TransformerTankInfo* are more direct and because an aggregation of tanks, i.e., the *PowerTransformer*, may not be persistent in the catalog.

1.1 Open Wye/Delta Transformer

The transformer in Figure 6 is an open wye / open delta bank, which is used to supply inexpensive three-phase service to smaller customers. The “lighting leg” (Tank A) usually has a different rating than the “power leg” (Tank B). Table 1 shows the connection and phasing information for each *TransformerTank*, which comprise one unbalanced *PowerTransformer*.

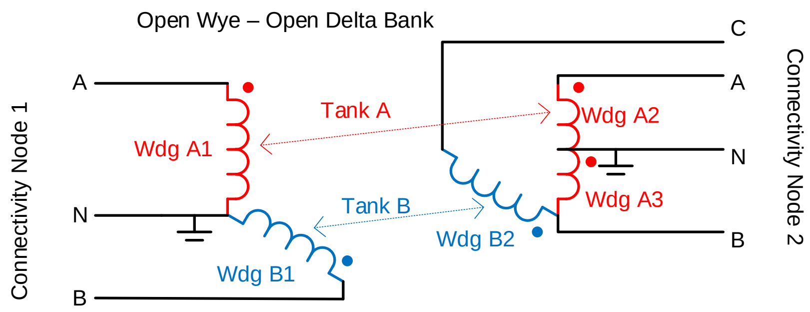


Figure 6: Money-saving open wye/delta transformer bank

Table 1: Open wye/delta TransformerTank and TransformerTankEnd connections

Tank	End	ratedU	ratedS	connectionKind	phaseAngleClock	Phases
A	A1	7200	100e3	I	0	AN
	A2	120	50e3	I	0	AN
	A3	120	50e3	I	6	BN
B	B1	7200	50e3	I	0	BN
	B2	240	50e3	I	0	BC

1.2 Autotransformer

An autotransformer is made by connecting two transformer windings in series, so there is a metallic connection between the two voltage levels. The main advantage is a cost savings, because the MVA rating is higher than for the same two windings connected as a conventional transformer. The main disadvantage is probably higher short-circuit levels, because autotransformers have lower impedance. Two common applications are:

1. Transformations between two extra-high voltage (EHV) levels in a substation, where the cost savings are important.
2. Line voltage regulators on distribution feeders, where the regulating (buck/boost) winding is connected in auto.

Figure 7 shows a two-winding transformer to the left, with a short-circuit test connection on the L winding terminal. All current is transformed magnetically, and the turns ratio is n1:n2. To the right, Figure 7 shows the same two windings connected as an autotransformer. The red current flows directly to the short-circuit test connection on the L winding terminal. In addition, the magnetically transformed blue current also contributes to the short-circuit current, which is higher than in the two-winding case. The autotransformer turns ratio is (n1+n2):n2.

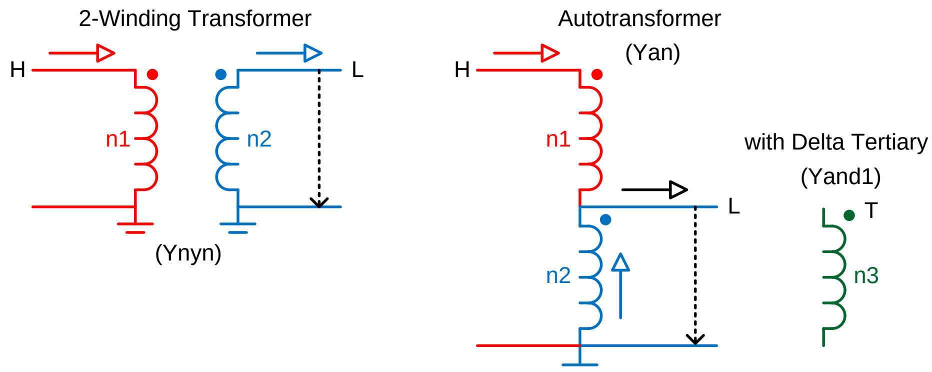
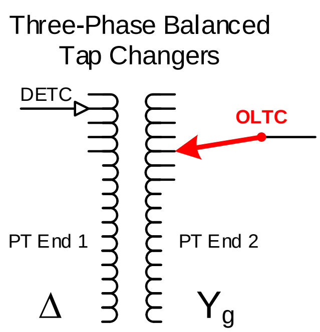


Figure 7: Two-winding transformer connected as an autotransformer

1.3 Ratio Tap Changers

Figure 8 (left) shows a two-winding substation transformer, nominally 138 kV delta / 12.47 kV wye grounded, with two tapped windings. On the high-voltage side, a no-load or de-energized tap changer (DETC) is often provided to permanently adapt the transformer to nominal system voltage. The DETC taps are usually +/- 5% in steps of 2.5%. Suppose the rated voltage as manufactured was 135 kV. Setting the DETC +1 tap would emulate a rating of 138.375 kV, which is closer to the system nominal voltage. The DETC would be set during installation and not changed. There is a minor effect on impedance characteristics when adjusting the DETC, but many analysts ignore that. The DETC setting may be accounted for by adjusting rated voltage to 138.375 kV, by just assuming rated voltage matches the system nominal of 138 kV, or by using the actual rated voltage of 135 kV plus the DETC. On the low-voltage side, an on-load tap changer (OLTC) is provided for dynamic adjustments. The OLTC typically has smaller steps and covers a broader range, say +/- 10% in steps of 1%. It could change taps many times per day. Normally, all three phases of the OLTC move together, either by direct mechanical connection to the same motor drive, i.e., they are "ganged", or by electronic control that still moves each phase tap together. If each phase tap moves independently, that would create a phase unbalance that's not represented anywhere else in the balanced CPSM (apart from ground faults).
Figure 8 (right) shows a three-phase distribution regulator in the same cabinet, in which each phase tap position can vary independently to correct voltage unbalance. This device comprises three autotransformers and three tap changers in the same cabinet. It can't be modeled in CIM without changing the semantics of what a "transformer tank" means.



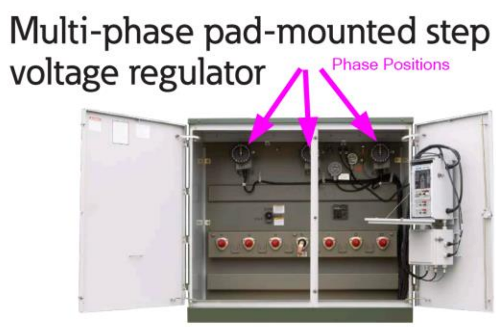


Figure 8: Balanced Three-phase Tap Changers Adjust Each Phase Identically on a CIM PowerTransformer (left); however, independent phase regulators may also appear in the same box (right, from Eaton brochure).

Figure 9 (left) shows the schematic representation of three independent voltage regulators that can correct voltage unbalances. These may appear out on the line as in Figure 9 (center), where the three phase regulators are mounted on adjacent poles. Single-phase regulators also appear out on the feeder, or two single-phase regulators connected phase-to-phase (near each other) in the “open delta” configuration. Figure 9 (right) shows two feeder regulators (one circled in magenta) in the substation. Because each feeder may experience different loading and unbalance levels, it’s preferable to provide a regulator for each one rather than an OLTC on the substation transformer.

Figure 9: Independent Single-phase Tap Changers apply to CIM TransformerTanks on a Feeder (center) or in a Substation (right). Three-phase examples are shown here, but single-phase and two-phase open-delta configurations also appear on real feeders.

Figure 10 depicts the CIM classes that represent voltage (magnitude) regulating transformers. To the left, *PowerTransformerEnd* would associate off-page with *PowerTransformer* for a balanced model. Alternatively, *TransformerTankEnd* would associate off-page with *TransformerTank* for a possibly unbalanced model, and in turn the *TransformerTank* would associate off-page with a *PowerTransformer*. Only *TransformerTankEnd* provides access to individual phases, so it’s the only option for Figure 9, or Figure 8 (right). Please see the Appendix for some notes and suggestions on the UML class documentation. Most importantly, there is a possible redundancy or inconsistency between *Terminal.phases* and other *phases* attributes in the “Wires Phase Model”, including *TransformerTankEnd.phases* in Figure 10. A consistent practice would be to use the “Wires Phase Model” and not the *Terminal.phases* attribute, with no ill effects to unbalanced power flow analysis.
Figure 11 is the object diagram for the balanced transformer in Figure 8 (left). Associations are one-way, in direction of the arrows. Only a few selected attributes are specified to illustrate some points:

- The *endNumber* begins at 1, for the winding with highest voltage rating
- The DETC has a *normalStep* of 1 and it won’t be controlled. From information given earlier, one could infer that *neutralStep* = 0, *lowStep* = -2, *highStep* = 2, *stepVoltageIncrement* = 2.5% and <u>*neutralU*</u> = 135000. *step* = 1 and it won’t change. Again, we could have ignored the DETC completely and simply adjusted *ratedU* on the *hv1:PowerTransformerEnd*.
- The OLTC might have settings of *neutralStep* = *step* = 0, lowStep = -10, *highStep* = 10, *stepVoltageIncrement* = 1.0% and <u>*neutralU*</u> = 12470 (or 7200). The balanced OLTC senses voltage on just one phase, i.e., *monitoredPhase* = A in this example, but any phase voltage would be the same in a balanced model.
- The units of *targetValue* and *targetDeadband* are documented to be *Voltage* because *mode* = *voltage*. So, if not representing the potential transformer (PT)[1], one would choose *neutralU* = 12470, *targetValue* = 12470 and *targetDeadband* = 208.
- If we’d rather model the actual device settings, then *targetValue* = 120 and *targetDeadband* = 2, because *ptRatio* is going to be 60 in this case. That leads to *neutralU* = 7200 for a wye-connected tap changer. To

[1] Voltage transformer (VT) is now the preferred term.

represent the *ptRatio*, we must create a *TapChangerInfo* and navigate there through an *Asset* instance. Instead, a direct link to some *DataSheetInfo* subclass might be considered.

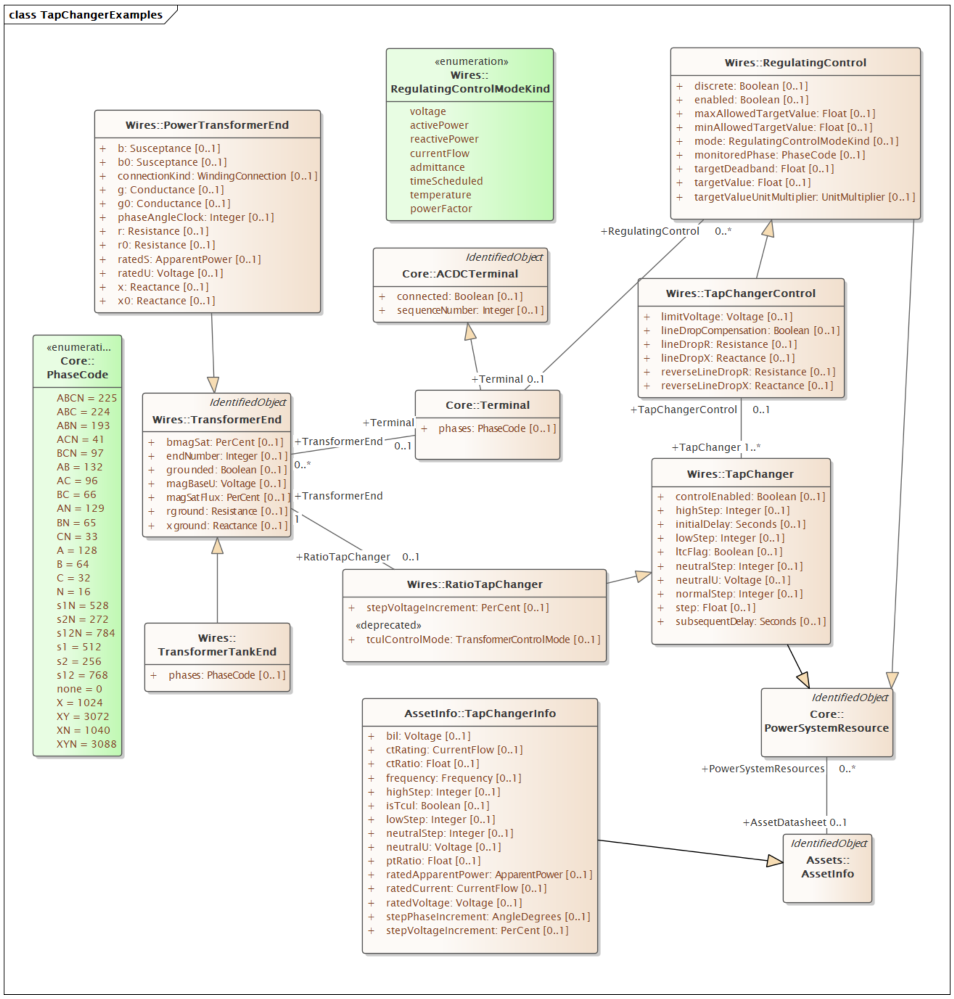


Figure 10: UML for voltage regulators in CIM17v38_CIM13v13_CIM03v17a

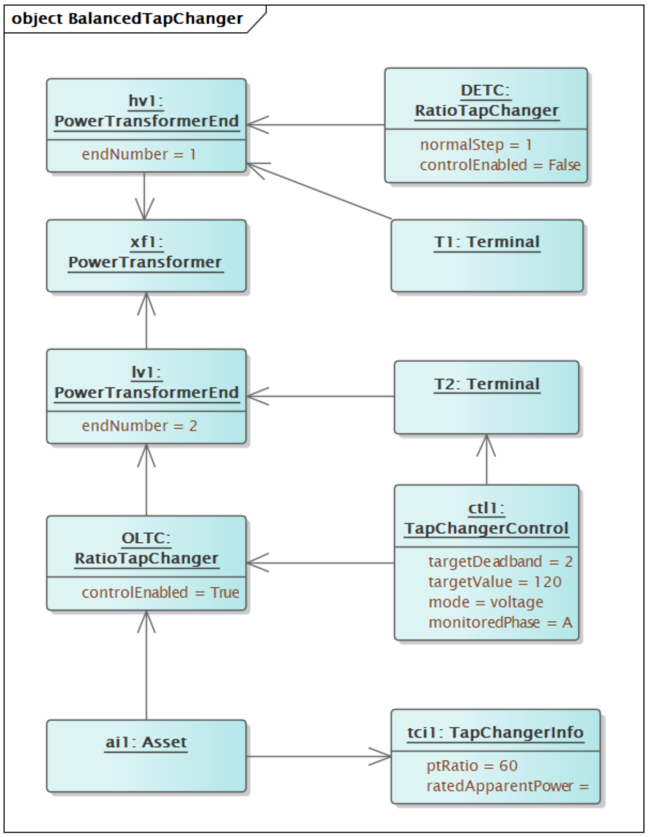


Figure 11: Balanced Tap Changer Object Diagram

Figure 12 represents the object diagram for unbalanced regulators in Figure 9. It looks more complicated, but for the most part that's because we duplicate the same four-object pattern of end, terminal, control and tap changer three times, once per phase. Off-page, the three *TransformerTank* instances would associate with three *TransformerTankEnd* instances and then three *Terminal* instances for *endNumber* = 1. To implement line-drop compensation (LDC) we use *lineDropR* and *lineDropX* attributes, which in turn require the current transformer (CT) information found <u>only</u> in *TapChangerInfo*. In this case the LDC attribute values are the same in each phase, but that's not always the case, especially in the "open delta" configuration of two regulators connected phase-to-phase. From earlier discussion, the various *targetValue*, *targetDeadband* and *step*-related attribute values would also be specified.

The unbalanced regulator in Figure 8 (right) would have to be modeled as in Figure 12, despite it having one tank and not three tanks. The semantics don't impede power flow analysis, but they could be an issue for asset management. There is a *PowerTransformerInfo* class that collects *TransformerTankInfo*, which might help with asset management.

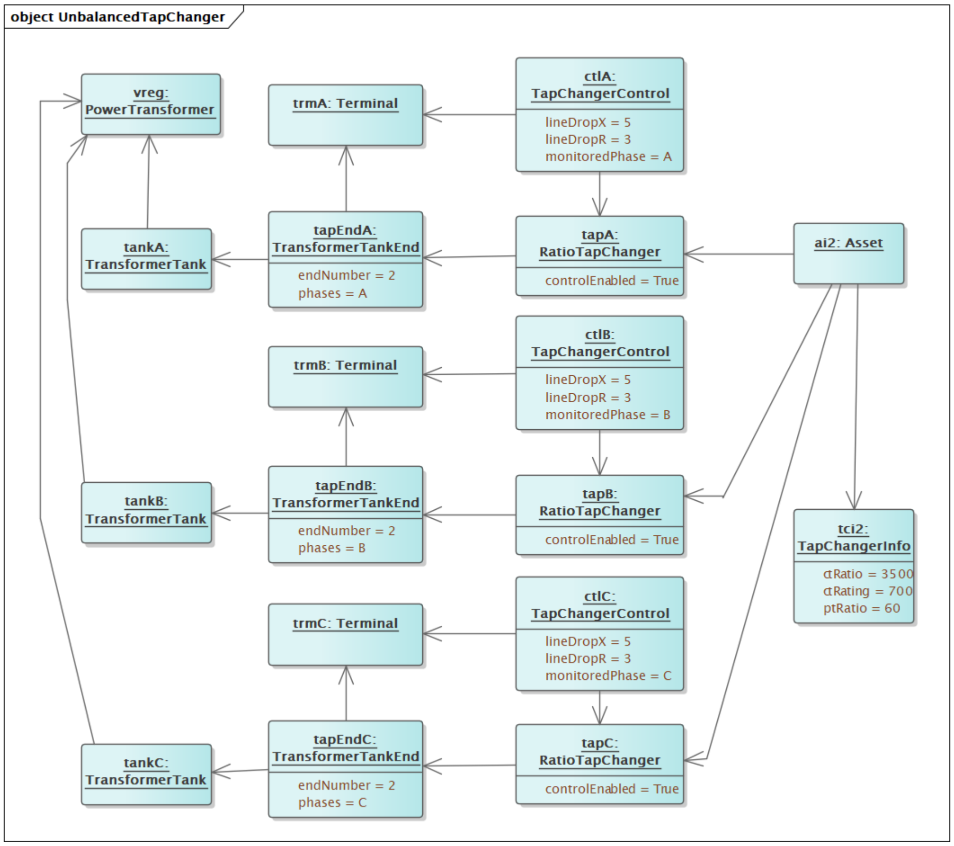


Figure 12: Unbalanced Regulator Object Diagram