

Meta-modelo CVD

1 Introducción

Este meta-modelo constituye una propuesta de meta-modelo para los modelos mediante los cuales formular el resultado de validar otros modelos y aspira a ser capaz de cubrir todo el espectro de violaciones de restricciones que puedan darse en modelos MDSE, permitiendo modelar con mayor o menor detalle los datos necesarios para describir tales incumplimientos. Para ello, presenta una jerarquía de clases que posibilita dicho modelado, incremental en cuanto a detalle, de manera tanto más detallada cuanto más se profundiza en la jerarquía.

2 Clases principales

El meta-modelo CVD presenta una estructura convencional, con una clase contenedor principal (CVD_Model) y una clase raíz (CVD) de la que heredan el resto de clases, conformando en conjunto la anteriormente mencionada jerarquía de clases destinadas al modelado de (los datos necesarios para describir) incumplimientos de restricciones.

La Fig. 2.1 muestra las clases contenedor principal y raíz, así como las principales subclases de ésta última.

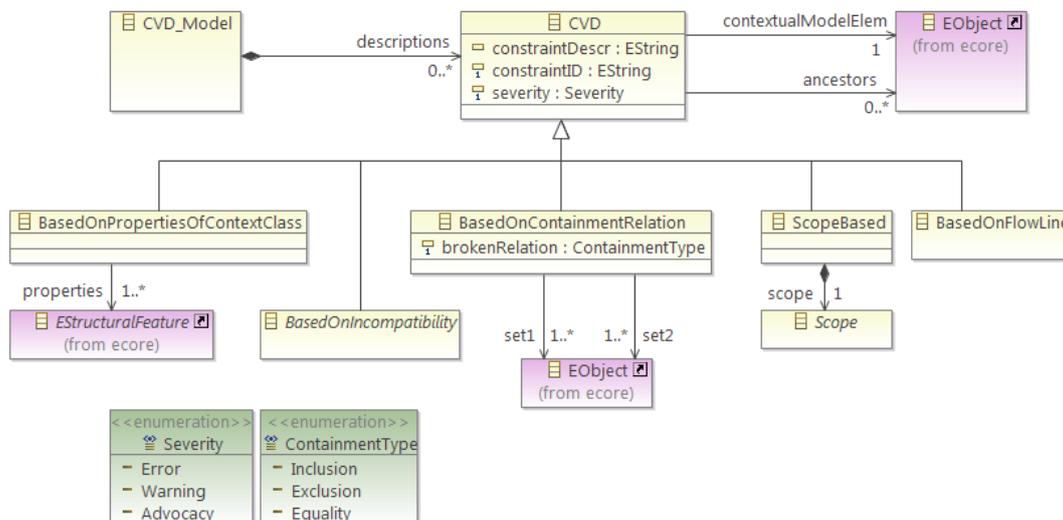


Figura 2.1 – Núcleo del meta-modelo CVD

Un modelo conforme a CVD posee una única instancia de CVD_Model, la cual contiene a través de la asociación-composición `descriptions` al resto de elementos de modelo, instancias de CVD o de cualquiera de sus subclases.

2.1 Clase CVD_Model (contenedor principal)

El meta-modelo CVD define una clase contenedor principal llamada CVD_Model cuya única instancia en un modelo conforme tiene por finalidad albergar, a través de la asociación-composición `descriptions`, el resto de elementos del modelo, instancias que contienen los datos base para describir el conjunto de incumplimientos detectado en un modelo respecto a

las restricciones especificadas para su meta-modelo. Estos elementos de modelo son instancias del resto de clases del meta-modelo, bien de la clase raíz `CVD` o de cualquiera de sus subclases. Así, el resultado de validar un modelo que cumpla todas las restricciones impuestas sobre su meta-modelo sería un modelo `CVD` vacío en cuanto a instancias de descripción de incumplimientos.

2.2 Clase CVD (clase raíz)

- **Naturaleza de las restricciones a ser descrito su incumplimiento.**

Esta clase permite modelar de forma genérica la descripción de incumplimientos de restricciones de cualquier tipo.

Únicamente con esta clase raíz sería suficiente para formular como modelo el conjunto de incumplimientos de restricciones detectados en un modelo, pues la información que describe es apta para cualquier restricción, independientemente de su naturaleza, semántica o formulación OCL.

- **Estructura.**

Esta clase define únicamente el identificador de la restricción (`constraintID`) junto a una descripción textual opcional de ella (`constraintDescr`) así como la severidad (`severity`) asociada al problema representado por el incumplimiento en cuestión. También se referencia el elemento de modelo en el cual se ha localizado la violación (`contextualModelElem`) junto a aquellos otros que constituyen la ruta desde él hacia el contenedor principal del modelo (`ancestors`).

Figura 2.2

- **Relación con el meta-modelo CC.**

Esta clase puede ser referenciada por instancias de cualquier clase en el meta-modelo CC.

- **Manifestación de incumplimiento.**

Con esta información descriptiva sería posible, por ejemplo, manifestar un incumplimiento construyendo un mensaje según el patrón (los *tokens* en negrita corresponden a datos modelados):

```
severity:  
The constraintID constraint has been violated at contextualModelElem.ID : contextualModelElem.Type  
- Location: mainContainer.ID >> ... ancestors[i].ID ... >> contextualModelElem.ID  
- Constraint explanation: constraintDescr
```

MAST 2:

Todas las restricciones especificadas para el meta-modelo MAST 2 pueden mapearse a esta clase CVD, pues mediante ella pueden modelarse (los datos necesarios para describir) de manera genérica los incumplimientos de cualquiera de ellas, de tal forma que en un modelo MAST 2, el incumplimiento por parte de un elemento de modelo de cualquiera de tales restricciones podría manifestarse mediante mensajes contruídos según el patrón anterior.

Por ejemplo, para la restricción ~~1-2-a)~~

Error:

The **speedFactorPositive** constraint has been violated at **Proc_1** : **Regular_Processor**.

- Location: Teleoperated_Robot >> Proc_1
 - Constraint explanation: **the speedFactorPositive constraint is intended to ...**
-

Sin embargo, en el meta-modelo CVD se define toda una jerarquía de subclases de la clase raíz `CVD` con el objetivo de posibilitar el modelado de descripciones más completas acerca de incumplimientos de restricciones. Así, en lugar de formular mediante ella las violaciones detectadas, será posible asociar a cada restricción especificada sobre un meta-modelo la subclase de `CVD` que sea más adecuada para describir con mayor nivel de detalle los incumplimientos de ella producidos. Estas subclases se presentan en la siguiente sección.

3 Subclases de CVD

Figura 3.1

3.1 BasedOnPropertiesOfClass

- **Naturaleza de las restricciones a ser descrito su incumplimiento.**

Esta clase es adecuada exclusivamente para modelar la descripción de incumplimientos de restricciones especificadas sobre un conjunto de propiedades (atributos y referencias) de la clase contexto.

- **Estructura.**

Junto a las posibilidades de modelado heredadas de la clase CVD, esta clase define una asociación `properties` mediante la que especificar aquellas propiedades involucradas en la restricción.

Figura 3.2

- **Relación con el meta-modelo CC.**

Lo natural es que esta clase sea referenciada por instancias de la clase del mismo nombre en el meta-modelo CC o de subclases suyas.

- **Manifestación de incumplimiento.**

Con esta información descriptiva sería posible, por ejemplo, manifestar un incumplimiento construyendo un mensaje según el patrón (los *tokens* en negrita corresponden a datos modelados):

severity:

The **constraintID** constraint has been violated at **contextualModelElem.ID** : **contextualModelElem.Type** by properties **properties[0].ID**, **properties[1].ID**, ... and **properties[N].ID**

- Location: mainContainer.ID >> ... **ancestor[i].ID** ... >> **contextualModelElem.ID**

- Constraint explanation: **constraintDescr**

MAST 2:

De entre las restricciones especificadas sobre el meta-modelo MAST2, todas las del primer bloque pueden mapearse a esta clase BasedOnPropertiesOfClass, pues mediante ella pueden modelarse (los datos necesarios para describir) de manera detallada los incumplimientos de cualquiera de ellas, de tal forma que en un modelo MAST 2, el incumplimiento por parte de un elemento de modelo de cualquiera de tales restricciones podría manifestarse mediante mensajes contruídos según el patrón anterior.

*Por ejemplo, para las restricciones **1.2.a)** y **1.3.a)***

Error:

The **speedFactorPositive** constraint has been violated at **Proc_1 : Regular_Processor** by property **Speed_Factor**.

- Location: Teleoperated_Robot >> Proc_1

- Constraint explanation: **the speedFactorPositive constraint is intended to ...**

Error:

The **maxInterrPrioGreaterThanMin** constraint has been violated at **Proc_1 : Regular_Processor** by properties **Max_Interrupt_Priority** and **Min_Interrupt_Priority**.

- Location: Teleoperated_Robot >> Proc_1

-
- Constraint explanation: **the maxInterrPrioGreaterThanMin constraint is intended to**
...
-

3.2 BasedOnIncompatibility

- **Naturaleza de las restricciones a ser descrito su incumplimiento.**

Esta clase es adecuada exclusivamente para modelar la descripción de incumplimientos de restricciones consistentes en establecer incompatibilidades entre subclases de dos clases (típicamente abstractas) del meta-modelo de dominio que se hallan relacionadas con la clase contexto a través de sendas cadenas de asociaciones.

Se puede dar el caso particular de que una de las clases coincida con la propia clase contexto, en cuyo caso la cadena de asociaciones correspondiente sería nula.

- **Estructura.**

`BasedOnIncompatibility` es una clase abstracta que, salvo aportar semántica, no aumenta las posibilidades de modelado heredadas de la clase `CVD`. Ello se consigue a través de sus subclases, detalladas posteriormente.

Figura 3.3

- **Relación con el meta-modelo CC.**

Al ser abstracta, esta clase no ha de ser referenciada por instancias de clases del meta-modelo CC.

- **Manifestación de incumplimiento.**

3.3 BasedOnContainmentRelation

- **Naturaleza de las restricciones a ser descrito su incumplimiento.**

Esta clase es adecuada exclusivamente para modelar la descripción de incumplimientos de restricciones consistentes en establecer una relación de contención entre dos conjuntos de instancias de una clase.

Cada conjunto viene indicado por el extremo final de sendas cadenas de asociaciones que parten de la clase contexto y la enlazan con la clase que tipa a los elementos de los conjuntos.

- **Estructura.**

Junto a las posibilidades de modelado heredadas de la clase CVD, esta clase define dos asociaciones `set1` y `set2` mediante las que especificar aquellas cadenas de referencias cuyo extremo final indica los conjuntos de elementos de modelo sobre los que se establece la relación de contención.

Figura 3.4

- **Relación con el meta-modelo CC.**

Esta clase sólo puede ser referenciada por instancias de la clase del mismo nombre en el meta-modelo CC.

- **Manifestación de incumplimiento.**

Con esta información descriptiva sería posible, por ejemplo, manifestar un incumplimiento construyendo un mensaje según el patrón (los *tokens* en negrita corresponden a datos modelados):

severity:

The **constraintID** constraint has been violated at **contextualModelElem.ID** : **contextualModelElem.Type** by sets of model elements

{set1[0].ID, set1[1].ID...} : **set1[0].Type**

and

{set2[0].ID, set2[1].ID...} : **set1[0].Type**

because the 2nd set is not **included in / excluded from / equal to** the 1st one.

- Location: mainContainer.ID >> ... **ancestor[i].ID** ... >> **contextualModelElem.ID**

- Constraint explanation: **constraintDescr**

MAST 2:

De entre las restricciones especificadas sobre el meta-modelo MAST2, la restricción ~~X.X.X~~ es un ejemplo de restricción que puede mapearse a esta clase BasedOnContainmentRelation, pues mediante ella pueden modelarse (los datos necesarios para describir) de manera detallada sus incumplimientos, de tal forma que en un modelo MAST 2, su incumplimiento por parte de un procesador podría manifestarse mediante un mensaje construido según el patrón anterior.

Error:

The **timerListIncludesSystemTimer** constraint has been violated at **Proc_1** : **Regular_Processor** by sets of model elements

{Timer_1, Timer_2} : **Ticker**

and

{Timer_0} : Ticker

because the 2nd is not **included in** the 1st one.

- Location: Teleoperated_Robot >> Proc_1
 - Constraint explanation: **the timerListIncludesSystemTimer constraint is intended to**
...
-

3.4 ScopeBased

- **Naturaleza de las restricciones a ser descrito su incumplimiento.**

Esta clase es adecuada exclusivamente para modelar la descripción de incumplimientos de restricciones cuya satisfacción depende no sólo del estado de un elemento de modelo, sino también del ámbito o población de elementos de modelo dentro del cual el primero se halla.

- **Estructura.**

Junto a las posibilidades de modelado heredadas de la clase CVD, esta clase define una asociación-composición `scope` que contiene la descripción del ámbito o población de elementos de modelo en cuyo seno se **incumple** la restricción.

Figura 3.5

En el caso más general, el ámbito puede ser el modelo entero, lo cual se representa mediante la clase `GlobalScope`, mientras que la clase `BasicScope` sirve para modelar ámbitos más específicos, esto es, poblaciones de elementos de modelo alojados en el extremo de una asociación con cardinalidad múltiple. Esto se consigue indicando un elemento de modelo (`scopeModelElem`) y una referencia de cardinalidad múltiple (`scopeRef`) perteneciente a la clase del `scopeModelElem`. Si el tipo del extremo viene definido por una clase que posee subclases, es posible restringir la población a aquellos elementos que sean exclusivamente instancias de una o más de esas subclases, lo cual se consigue especificando a través de la referencia `subClasses` qué subclases de filtrado se aplican. Por último, la clase `CombinedScope` representa la unión de dos o más ámbitos básicos.

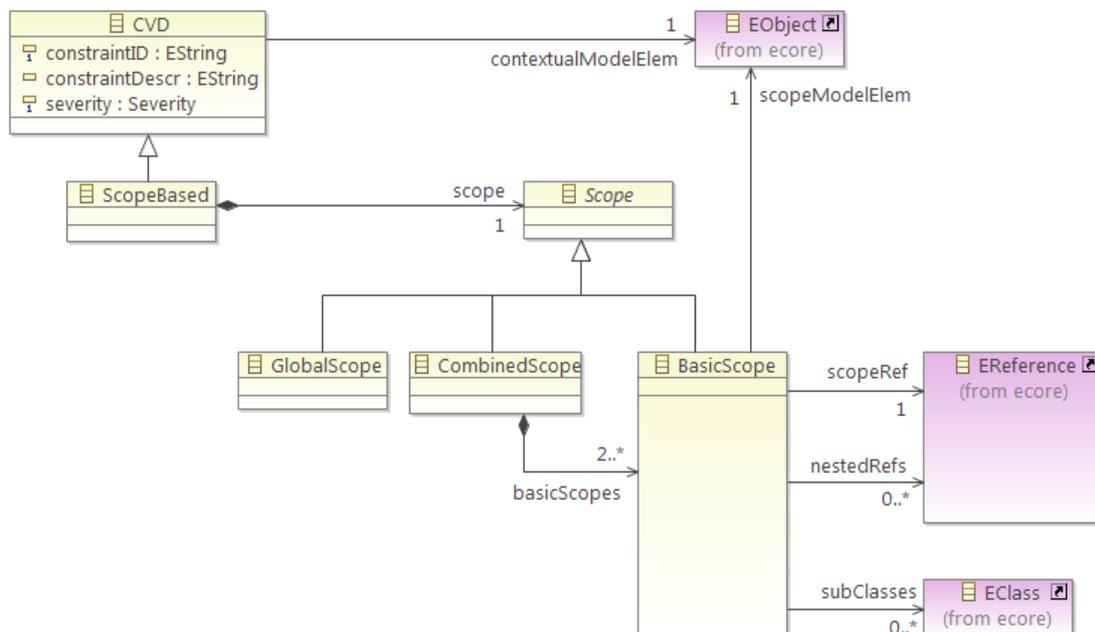


Figura 3.6

- **Relación con el meta-modelo CC.**

Lo natural es que esta clase sea referenciada por instancias de la clase del mismo nombre en el meta-modelo CC o de subclases suyas.

- **Manifestación de incumplimiento**

Con esta información descriptiva sería posible, por ejemplo, manifestar un incumplimiento construyendo un mensaje según el patrón (los *tokens* en negrita corresponden a datos modelados):

severity:

The **constraintID** constraint has been violated at **contextualModelElem.ID** : **contextualModelElem**.Type within the scope **scopeModelElem.ID** + "." + **scopeRef.ID**, regarding **subclasses[0].ID**, **subclasses[1].ID**, ... and **subclasses[N].ID** model elements.

- Location: mainContainer.ID >> ... **ancestor[i].ID** ... >> **contextualModelElem.ID**

- Constraint explanation: **constraintDescr**

MAST 2:

De entre las restricciones especificadas sobre el meta-modelo MAST2, la restricción ~~X.X.x~~ es un ejemplo de restricción que puede mapearse a esta clase ScopeBased, pues mediante ella pueden modelarse (los datos necesarios para describir) de manera detallada sus incumplimientos, de tal forma que en un modelo MAST 2, su incumplimiento por parte de un recurso de procesamiento podría manifestarse mediante un mensaje construido según el patrón anterior.

Error:

The **namelsUniqueAmongProcRsrcsInElemList** constraint has been violated at **Proc_1** : **Regular_Processor** within the scope **Teleoperated_Robot.Element_List**, regarding **Processing_Resource** model elements.

- Location: Teleoperated_Robot >> Proc_1

- Constraint explanation: **The namelsUniqueAmongProcRsrcsInElemList constraint is intended to...**

3.5 BasedOnFlowLine

- **Naturaleza de las restricciones a ser descrito su incumplimiento.**

Clase adecuada para modelar (los datos necesarios para describir) incumplimientos de restricciones de las estructuras de los elementos del modelo en base a cadenas de asociaciones. Por ejemplo restricciones de estructuras cíclicas, abiertas, en árbol, multiplicidad de los elementos en las cadenas, etc.

- **Estructura.**
- **Relación con el meta-modelo CC.**
- **Manifestación de incumplimiento.**

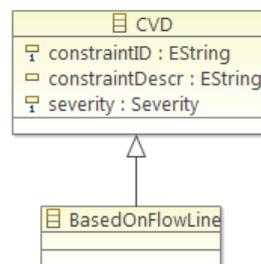


Figura 3.7

4 Subclases de BasedOnPropertiesOfContextClass

El meta-modelo CVD define varias subclases de `BasedOnPropertiesOfContextClass` buscando proporcionar formas de modelar (los datos necesarios para describir) aún en mayor detalle los incumplimientos de restricciones dentro de la variedad correspondiente a aquellas especificadas sobre propiedades de la clase contexto. La Fig. 4.1 muestra tales subclases, que se exponen a continuación.

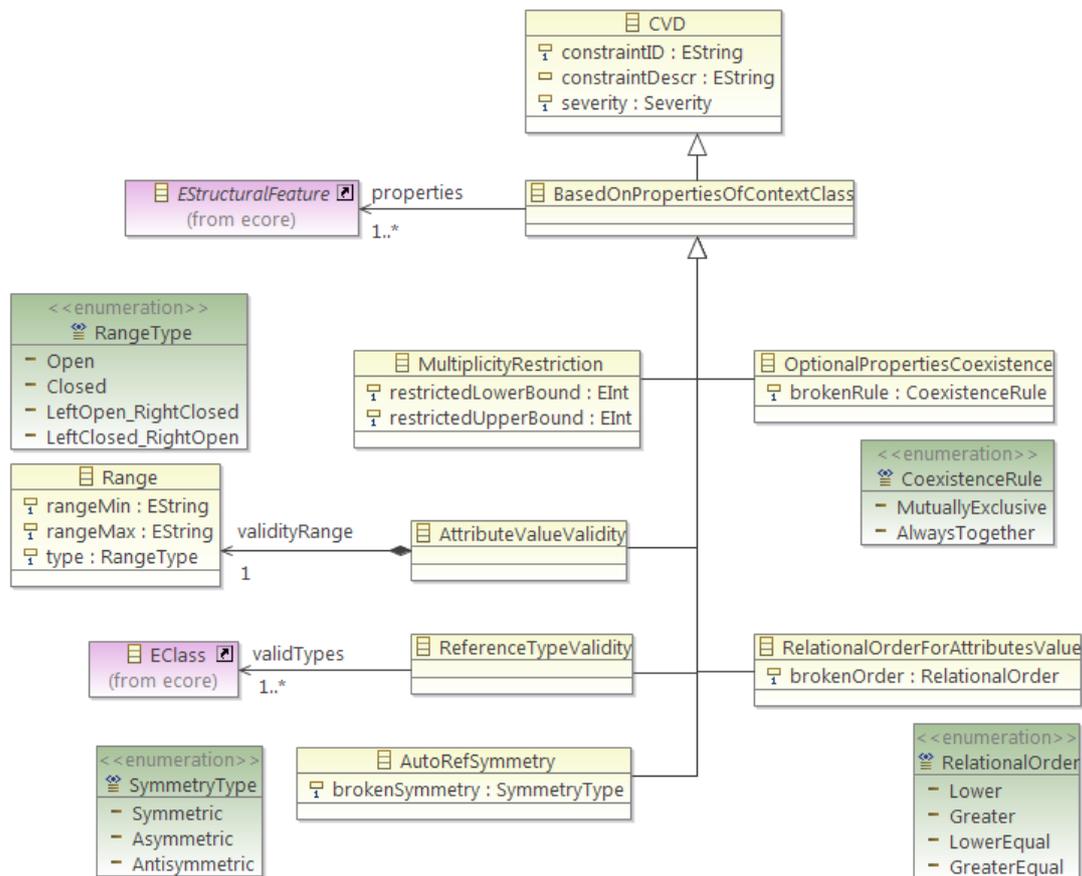


Figura 4.1

4.1 MultiplicityRestriction

- **Naturaleza de las restricciones a ser descrito su incumplimiento.**

Esta clase es adecuada exclusivamente para modelar la descripción de incumplimientos de restricciones consistentes en restringir la multiplicidad de una propiedad de la clase contexto.

- **Estructura.**

Junto a las posibilidades de modelado heredadas de `BasedOnPropertiesOfContextClass`, esta clase define dos atributos de tipo entero (`restrictedLowerBound` y `restrictedUpperBound`) que representan los límites de multiplicidad restringida, la cual no ha sido respetada.

Figura 4.2

- **Relación con el meta-modelo CC.**

Esta clase sólo puede ser referenciada por instancias de la clase del mismo nombre en el meta-modelo CC.

- **Manifestación de incumplimiento.**

Con esta información descriptiva sería posible, por ejemplo, manifestar un incumplimiento construyendo un mensaje según el patrón (los *tokens* en negrita corresponden a datos modelados):

severity:

The **constraintID** constraint has been violated at **contextualModelElem.ID** : **contextualModelElem.Type** by property **properties[0].ID**.

Allowed number of values / instances is [**restrictedLowerBound**, **restrictedUpperBound**]

- Location: mainContainer.ID >> ... **ancestor[i].ID** ... >> **contextualModelElem.ID**

- Constraint explanation: **constraintDescr**

MAST 2:

De entre las restricciones especificadas sobre el meta-modelo MAST2, no existe ninguna que pueda mapearse a esta clase MultiplicityRestriction. Una posible restricción impuesta por una herramienta externa que procesase modelos MAST 2 y mapeable a esta clase podría ser, por ejemplo, que un procesador deba obligatoriamente tener entre uno y dos temporizadores asociados. Su incumplimiento por parte de un procesador podría manifestarse mediante un mensaje construido según el patrón anterior.

Error:

The **onlyOneOrTwoTimers** constraint has been violated at **Proc_1** : **Regular_Processor** by property **Timer_List**.

Allowed number of instances is [**1**, **2**]

- Location: Teleoperated_Robot >> Proc_1

- Constraint explanation: **The onlyOneOrTwoTimers constraint is intended to...**

4.2 AttributeValueValidity

- **Naturaleza de las restricciones a ser descrito su incumplimiento.**

Esta clase es adecuada exclusivamente para modelar la descripción de incumplimientos de restricciones consistentes en restringir el abanico de valores que puede tomar un atributo de una clase.

Típicamente, el atributo es de un tipo primitivo numérico (int, float, etc.), pero también podría tratarse de atributos de tipo no numérico (un tipo enumerado, char, etc.).

- **Estructura.**

Junto a las capacidades de modelado heredadas de `BasedOnPropertiesOfContextClass`, esta clase define una asociación-composición `validityRange` que contiene la descripción del intervalo de validez, indicando sus extremos y su tipo { abierto (... , ...), cerrado [... , ...], semiabierto cerrado por la derecha (... , ...] semiabierto cerrado por la izquierda [... , ...) }.

Figura 4.3

- **Relación con el meta-modelo CC.**

Esta clase sólo puede ser referenciada por instancias de la clase del mismo nombre en el meta-modelo CC.

- **Manifestación de incumplimiento.**

Con esta información descriptiva sería posible, por ejemplo, manifestar un incumplimiento construyendo un mensaje según el patrón (los *tokens* en negrita corresponden a datos modelados):

```
severity:
The constraintID constraint has been violated at contextualModelElem.ID : contextualModelElem.Type
by attribute properties[0].ID.
Its value contextualModelElem.valueOf(properties[0]) is not within the valid range (rangeMin,
rangeMax).
- Location: mainContainerModelElem.ID >> ... ancestor[i].ID ... >> contextualModelElem.ID
- Constraint explanation: constraintDescr
```

MAST 2:

De entre las restricciones especificadas sobre el meta-modelo MAST2, la restricción ~~4.2.a~~ es un ejemplo de restricción que puede mapearse a esta clase `AttributeValueValidity`, pues mediante ella pueden modelarse (los datos necesarios para describir) de manera detallada sus incumplimientos, de tal forma que en un modelo MAST 2, su incumplimiento por parte de un recurso de procesamiento podría manifestarse mediante un mensaje construido según el patrón anterior.

Error:

The **speedFactorPositive** constraint has been violated at **Proc_1** : **Regular_Processor** by attribute **Speed_Factor**.
Its value **-3.5** is not within the valid range (**0.0**, **+inf**).

-
- Location: Teleoperated_Robot >> Proc_1
 - Constraint explanation: **The speedFactorPositive constraint is intended to...**
-

4.3 ReferenceTypeValidity

- **Naturaleza de las restricciones a ser descrito su incumplimiento.**

Esta clase es adecuada exclusivamente para modelar la descripción de incumplimientos de restricciones consistentes en restringir el abanico de clases que pueden tipar a una referencia.

Típicamente, esta situación se da cuando en el meta-modelo se ha establecido una clase abstracta como tipo de una referencia pero en determinadas circunstancias sólo se consideran admisibles ciertas subclases de ella.

- **Estructura.**

Junto a las capacidades de modelado heredadas de `BasedOnPropertiesOfContextClass`, esta clase define una asociación `validTypes` mediante la que especificar las clases consideradas admisibles.

Figura 4.4

- **Relación con el meta-modelo CC.**

Esta clase sólo puede ser referenciada por instancias de la clase del mismo nombre en el meta-modelo CC.

- **Manifestación de incumplimiento.**

Con esta información descriptiva sería posible, por ejemplo, manifestar un incumplimiento construyendo un mensaje según el patrón (los *tokens* en negrita corresponden a datos modelados):

```
severity:
The constraintID constraint has been violated at contextualModelElem.ID : contextualModelElem.Type
by reference properties[0].ID.
Its type contextualModelElem.typeOf(properties[0]) is not among the valid types {validTypes[0].ID, ...,
validTypes[N].ID}
- Location: mainContainer.ID >> ... ancestor[i].ID ... >> contextualModelElem.ID
- Constraint explanation: constraintDescr
```

MAST 2:

De entre las restricciones especificadas sobre el meta-modelo MAST2, la restricción ~~2-12.a~~ es un ejemplo de restricción que puede mapearse a esta clase `ReferencetypeValidity`, pues mediante ella pueden modelarse (los datos necesarios para describir) de manera detallada sus incumplimientos, de tal forma que en un modelo MAST 2, su incumplimiento por parte de un thread podría manifestarse mediante un mensaje construido según el patrón anterior.

Error:

```
The schedulingParamsAcceptable constraint has been violated at Thread_1 : Thread by
reference Scheduling_Params.
Its type AFDX_Virtual_Link is not among the valid types {Fixed_Priority_Params,
EDF_Based_Params, ... }
```

```
- Location: Teleoperated_Robot >> Thread_1
- Constraint explanation: The schedulingParamsAcceptable constraint is intended
to...
```

4.4 AutoRefSymmetry

- **Naturaleza de las restricciones a ser descrito su incumplimiento.**

Esta clase es adecuada exclusivamente para modelar la descripción de incumplimientos de restricciones consistentes en...

- **Estructura.**

Junto a las capacidades de modelado heredadas de `BasedOnPropertiesOfContextClass`, esta clase define un atributo `brokenSymmetry` que representa el tipo de simetría incumplida (simetría, asimetría o antisimetría).

- **Relación con el meta-modelo CC.**

- **Manifestación de incumplimiento.**

Con esta información descriptiva sería posible, por ejemplo, manifestar un incumplimiento construyendo un mensaje según el patrón (los *tokens* en negrita corresponden a datos modelados):

severity:

The **constraintID** constraint has been violated at **contextualModelElem.ID** : **contextualModelElem.Type** by property **properties[0].ID**, which ...

- Location: mainContainer.ID >> ... **ancestor[i].ID** ... >> **contextualModelElem.ID**

- Constraint explanation: **constraintDescr**

MAST 2:

De entre las restricciones especificadas sobre el meta-modelo MAST2, la restricción ~~X:X:x~~ es un ejemplo de restricción que puede mapearse a esta clase AutoRefSymmetry, pues mediante ella pueden modelarse (los datos necesarios para describir) de manera detallada sus incumplimientos, de tal forma que en un modelo MAST 2, su incumplimiento por parte de un XXX podría manifestarse mediante un mensaje construído según el patrón anterior.

Error:

The **xxxxxxxxxxxx** constraint has been violated at **X : X** by reference

- Location: Teleoperated_Robot >> X

- Constraint explanation: **The xxxxxxxxxxxx constraint is intended to...**

4.5 OptionalPropertiesCoexistence

- **Naturaleza de las restricciones a ser descrito su incumplimiento.**

Esta clase es adecuada exclusivamente para modelar (los datos necesarios para describir) incumplimientos de restricciones consistentes en imponer sobre un conjunto de propiedades opcionales en una clase la condición de que sean mutuamente excluyentes o de que hayan de estar presentes todas en bloque.

- **Estructura.**

Junto a las capacidades de modelado heredadas de `BasedOnPropertiesOfClass`, esta clase define un atributo `brokenRule` que representa el tipo de coexistencia **incumplida** (exclusión mútua o siempre aparecer juntas).

Figura 4.5

- **Relación con el meta-modelo CC.**

Esta clase sólo puede ser referenciada por instancias de la clase del mismo nombre en el meta-modelo CC.

- **Manifestación de incumplimiento.**

Con esta información descriptiva sería posible, por ejemplo, manifestar un incumplimiento construyendo un mensaje según el patrón (los *tokens* en negrita corresponden a datos modelados):

severity:

The **constraintID** constraint has been violated at **contextualModelElem.ID** : **contextualModelElem**.Type by properties {**properties[0].ID**, **properties[1].ID**, ..., **properties[N].ID**} which **are mutually exclusive / must appear always together**.

- Location: mainContainer.ID >> ... **ancestor[i].ID** ... >> **contextualModelElem.ID**

- Constraint explanation: **constraintDescr**

MAST 2:

De entre las restricciones especificadas sobre el meta-modelo MAST2, la restricción ~~X.X.x~~ es un ejemplo de restricción que puede mapearse a esta clase `OptionalPropertiesCoexistence`, pues mediante ella pueden modelarse (los datos necesarios para describir) de manera detallada sus incumplimientos, de tal forma que en un modelo MAST 2, su incumplimiento por parte de una operación simple podría manifestarse mediante un mensaje construido según el patrón anterior.

Error:

The **mutexListMutexToLockList** constraint has been violated at **Oper_1** : **Simple_Operation** by properties {**Mutex_List**, **Mutex_To_Lock_List**}, which are **mutually exclusive**.

- Location: Teleoperated_Robot >> Oper_1

- Constraint explanation: **The mutexListMutexToLockList constraint is intended to...**

4.6 RelationalOrderForAttributesValue

- **Naturaleza de las restricciones a ser descrito su incumplimiento.**

Esta clase es adecuada exclusivamente para modelar la descripción de incumplimientos de restricciones consistentes en, dado un subconjunto dentro de los atributos de una clase, imponer un determinado orden relacional entre los valores que adoptan los atributos pertenecientes a tal subconjunto.

Típicamente, los atributos sobre los que se especifica la restricción poseerán una semántica íntimamente relacionada.

Al igual que en el caso de `AttributeValueValidity`, el tipo de los atributos no ha de ser obligatoriamente numérico, sino que en general podría ser cualquier tipo para el que se haya establecido un orden bien definido, como por ejemplo 'char' o un tipo enumerado.

- **Estructura.**

Junto a las capacidades de modelado heredadas de `BasedOnPropertiesOfContextClass`, esta clase define un atributo `brokenOrder` que representa el tipo de orden relacional **incumplido** { <, >, ≤, ≥ }.

Figura 4.6

- **Relación con el meta-modelo CC.**

Esta clase sólo puede ser referenciada por instancias de la clase del mismo nombre en el meta-modelo CC.

- **Manifestación de incumplimiento.**

Con esta información descriptiva sería posible, por ejemplo, manifestar un incumplimiento construyendo un mensaje según el patrón (los *tokens* en negrita corresponden a datos modelados):

severity:

The **constraintID** constraint has been violated at **contextualModelElem.ID** : **contextualModelElem.Type** by attributes **properties[0].ID** (**contextualModelElem.valueOf(properties[0])**), **properties[1].ID** (**contextualModelElem.valueOf(properties[1])**), ... and **properties[N].ID** (**contextualModelElem.valueOf(properties[N])**), which do not respect the order **brokenOrder**.

- Location: mainContainer.ID >> ... **ancestor[i].ID** ... >> **contextualModelElem.ID**

- Constraint explanation: **constraintDescr**

MAST 2:

De entre las restricciones especificadas sobre el meta-modelo MAST2, la restricción ~~X.X.x~~ es un ejemplo de restricción que puede mapearse a esta clase `RelationalOrderForAttributesValue`, pues mediante ella pueden modelarse (los datos necesarios para describir) de manera detallada sus incumplimientos, de tal forma que en un modelo MAST 2, su incumplimiento por parte de un procesador podría manifestarse mediante un mensaje construido según el patrón anterior.

Error:

The **maxInterrPrioGreaterThanMin** constraint has been violated at **Proc_1 : Regular_Processor** by attributes **Max_Interrupt_Priority** (32) and **Min_Interrupt_Priority** (33), which do not respect the order \geq .

- Location: Teleoperated_Robot >> Proc_1
 - Constraint explanation: **the maxInterrPrioGreaterThanMin constraint is intended to**
...
-

5 Subclases de BasedOnIncompatibility

El meta-modelo CVD define varias subclases de `BasedOnIncompatibility` buscando proporcionar formas de modelar (los datos necesarios para describir) aún en mayor detalle los incumplimientos de restricciones dentro de la variedad correspondiente a aquellas consistentes en establecer incompatibilidades entre subclases de dos clases del meta-modelo. La **Fig. 6-5** muestra tales especializaciones de `BasedOnIncompatibility`, que se exponen a continuación.

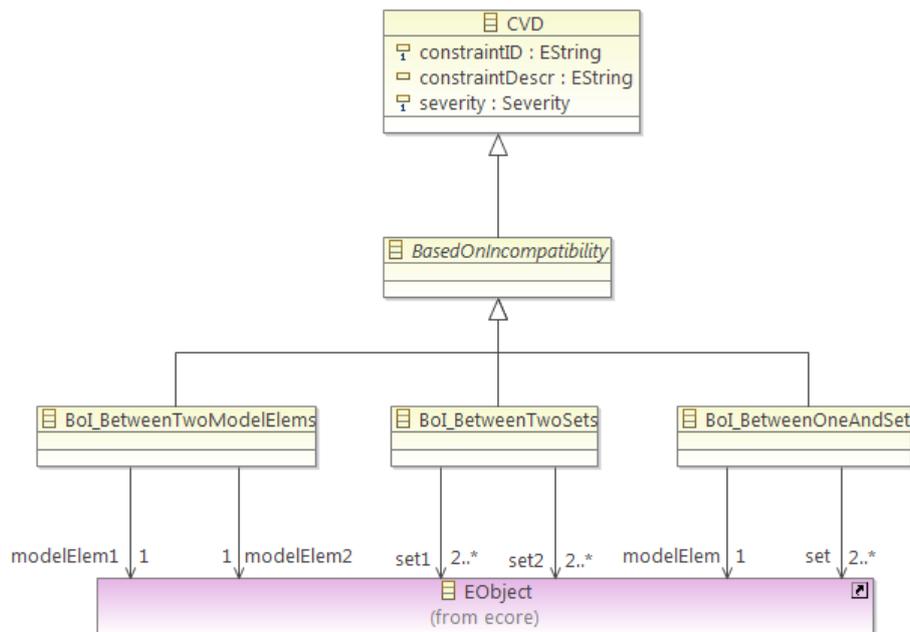


Figura 5.1

5.1 BoI_BetweenTwoModelElems

- **Naturaleza de las restricciones a ser descrito su incumplimiento.**

Esta clase es adecuada exclusivamente para modelar (los datos necesarios para describir) incumplimientos de restricciones mapeables a la clase `BasedOnIncompatibility` cuando la compatibilidad incumplida se da entre dos elementos de modelo individuales.

- **Estructura.**

Junto a las capacidades de modelado heredadas de `CVD`, esta clase define dos referencias `modelElem1` y `modelElem2` que indican los elementos de modelo incompatibles.

- **Relación con el meta-modelo CC.**

- **Manifestación de incumplimiento.**

Con esta información descriptiva sería posible, por ejemplo, manifestar un incumplimiento construyendo un mensaje según el patrón (los *tokens* en negrita corresponden a datos modelados):

severity:

The **constraintID** constraint has been violated at **contextualModelElem.ID : contextualModelElem.Type** by incompatible model elements **modelElem1.ID : modelElem1.Type** and **modelElem2.ID : modelElem2.Type**.

- Location: mainContainer.ID >> ... **ancestor[i].ID** ... >> **contextualModelElem.ID**

- Constraint explanation: **constraintDescr**

MAST 2:

De entre las restricciones especificadas sobre el meta-modelo MAST2, las restricciones [3.5.a\)](#), [3.6.a\)](#), [3.7.a\)](#) y [3.8.a\)](#) son un ejemplo de restricción que puede mapearse a esta clase `BoI_BetweenTwoModelElems`, pues mediante ella pueden modelarse (los datos necesarios para describir) de manera detallada sus incumplimientos, de tal forma que en un modelo MAST 2, el incumplimiento de cualquiera de ellas podría manifestarse mediante un mensaje construído según el patrón anterior.

Error:

The **policyCompatibleWithHost** constraint has been violated at **Sched_1 : Primary_Scheduler** by incompatible model elements **XXX : FP_Packet_Based_Policy** and **Proc_1 : Regular_Processor**.

- Location: Teleoperated_Robot >> Sched_1

- Constraint explanation: **the policyCompatibleWithHost constraint is intended to ...**

Error:

The **compatibleWithSchedulerHost** constraint has been violated at **Server_1 : Thread** by incompatible model elements **Server_1 : Thread** and **Net_1 : Packet_Based_Network**.

- Location: Teleoperated_Robot >> Server_1

- Constraint explanation: **the compatibleWithSchedulerHost constraint is intended to**

...

Error:

The **schedulingParamsCompatibleWithSchedulerPolicy** constraint has been violated at **Server_1 : Thread** by incompatible model elements **XXX : EDF_Params** and **XXX : Fixed_Priority_Policy**.

- Location: Teleoperated_Robot >> Server_1
- Constraint explanation: **the schedulingParamsCompatibleWithSchedulerPolicy constraint is intended to ...**

Error:

The **operCompatibleWithServer** constraint has been violated at **XXX : Step** by incompatible model elements **Mssg_1 : Message_Operation** and **Server_1 : Thread**.

- Location: Teleoperated_Robot >> **XXX**
 - Constraint explanation: **the operCompatibleWithServer constraint is intended to ...**
-

5.2 BoI_BetweenTwoSets

- **Naturaleza de las restricciones a ser descrito su incumplimiento.**

Esta clase es adecuada exclusivamente para modelar (los datos necesarios para describir) incumplimientos de restricciones mapeables a la clase `BasedOnIncompatibility` cuando la compatibilidad incumplida se da entre dos conjuntos de elementos de modelo.

- **Estructura.**

Junto a las capacidades de modelado heredadas de `CVD`, esta clase define dos referencias `set1` y `set2` que indican los conjuntos de elementos entre los que se produce la incompatibilidad.

- **Relación con el meta-modelo CC.**

- **Manifestación de incumplimiento.**

Con esta información descriptiva sería posible, por ejemplo, manifestar un incumplimiento construyendo un mensaje según el patrón (los *tokens* en negrita corresponden a datos modelados):

severity:

The **constraintID** constraint has been violated at **contextualModelElem.ID** : **contextualModelElem.Type** by the incompatible sets of model elements {**set1[0].ID**, ..., **set1[N].ID**} and {**set2[0].ID**, ..., **set2[M].ID**}

- Location: mainContainer.ID >> ... **ancestor[i].ID** ... >> **contextualModelElem.ID**

- Constraint explanation: **constraintDescr**

MAST 2:

5.3 BoI_BetweenOneAndSet

- **Naturaleza de las restricciones a ser descrito su incumplimiento.**

Esta clase es adecuada exclusivamente para modelar (los datos necesarios para describir) incumplimientos de restricciones mapeables a la clase `BasedOnIncompatibility` cuando la compatibilidad incumplida se da entre un elemento de modelo y un conjunto de elementos de modelo.

- **Estructura.**

Junto a las capacidades de modelado heredadas de `CVD`, esta clase define dos referencias `modelElem` y `set` que indican cuáles son el elemento de modelo y el conjunto de elementos de modelo entre los que se produce la incompatibilidad.

- **Relación con el meta-modelo CC.**

- **Manifestación de incumplimiento.**

Con esta información descriptiva sería posible, por ejemplo, manifestar un incumplimiento construyendo un mensaje según el patrón (los *tokens* en negrita corresponden a datos modelados):

severity:

The **constraintID** constraint has been violated at **contextualModelElem.ID** : **contextualModelElem.Type** by the incompatible model element **modelElem.ID** and set {**set[0].ID**, ..., **set[M].ID**}

- Location: mainContainer.ID >> ... **ancestor[i].ID** ... >> **contextualModelElem.ID**

- Constraint explanation: **constraintDescr**

MAST 2:

6 Subclases de ScopedBased

El meta-modelo CVD define varias subclases de `ScopeBased` buscando proporcionar formas de modelar aún en mayor detalle la descripción de incumplimientos de restricciones dentro de la variedad correspondiente a aquellas cuya satisfacción depende no sólo del estado de un elemento de modelo, sino también del ámbito dentro del cual éste se halla. La Fig. 6.1 muestra tales subclases, que se exponen a continuación.

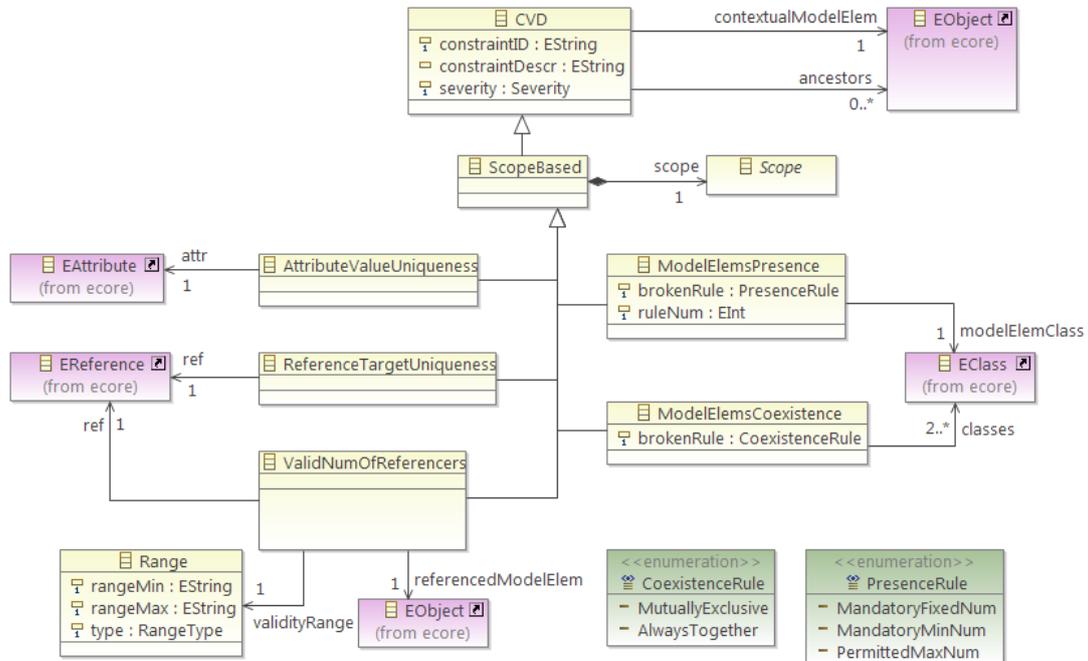


Figura 6.1

6.1 AttributeValueUniqueness

- **Naturaleza de las restricciones a ser descrito su incumplimiento.**

Esta clase es adecuada exclusivamente para modelar la descripción de incumplimientos de restricciones consistentes en establecer que, en un ámbito o población de elementos de modelo, el valor de un atributo dado ha de ser único para cada elemento de modelo en ese ámbito.

- **Estructura.**

Junto a las capacidades de modelado heredadas de `ScopeBased`, esta clase define una referencia `attr` mediante la que especificar cuál es el atributo cuyo valor no puede hallarse repetido entre los elementos de modelo del ámbito.

Figura 5.2

- **Relación con el meta-modelo CC.**

Esta clase sólo puede ser referenciada por instancias de la clase del mismo nombre en el meta-modelo CC.

- **Manifestación de incumplimiento.**

Con esta información descriptiva sería posible, por ejemplo, manifestar un incumplimiento construyendo un mensaje según el patrón (los *tokens* en negrita corresponden a datos modelados):

severity:

The **constraintID** constraint has been violated at **contextualModelElem.ID** :**contextualModelElem**.Type within the scope **scopeModelElem.ID** + "." + **scopeRef.ID**, regarding **subclasses[0].ID**, **subclasses[1].ID**, ... and **subclasses[N].ID** model elements. Value of attribute **attr.ID** is not unique.

- Location: mainContainer.ID >> ... **ancestor[i].ID** ... >> **contextualModelElem.ID**

- Constraint explanation: **constraintDescr**

MAST 2:

De entre las restricciones especificadas sobre el meta-modelo MAST2, la restricción ~~X.X.x~~ es un ejemplo de restricción que puede mapearse a esta clase `AttributeValueUniqueness`, pues mediante ella pueden modelarse (los datos necesarios para describir) de manera detallada sus incumplimientos, de tal forma que en un modelo MAST 2, su incumplimiento por parte de un recurso de procesamiento podría manifestarse mediante un mensaje construido según el patrón anterior.

Error:

The **namelsUniqueAmongProcRsrcsInElemList** constraint has been violated at **Proc_1** : **Regular_Processor** within the scope **Teleoperated_Robot.Element_List**, regarding **Processing_Resource** model elements.

Value of attribute **name** is not unique.

- Location: Teleoperated_Robot >> Proc_1

- Constraint explanation: **The namelsUniqueAmongProcRsrcsInElemList constraint is intended to...**

6.2 ReferenceTargetUniqueness

- **Naturaleza de las restricciones a ser descrito su incumplimiento.**

Esta clase es adecuada exclusivamente para modelar la descripción de incumplimientos de restricciones consistentes en establecer que, en un ámbito o población de elementos de modelo, el *target* de una referencia dada ha de ser único para cada elemento de modelo en ese ámbito.

- **Estructura.**

Junto a las capacidades de modelado heredadas de `ScopeBased`, esta clase define una referencia `ref` que indica cuál es la referencia cuyo *target* no puede hallarse repetido entre los elementos de modelo del ámbito.

Figura 5.3

- **Relación con el meta-modelo CC.**

Esta clase sólo puede ser referenciada por instancias de la clase del mismo nombre en el meta-modelo CC.

- **Manifestación de incumplimiento.**

Con esta información descriptiva sería posible, por ejemplo, manifestar un incumplimiento construyendo un mensaje según el patrón (los *tokens* en negrita corresponden a datos modelados):

severity:

The **constraintID** constraint has been violated at **contextualModelElem.ID** :**contextualModelElem.Type** within the scope **scopeModelElem.ID** + "." + **scopeRef.ID**, regarding **subclasses[0].ID**, **subclasses[1].ID**, ... and **subclasses[N].ID** model elements.

Target of reference **ref.ID** is not unique.

- Location: mainContainer.ID >> ... **ancestor[i].ID** ... >> **contextualModelElem.ID**

- Constraint explanation: **constraintDescr**

MAST 2:

De entre las restricciones especificadas sobre el meta-modelo MAST2, la restricción ~~X.X.x)~~ es un ejemplo de restricción que puede mapearse a esta clase `ReferenceTargetUniqueness`, pues mediante ella pueden modelarse (los datos necesarios para describir) de manera detallada sus incumplimientos, de tal forma que en un modelo MAST 2, su incumplimiento por parte de un recurso de procesamiento podría manifestarse mediante un mensaje construido según el patrón anterior.

Error:

The **primarySchedulersDoNotShareHost** constraint has been violated at **Teleoperated_Robot : Mast_Model**, regarding **Primary_Scheduler** model elements.

Target of ref **host** is not unique.

- Location: Teleoperated_Robot

- Constraint explanation: **The primarySchedulersDoNotShareHost constraint is intended to...**

6.3 ModelElemsPresence

- **Naturaleza de las restricciones a ser descrito su incumplimiento.**

Esta clase es adecuada exclusivamente para modelar la descripción de incumplimientos de restricciones consistentes en acotar el número de elementos de modelo *de un determinado tipo* dentro de un ámbito.

La acotación puede significar establecer un número fijo y obligatorio de elementos o bien una cota superior / inferior.

- **Estructura.**

Junto a las capacidades de modelado heredadas de `ScopeBased`, esta clase define dos atributos `brokenRule` y `ruleNum` que representan conjuntamente la regla de presencia **incumplida**.

El atributo `brokenRule` indica si la restricción **incumplida** establecía un número obligatorio de elementos de modelo, una cota inferior o una cota superior mientras que el atributo `ruleNum` especifica el valor de ese número o cota.

Además, la clase define una referencia `modelElemClass` mediante la que especificar el tipo de aquellos elementos de modelo dentro del ámbito sobre los que se encuentra establecida la regla de presencia incumplida.

Figura 5.4

- **Relación con el meta-modelo CC.**

Esta clase sólo puede ser referenciada por instancias de la clase del mismo nombre en el meta-modelo CC.

- **Manifestación de incumplimiento.**

Con esta información descriptiva sería posible, por ejemplo, manifestar un incumplimiento construyendo un mensaje según el patrón (los *tokens* en negrita corresponden a datos modelados):

```
severity:
The constraintID constraint has been violated at contextualModelElem.ID : contextualModelElem.Type
within the scope scope.scopeModelElem.ID + "." + scope.scopeRef.ID of kind scope.filterSubClass.
There are presentModelElemsNum presentModelElemsType instances and [there has to be / there
has to be at least / there cannot be more than] ruleNum.
- Location: mainContainer.ID >> ... ancestor[i].ID ... >> contextualModelElem.ID
- Constraint explanation: constraintDescr
```

MAST 2:

Las restricciones 2.1.b) y 2.4.a) del documento “MAST 2 constraints” son ejemplos de restricciones que pueden mapearse a esta clase ModelElemsPresence, pues mediante ella pueden modelarse los datos necesarios para describir de manera detallada su incumplimiento, de tal forma que en un modelo MAST 2, su incumplimiento por parte de un ... podría manifestarse mediante un mensaje construido según el patrón anterior.

Error:

The **atLeastOnePrimaryScheduler** constraint has been violated at **Teleoperated_Robot : Mast_Model** within the scope **Teleoperated_Robot.Element_List** of kind **Primary_Scheduler**.

There are **0 Primary_Scheduler** instances and there has to be at least **1**.

- Location: Teleoperated_Robot
- Constraint explanation: **The atLeastOnePrimaryScheduler constraint is intended to...**

Error:

The **atLeastOneStep** constraint has been violated at **E2EF_1 : Regular_End_To_End_Flow** within the scope **E2EF_1.Flow_Element_List** of kind **Step**. There are **0 Step** instances and there has to be at least **1**.

- Location: Teleoperated_Robot >> E2EF_1
 - Constraint explanation: **The atLeastOneStep constraint is intended to...**
-

6.4 ModelElemsCoexistence

- **Naturaleza de las restricciones a ser descrito su incumplimiento.**

Esta clase es adecuada exclusivamente para modelar la descripción de incumplimientos de restricciones consistentes en

- **Estructura.**

Junto a las capacidades de modelado heredadas de `ScopeBased`, esta clase define un atributo `brokenRule` que representa ... y una referencia `classes` que representa

- **Relación con el meta-modelo CC.**

- **Manifestación de incumplimiento.**

Con esta información descriptiva sería posible, por ejemplo, manifestar un incumplimiento construyendo un mensaje según el patrón (los *tokens* en negrita corresponden a datos modelados):

severity:

The **constraintID** constraint has been violated at **contextualModelElem.ID** : **contextualModelElem.Type** within the scope **scope.scopeModelElem.ID** + "." + **scope.scopeRef.ID** of kind **scope.filterSubClass**.

- Location: mainContainer.ID >> ... **ancestor[i].ID** ... >> **contextualModelElem.ID**

- Constraint explanation: **constraintDescr**

MAST 2:

Las restricciones 2.1.b) y 2.4.a) del documento "MAST 2 constraints" son ejemplos de restricciones que pueden mapearse a esta clase ModelElemsPresence, pues mediante ella pueden modelarse los datos necesarios para describir de manera detallada su incumplimiento, de tal forma que en un modelo MAST 2, su incumplimiento por parte de un ... podría manifestarse mediante un mensaje construido según el patrón anterior.

Error:

~~The **atLeastOnePrimaryScheduler** constraint has been violated at **Teleoperated_Robot : Mast_Model** within the scope **Teleoperated_Robot.Element_List** of kind **Primary_Scheduler**.
There are **0 Primary_Scheduler** instances and there has to be at least **1**.~~

~~Location: **Teleoperated_Robot**~~

~~Constraint explanation: **The atLeastOnePrimaryScheduler constraint is intended to...**~~
