A Framework for Interoperability Between Models with Hybrid Tools

Germán Braun[1,2*], Pablo Rubén Fillottrani[3,4†] and C. Maria Keet[5†]

[1*]Universidad Nacional del Comahue, Buenos Aires 1400, Argentina.
[2]Consejo Nacional de Investigaciones Científicas y Técnicas, Godoy Cruz 2290 (C1425FQB), Buenos Aires, Argentina.
[3]Universidad Nacional del Sur, San Andrés 800, Bahía Blanca, Argentina.
[4]Comisión de Investigaciones Científicas de la provincia de Buenos Aires, calle 526 e/10 y 11, La Plata, Argentina.
[5]Department of Computer Science, University of Cape Town, 18 University Avenue, Rondebosch, Cape Town, 7700, South Africa.

*Corresponding author(s). E-mail(s): german.braun@fi.uncoma.edu.ar;
Contributing authors: prf@cs.uns.edu.ar; mkeet@cs.uct.ac.za;
†These authors contributed equally to this work.

1 Interoperability Rules for crowd 2.0

This document includes the set of Interoperability Rules currently implemented in the tool crowd 2.0[1]

1.1 *UML/KF Rules*

(**UML-O1**) Class $\xrightarrow{\text{UML to KF}}$ `Object Type`
in: Class

[1] https://crowd-app.fi.uncoma.edu.ar/

out: `Object Type`

(**UML-1O**) `Object Type` $\overset{\text{KF to UML}}{\Longrightarrow}$ Class
in: `Object Type`
out: Class

(**UML-R1**) Association end $\overset{\text{UML to KF}}{\Longrightarrow}$ `Role`
in: Association end
in: Association(Association end: Class, __)
in: MultiplicityConstraint(Association end, min, max)
out: Association end → `Role`
out: `UML-O1(Class)`
out: `UML-A1(Association)`
out: `UML-MC1(MultiplicityConstraint)`
out: `Role(Relationship, Object type, CardinalityConstraint)`

(**UML-1R**) `Role` $\overset{\text{KF to UML}}{\Longrightarrow}$ Association end
in: `Role(Relationship, Object type, CardinalityConstraint)`
in: `Relationship(Role: Object type, __)`
in: `CardinalityConstraint(Role, min, max)`
out: `Role` → Association end
out: UML-1O(Object type)
out: UML-1A(Relationship)
out: UML-1MC(CardinalityConstraint)

(**UML-DT1**) Data type $\overset{\text{UML to KF}}{\Longrightarrow}$ `Data type`
in: Data type
out: `Data type`

(**UML-1DT**) `Data type` $\overset{\text{KF to UML}}{\Longrightarrow}$ Data type
in: `Data type`
out: Data type

(**UML-M1**) Mandatory role $\overset{\text{UML to KF}}{\Longrightarrow}$ `Mandatory`
in: MultiplicityConstraint(Association end, 1, max)
out: `Mandatory(UML-R1(Association end))`

(**UML-1M**) `Mandatory` $\overset{\text{KF to UML}}{\Longrightarrow}$ Mandatory role
in: `Mandatory(Role)`
out: MultiplicityConstraint(UML-1R(Role), 1, _)

(**UML-S1**) Subclass $\overset{\text{UML to KF}}{\Longrightarrow}$ `Subsumption`
in: Subclass(Child: Class, Parent : Class)
out: `Subsumption(Child: UML-O1(Class), Parent: UML-O1(Class))`

(**UML-1S**) `Subsumption` $\overset{\text{KF to UML}}{\Longrightarrow}$ Subclass
in: `Subsumption(Child: Object type, Parent: Object type)`
out: Subclass(Child: UML-1O(Object type), Parent: UML-1O(Object type))

(**UML-C1**) Complete $\overset{\text{UML to KF}}{\Longrightarrow}$ `Completeness constraint`
in: Subclass(Child-1: Class, Parent: Class)
in: Subclass(Child-2: Class, Parent: Class)
in: Complete(Child-1, Child-2)
out: `UML-S1(Subclass)`
out: `UML-S1(Subclass)`
out: `CompletenessConstraint(Child-1: Object type, Child-2: Object type)`

(**UML-1C**) `Completeness constraint` $\overset{\text{KF to UML}}{\Longrightarrow}$ Complete
in: `Subsumption(Child-1: Object type, Parent: Object type)`
in: `Subsumption(Child-2: Object type, Parent: Object type)`
in: `CompletenessConstraint(Child-1, Child-2)`
out: UML-1S(Subsumption)
out: UML-1S(Subsumption)
out: Complete(Child-1: Class, Child-2: Class)

(**UML-D1**) Disjoint $\overset{\text{UML to KF}}{\Longrightarrow}$ `Disjoint object type`
in: Subclass(Child-1: Class, Parent: Class)
in: Subclass(Child-2: Class, Parent: Class)
in: Disjoint(Child-1, Child-2)
out: `UML-S1(Subclass)`
out: `UML-S1(Subclass)`
out: `DisjointObjectType(Child-1: Object type, Child-2: Object`

type)

(**UML-1D**) Disjoint object type $\xrightarrow{\text{KF to UML}}$ Disjoint
in: Subsumption(Child-1: Object type, Parent: Object type)
in: Subsumption(Child-2: Object type, Parent: Object type)
in: DisjointObjectType(Child-1, Child-2)
out: UML-1S(Subsumption)
out: UML-1S(Subsumption)
out: Disjoint(Child-1: Class, Child-2: Class)

(**UML-ATT1**) Attribute $\xrightarrow{\text{UML to KF}}$ Attribute
in: Attribute(Class, Data type)
out: Attribute(UML-O1(Class), UML-DT1(Datatype))

(**UML-1ATT**) Attribute $\xrightarrow{\text{KF to UML}}$ Attribute
in: Attribute(Object type, Data type)
out: Attribute(UML-1O(Object type), UML-1DT(Datatype))

(**UML-A1**) Association $\xrightarrow{\text{UML to KF}}$ Relationship
in: Association(Association end: Class, Association end: Class)
out: Association → Relationship
out: Relationship(UML-R1(Association end):UML-O1(Class), UML-R1(Association end):UML-O1(Class))

(**UML-1A**) Relationship $\xrightarrow{\text{KF to UML}}$ Association
in: Relationship(Role: Object type, Role: Object type)
out: Relationship → Association
out: Association(UML-1R(Role):UML-1O(Object type), UML-1R(Role):UML-1O(Object type))

(**UML-MC1**) Multiplicity constraint $\xrightarrow{\text{UML to KF}}$ Object type cardinality constraint
in: MultiplicityConstraint(Association end, min, max)
out: ObjectTypeCardinalityConstraint(UML-R1(Association end), min, max)

(**UML-1MC**) `Object type cardinality constraint` $\overset{\text{KF to UML}}{\Longrightarrow}$ Multiplicity constraint
in: `ObjectTypeCardinalityConstraint(Role, min, max)`
out: MultiplicityConstraint(UML-1R(Role), min, max)

(**UML-SA1**) Subtyping of Association $\overset{\text{UML to KF}}{\Longrightarrow}$ `Sub - relationship`
in: Subtyping(Child: Association, Parent: Association)
out: Association → `Relationship`
out: Association → `Relationship`
out: `Subsumption(Child:Relationship, Parent:Relationship)`

(**UML-1SA**) `Sub - relationship` $\overset{\text{KF to UML}}{\Longrightarrow}$ Subtyping of association
in: `Subsumption(Child:Relationship, Parent:Relationship)`
out: `Relationship` → Association
out: `Relationship` → Association
out: Subtyping(Child:Association, Parent:Association)

1.2 *ER/KF Rules*

(**ER-O1**) Entity type $\overset{\text{ER to KF}}{\Longrightarrow}$`Object Type`
in: Entity type
out: `Object Type`

(**ER-1O**) `Object Type` $\overset{\text{KF to ER}}{\Longrightarrow}$Entity type
in: `Object Type`
out: Entity type

(**ER-R1**) Component of relationship $\overset{\text{ER to KF}}{\Longrightarrow}$`Role`
in: Component of relationship
in: Relationship(Component of relationship: Entity type, __)
in: CardinalityConstraint(Component of relationship, min, max)
out: Component of relationship → `Role`
out: `ER-O1(Entity type)`
out: `ER-A1(Relationship)`
out: `ER-MC1(CardinalityConstraint)`
out: `Role(Relationship, Object type, CardinalityConstraint)`

(**ER-1R**) `Role` $\overset{\text{KF to ER}}{\Longrightarrow}$ Component of relationship
in: `Role(Relationship, Object type, CardinalityConstraint)`
in: `Relationship(Role: Object type, __)`
in: `CardinalityConstraint(Role, min, max)`
out: `Role` → Component of relationship
out: ER-1O(Object type)
out: ER-1A(Relationship)
out: ER-1MC(CardinalityConstraint)

(**ER-M1**) Mandatory $\overset{\text{ER to KF}}{\Longrightarrow}$`Mandatory`
in: CardinalityConstraint(Component of relationship, 1, max)
out: `Mandatory(ER-R1(Component of relationship))`

(**ER-1M**) `Mandatory` $\overset{\text{KF to ER}}{\Longrightarrow}$Mandatory
in: `Mandatory(Role)`
out: CardinalityConstraint(ER-1R(Role), 1, *)

(**ER-S1**) Subtype $\overset{\text{ER to KF}}{\Longrightarrow}$ `Subsumption`
in: Subtype(Child: Entity type, Parent: Entity type)
out: `Subsumption(Child: ER-O1(Entity type), Parent: ER-O1(Entity type))`

(**ER-1S**) `Subsumption` $\overset{\text{KF to ER}}{\Longrightarrow}$ Subtype
in: `Subsumption(Child: Object type, Parent: Object type)`
out: Subtype(Child: ER-1O(Object type), Parent: ER-1O(Object type))

(**ER-C1**) Total $\overset{\text{ER to KF}}{\Longrightarrow}$ `Completeness constraint`
in: Subtype(Child-1: Entity type, Parent: Entity type)
in: Subtype(Child-2: Entity type, Parent: Entity type)
in: Total(Child-1, Child-2)
out: `ER-S1(Subtype)`
out: `ER-S1(Subtype)`
out: `CompletenessConstraint(Child-1: Object type, Child-2: Object type)`

(**ER-1C**) `Completeness constraint` $\overset{\text{KF to ER}}{\Longrightarrow}$ Total
in: `Subsumption(Child-1: Object type, Parent: Object type)`

in: `Subsumption(Child-2: Object type, Parent: Object type)`
in: `CompletenessConstraint(Child-1, Child-2)`
out: ER-1S(Subsumption)
out: ER-1S(Subsumption)
out: Total(Child-1: Entity type, Child-2: Entity type)

(**ER-D1**) Disjoint $\overset{\text{ER to KF}}{\Longrightarrow}$ `Disjoint object type`
in: Subtype(Child-1: Entity type, Parent: Entity type)
in: Subtype(Child-2: Entity type, Parent: Entity type)
in: Disjoint(Child-1, Child-2)
out: `ER-S1(Subtype)`
out: `ER-S1(Subtype)`
out: `DisjointObjectType(Child-1: Object type, Child-2: Object type)`

(**ER-1D**) `Disjoint object type` $\overset{\text{KF to ER}}{\Longrightarrow}$ Disjoint
in: `Subsumption(Child-1: Object type, Parent: Object type)`
in: `Subsumption(Child-2: Object type, Parent: Object type)`
in: `DisjointObjectType(Child-1, Child-2)`
out: ER-1S(Subsumption)
out: ER-1S(Subsumption)
out: Disjoint(Child-1: Entity type, Child-2: Entity type)

(**ER-ATT1**) Attribute $\overset{\text{ER to KF}}{\Longrightarrow}$ `Attribute`
in: Attribute(Entity type, __)
out: `ER-O1(Entity type)`
out: `ER-D1(__)` *// Datatype given by the user*
out: `Attribute(Object type, Data type)`

(**ER-1ATT**) `Attribute` $\overset{\text{KF to ER}}{\Longrightarrow}$ Attribute
in: `Attribute(Object type, Data type)`
out: ER-1O(Object type)
out: Attribute(Entity type, __)

(**ER-A1**) Relationship $\overset{\text{ER to KF}}{\Longrightarrow}$ `Relationship`
in: Relationship(Component of relationship: Entity type, Component of relationship: Entity type)
out: Relationship → `Relationship`

out: `Relationship(ER-R1(Component of relationship):ER-O1(Entity type), ER-R1(Component of relationship):ER-O1(Entity type))`

(**ER-1A**) `Relationship` $\overset{\text{KF to ER}}{\Longrightarrow}$ Relationship
in: `Relationship(Role: Object type, Role: Object type)`
out: `Relationship` → Relationship
out: Relationship(ER-1R(Role):ER-1O(Object type), ER-1R(Role):ER-1O(Object type))

(**ER-MC1**) Cardinality constraint $\overset{\text{ER to KF}}{\Longrightarrow}$ `Object type cardinality constraint`
in: CardinalityConstraint(Component of relationship, min, max)
out: `ObjectTypeCardinalityConstraint(ER-R1(Component of relationship), min, max)`

(**ER-1MC**) `Object type cardinality constraint` $\overset{\text{KF to ER}}{\Longrightarrow}$ Cardinality constraint
in: `ObjectTypeCardinalityConstraint(Role, min, max)`
out: CardinalityConstraint(ER-1R(Role), min, max)

(**ER-SA1**) Subtyping of Relationship $\overset{\text{ER to KF}}{\Longrightarrow}$ `Sub - relationship`
in: Subtyping(Child:Relationship, Parent:Relationship)
out: Relationship → `Relationship`
out: Relationship → `Relationship`
out: `Subsumption(Child:Relationship, Parent:Relationship)`

(**ER-1SA**) `Sub - relationship` $\overset{\text{KF to ER}}{\Longrightarrow}$ Subtyping of relationship
in: `Subsumption(Child:Relationship, Parent:Relationship)`
out: `Relationship` → Relationship
out: `Relationship` → Relationship
out: Subtyping(Child:Relationship, Parent:Relationship)

1.3 *ORM 2/KF Rules*

(**ORM2-O1**) Object type $\overset{\text{ORM 2 to KF}}{\Longrightarrow}$ `Object Type`
in: Object type

out: `Object Type`

(**ORM2-1O**) `Object Type` $\overset{\text{KF to ORM 2}}{\Longrightarrow}$ Object type
in: `Object Type`
out: Object type

(**ORM2-R1**) Role $\overset{\text{ORM 2 to KF}}{\Longrightarrow}$ `Role`
in: Role
in: FactType(Role: Object type, _)
in: FrequencyConstraint(Role, min, max)
out: Role → `Role`
out: `ORM2-O1(Object type)`
out: `ORM2-A1(FactType)`
out: `ORM2-MC1-1/ORM2-MC1-2(FrequencyConstraint)`
out: `Role(Relationship, Object type, CardinalityConstraint)`

(**ORM2-1R**) `Role` $\overset{\text{KF to ORM 2}}{\Longrightarrow}$ Role
in: `Role(Relationship, Object type, CardinalityConstraint)`
in: `Relationship(Role: Object type, _)`
in: `CardinalityConstraint(Role, min, max)`
out: `Role` → Role
out: ORM2-1O(Object type)
out: ORM2-1A(Relationship)
out: ORM2-1MC-1/ORM2-1MC-2(CardinalityConstraint)

(**ORM2-M1**) Mandatory $\overset{\text{ORM 2 to KF}}{\Longrightarrow}$ `Mandatory`
in: Mandatory(Role)
out: `Mandatory(ORM2-R1(Role)))`

(**ORM2-1M**) `Mandatory` $\overset{\text{KF to ORM 2}}{\Longrightarrow}$ Mandatory
in: `Mandatory(Role)`
out: Mandatory(ORM2-1R(Role)))

(**ORM2-S1**) Subtype $\overset{\text{ORM 2 to KF}}{\Longrightarrow}$ `Subsumption`
in: Subtype(Child: Object type, Parent: Object type)
out: `Subsumption(Child: ORM2-O1(Object type), Parent: ORM2-O1(Object type))`

(**ORM2-1S**) `Subsumption` $\overset{\text{KF to ORM 2}}{\Longrightarrow}$ Subtype
in: `Subsumption(Child: Object type, Parent: Object type)`
out: Subtype(Child: ORM2-1O(Object type), Parent: ORM2-1O(Object type))

(**ORM2-C1**) Total $\overset{\text{ORM 2 to KF}}{\Longrightarrow}$ `Completeness constraint`
in: Subtype(Child-1: Object type, Parent: Object type)
in: Subtype(Child-2: Object type, Parent: Object type)
in: Total(Child-1, Child-2)
out: `ORM2-S1(Subtype)`
out: `ORM2-S1(Subtype)`
out: `CompletenessConstraint(Child-1: Object type, Child-2: Object type)`

(**ORM2-1C**) `Completeness constraint` $\overset{\text{KF to ORM 2}}{\Longrightarrow}$ Total
in: `Subsumption(Child-1: Object type, Parent: Object type)`
in: `Subsumption(Child-2: Object type, Parent: Object type)`
in: `CompletenessConstraint(Child-1, Child-2)`
out: ORM2-1S(Subsumption)
out: ORM2-1S(Subsumption)
out: Total(Child-1: Object type, Child-2: Object type)

(**ORM2-D1**) Exclusive $\overset{\text{ORM 2 to KF}}{\Longrightarrow}$ `Disjoint object type`
in: Subtype(Child-1: Object type, Parent: Object type)
in: Subtype(Child-2: Object type, Parent: Object type)
in: Exclusive(Child-1, Child-2)
out: `ORM2-S1(Subtype)`
out: `ORM2-S1(Subtype)`
out: `DisjointObjectType(Child-1: Object type, Child-2: Object type)`

(**ORM2-1D**) `Disjoint object type` $\overset{\text{KF to ORM 2}}{\Longrightarrow}$ Exclusive
in: `Subsumption(Child-1: Object type, Parent: Object type)`
in: `Subsumption(Child-2: Object type, Parent: Object type)`
in: `DisjointObjectType(Child-1, Child-2)`
out: ORM2-1S(Subsumption)
out: ORM2-1S(Subsumption)
out: Exclusive(Child-1: Object type, Child-2: Object type)

(**ORM2-VT1**) Value type $\overset{\text{ORM 2 to KF}}{\Longrightarrow}$ `Value type`
in: Value type
in: MappedTo(Value type, Data type)
out: `ORM2-DT1(Data type)`
out: MappedTo → `MappedTo`
out: Value type → `Value type`
out: `MappedTo(Value type, Data type)`

(**ORM2-1VT**) `Value type` $\overset{\text{KF to ORM 2}}{\Longrightarrow}$ Value type
in: `Value type` ∧ `MappedTo(Value type, Data type)`
out: ORM2-1DT(Data type)
out: `MappedTo` → MappedTo
out: `Value type` → Value type
out: MappedTo(Value type, Data type)

(**ORM2-A1**) Fact type $\overset{\text{ORM 2 to KF}}{\Longrightarrow}$ `Relationship`
in: Fact type(Role: Object type, Role: Object type)
out: Fact type → `Relationship`
out: `Relationship(ORM2-R1(Role):ORM2-O1(Object type), ORM2-R1(Role):ORM2-O1(Object type))`

(**ORM2-1A**) `Relationship` $\overset{\text{KF to ORM 2}}{\Longrightarrow}$ Fact type
in: `Relationship(Role: Object type, Role: Object type)`
out: `Relationship` → Fact type
out: FactType(ORM2-1R(Role):ORM2-1O(Object type), ORM2-1R(Role):ORM2-1O(Object type))

(**ORM2-MC1-1**) Frequency constraint $\overset{\text{ORM 2 to KF}}{\Longrightarrow}$ `Object type cardinality constraint`
in: FrequencyConstraint(Role, min, max), // min=0 o min=1
out: `ObjectTypeCardinalityConstraint(ORM2-R1(Role), 0, max)`

(**ORM2-MC1-2**) Frequency constraint $\overset{\text{ORM 2 to KF}}{\Longrightarrow}$ `Object type cardinality constraint`
in: FrequencyConstraint(Role, min, max), Mandatory(Role) $min > 1$
out: `ObjectTypeCardinalityConstraint(ORM2-R1(Role), min,max)`

(**ORM2-1MC-1**) `Object type cardinality constraint` $\xRightarrow{\text{KF to ORM 2}}$ Frequency constraint
in: `ObjectTypeCardinalityConstraint(Role, 0, max)`
out: FrequencyConstraint(ORM2-1R(Role), 0, max)

(**ORM2-1MC-2**) `Object type cardinality constraint` $\xRightarrow{\text{KF to ORM 2}}$ Frequency constraint
in: `ObjectTypeCardinalityConstraint(Role, min, max)` $min \leq 1$
out: `FrequencyConstraint(ORM2-1R(Role), min, max)`

(**ORM2-SA1**) Subset constraint on fact type $\xRightarrow{\text{ORM 2 to KF}}$ `Sub - relationship`
in: Subset(Child:Fact type, Parent:Fact type)
out: Fact type → `Relationship`
out: Fact type → `Relationship`
out: `Subsumption(Child:Relationship, Parent:Relationship)`

(**ORM2-1SA**) `Sub - relationship` $\xRightarrow{\text{KF to ORM 2}}$ Subset constraint on fact type
in: `Subsumption(Child:Relationship, Parent:Relationship)`
out: `Relationship` → Fact type
out: `Relationship` → Fact type
out: Subset(Child:Fact type, Parent:Fact type)

1.4 *MM/MM Rules*

(**MM-ATT-VT**) `Attribute` $\xRightarrow{\text{KF}}$ `Value type`
in: `Attribute(Object type, Data type)`
out: `Data type`
out: `Role`
out: `Relationship`
out: `MappedTo`
out: `Attribute` → `Value type`
out: `Relationship(Role:Object type, Role:Value type)`
out: `MappedTo(Value type, Data type)`

(**MM-VT-ATT**) `Value type` $\xRightarrow{\text{KF}}$ `Attribute`
in: `Value type` ∧ `MappedTo(Value type, Data type)`
out: `Data type`
out: `Object type`

out: `Attribute(Object type, Data type)`

1.5 *DL/KF Embedding Rules*

(**KO1**) Atomic Concept $C_i \xrightarrow{\text{DL to KF}}$ Object type
out: $C_i \rightarrow$ `Object type`

(**1KS**) $C_i \sqsubseteq C_j \xrightarrow{\text{DL to KF}}$ `Subsumption`
in: $C_i \sqsubseteq C_j$ *// A, B atomic concepts*
out: $C_i \rightarrow$ `Object type`
out: $C_j \rightarrow$ `Object type`
out: `Subsumption(Child: Object type, Parent: Object type)`

(**OK2**) $A \equiv B \xrightarrow{\text{DL to KF}}$ `KF`
in: normalised $A \equiv B$
out: $A \rightarrow$ `Object type`
out: $B \rightarrow$ `Object type`
out: `Subsumption(Child:` A`, Parent:` B`)`
out: `Subsumption(Child:` B`, Parent:` A`)`

1.6 *KF/DL Embedding Rules*

1.7 *KF/CNL (en) Rules*

(**KF-CNL-1O**) `Object Type` $\xrightarrow{\text{KF to CNL(en)}}$ text
in: `Object Type`
out: [Object Type] is an Object Type

(**UML-CNL-1R**) `Role` $\xrightarrow{\text{KF to CNL(en)}}$ text
in: `Role(Relationship, Object type, CardinalityConstraint)`
out: [Role] is a Role

(**UML-CNL-1DT**) `Data type` $\xrightarrow{\text{KF to CNL(en)}}$ text
in: `Data type`

Table 1 KF/DL Embedding Rules

KF	DL
`Object type O`	Concept O
`Role` $r_{endConcept}$	Role $r_{endConcept}$
`Data Type D`	Concept D
`Attribute A of data type DT for the object type O`	Role A $\exists A.\top \sqsubseteq O$ $\top \sqsubseteq \forall A.DT$ $O \sqsubseteq \leq 1\ A.DT$
`Binary Relationship R between O1 and O2`	Concept R $R \sqsubseteq \exists r_{o1}.O1$ $R \sqsubseteq \exists r_{o2}.O2$
`Object type O cardinality constraint:`	
(1)Range `(min, max)`	$O \sqsubseteq\ (\geq \min\ r_o^-.R) \sqcap (\leq \max\ r_o^-.R)$
(2)Range `(.. max)`	$O \sqsubseteq\ (\leq \max\ r_o^-.R)$
(3)Range`(min ..)`	$O \sqsubseteq\ (\geq \min\ r_o^-.R)$
`Mandatory role` r_o	$O \sqsubseteq \geq 1\ r_o^-$
`Object type subsumption`	$OSub \sqsubseteq OSup$
`Disjoint object type subsumption`	$O_1 \sqsubseteq OSup$ $O_2 \sqsubseteq OSup$ $\vdots$ $O_n \sqsubseteq OSup$ $O_i \sqsubseteq \prod_{j=i+1}^{n} \neg O_j,\ \text{for } i = 1, \ldots, n-1$
`Completeness object type subsumption`	$O_1 \sqsubseteq OSup$ $O_2 \sqsubseteq OSup$ $\ldots$ $O_n \sqsubseteq OSup$ $OSup \sqsubseteq O_1 \sqcup O_2 \sqcup \ldots \sqcup O_n$
`Relationship Subsumption`	$RChild \sqsubseteq RParent$

out: [Data type] is a Data Type

(**UML-CNL-1M**) `Mandatory` $\xrightarrow{\text{KF to CNL(en)}}$ text
in: `Mandatory(Role)`
out: [Role] is Mandatory

(**UML-CNL-1S**) `Subsumption` $\xrightarrow{\text{KF to CNL(en)}}$ text
in: `Subsumption(Child: Object type, Parent: Object type)`
out: Each [Child] is a [Parent]

(**UML-CNL-1C**) `Completeness constraint` $\xrightarrow{\text{KF to CNL(en)}}$ text
in: `Subsumption(Child-1: Object type, Parent: Object type)`
in: `Subsumption(Child-2: Object type, Parent: Object type)`

in: `CompletenessConstraint(Child-1, Child-2)`
out: UML-CNL-1S(Subsumption)
out: UML-CNL-1S(Subsumption)
out: [Child-1] and [Child-2] cover [Parent]

(**UML-CNL-1D**) `Disjoint object type` $\overset{\text{KF to CNL(en)}}{\Longrightarrow}$ text
in: `Subsumption(Child-1: Object type, Parent: Object type)`
in: `Subsumption(Child-2: Object type, Parent: Object type)`
in: `DisjointObjectType(Child-1, Child-2)`
out: UML-CNL-1S(Subsumption)
out: UML-CNL-1S(Subsumption)
out: [Child-1] and [Child-2] are disjoint from each other.

(**UML-CNL-1ATT**) `Attribute` $\overset{\text{KF to CNL(en)}}{\Longrightarrow}$ text
in: `Attribute(Object type, Data type)`
out: UML-CNL-1DT(Data type)
out: [Attribute] is an attribute with data type [Data type]
out: [Object type] has attribute [Attribute]

(**UML-CNL-1A**) `Relationship` $\overset{\text{KF to CNL(en)}}{\Longrightarrow}$ text
in: `Relationship(Role-1: Object type-1, Role-2: Object type-2)`
out: UML-CNL-1O(Object type-1)
out: UML-CNL-1O(Object type-2))
out: [Relationship] is a relationship between [Object type-1] and [Object type-2]
out: [Role-1] is a Role in the relationship [Relationship]
out: [Role-2] is a Role in the relationship [Relationship]

(**UML-CNL-1MC**) `Object type cardinality constraint` $\overset{\text{KF to CNL(en)}}{\Longrightarrow}$ text
in: `Role(Relationship, Object type-1, ObjectTypeCardinalityConstraint)`
in: `Relationship(Object type-1, Object type-2)`
in: `ObjectTypeCardinalityConstraint(Role, min, max)`
out: Each [Object type-1] [Role] at least [min] [Object type-2] and at most [max] [Object type-2]

(**UML-CNL-1SA**) `Sub - relationship` $\overset{\text{KF to CNL(en)}}{\Longrightarrow}$ text
in: `Subsumption(Child:Relationship, Parent:Relationship)`
out: Each [Child] is a [Parent]
