

---

## 附录 A. 2 工程计量(BIM) IFC 文件

SCHEMA IFC4;

TYPE IfcStrippedOptional = BOOLEAN;  
END\_TYPE;

TYPE IfcAbsorbedDoseMeasure = REAL;  
END\_TYPE;

TYPE IfcAccelerationMeasure = REAL;  
END\_TYPE;

TYPE IfcAmountOfSubstanceMeasure = REAL;  
END\_TYPE;

TYPE IfcAngularVelocityMeasure = REAL;  
END\_TYPE;

TYPE IfcArcIndex = LIST [3:3] OF IfcPositiveInteger;  
END\_TYPE;

TYPE IfcAreaDensityMeasure = REAL;  
END\_TYPE;

TYPE IfcAreaMeasure = REAL;  
END\_TYPE;

TYPE IfcBinary = BINARY;  
END\_TYPE;

TYPE IfcBoolean = BOOLEAN;  
END\_TYPE;

TYPE IfcBoxAlignment = IfcLabel;  
WHERE  
    WR1 : SELF IN ['top-left', 'top-middle', 'top-right', 'middle-left', 'center',  
    'middle-right', 'bottom-left', 'bottom-middle', 'bottom-right'];  
END\_TYPE;

TYPE IfcCardinalPointReference = INTEGER;  
WHERE  
    GreaterThanZero : SELF > 0;  
END\_TYPE;

---

```

TYPE IfcComplexNumber = ARRAY [1:2] OF REAL;
END_TYPE;

TYPE IfcCompoundPlaneAngleMeasure = LIST [3:4] OF INTEGER;
WHERE
    MinutesInRange : ABS(SELF[2]) < 60;
    SecondsInRange : ABS(SELF[3]) < 60;
    MicrosecondsInRange : (SIZEOF(SELF) = 3) OR (ABS(SELF[4]) < 1000000);
    ConsistentSign : ((SELF[1] >= 0) AND (SELF[2] >= 0) AND (SELF[3] >= 0) AND
        ((SIZEOF(SELF) = 3) OR (SELF[4] >= 0)))
OR
    ((SELF[1] <= 0) AND (SELF[2] <= 0) AND (SELF[3] <= 0) AND ((SIZEOF(SELF) = 3) OR
        (SELF[4] <= 0)));
END_TYPE;

TYPE IfcContextDependentMeasure = REAL;
END_TYPE;

TYPE IfcCountMeasure = NUMBER;
END_TYPE;

TYPE IfcCurvatureMeasure = REAL;
END_TYPE;

TYPE IfcDate = STRING;
END_TYPE;

TYPE IfcDateTime = STRING;
END_TYPE;

TYPE IfcDayInMonthNumber = INTEGER;
WHERE
    ValidRange : {1 <= SELF <= 31};
END_TYPE;

TYPE IfcDayInWeekNumber = INTEGER;
WHERE
    ValidRange : {1 <= SELF <= 7};
END_TYPE;

TYPE IfcDescriptiveMeasure = STRING;
END_TYPE;

```

---

```

TYPE IfcDimensionCount = INTEGER;
WHERE
    WR1 : { 0 < SELF <= 3 };
END_TYPE;

TYPE IfcDoseEquivalentMeasure = REAL;
END_TYPE;

TYPE IfcDuration = STRING;
END_TYPE;

TYPE IfcDynamicViscosityMeasure = REAL;
END_TYPE;

TYPE IfcElectricCapacitanceMeasure = REAL;
END_TYPE;

TYPE IfcElectricChargeMeasure = REAL;
END_TYPE;

TYPE IfcElectricConductanceMeasure = REAL;
END_TYPE;

TYPE IfcElectricCurrentMeasure = REAL;
END_TYPE;

TYPE IfcElectricResistanceMeasure = REAL;
END_TYPE;

TYPE IfcElectricVoltageMeasure = REAL;
END_TYPE;

TYPE IfcEnergyMeasure = REAL;
END_TYPE;

TYPE IfcFontStyle = STRING;
WHERE
    WR1 : SELF IN ['normal', 'italic', 'oblique'];
END_TYPE;

TYPE IfcFontVariant = STRING;
WHERE
    WR1 : SELF IN ['normal', 'small-caps'];
END_TYPE;

```

---

```

TYPE IfcFontWeight = STRING;
  WHERE
    WR1 : SELF IN
    ['normal', 'small-caps', '100', '200', '300', '400', '500', '600', '700', '800', '900'];
END_TYPE;

TYPE IfcForceMeasure = REAL;
END_TYPE;

TYPE IfcFrequencyMeasure = REAL;
END_TYPE;

TYPE IfcGloballyUniqueId = STRING(22) FIXED;
END_TYPE;

TYPE IfcHeatFluxDensityMeasure = REAL;
END_TYPE;

TYPE IfcHeatingValueMeasure = REAL;
  WHERE
    WR1 : SELF > 0.;
END_TYPE;

TYPE IfcIdentifier = STRING(255);
END_TYPE;

TYPE IfcIlluminanceMeasure = REAL;
END_TYPE;

TYPE IfcInductanceMeasure = REAL;
END_TYPE;

TYPE IfcInteger = INTEGER;
END_TYPE;

TYPE IfcIntegerCountRateMeasure = INTEGER;
END_TYPE;

TYPE IfcIonConcentrationMeasure = REAL;
END_TYPE;

TYPE IfcIsothermalMoistureCapacityMeasure = REAL;
END_TYPE;

```

---

```
TYPE IfcKinematicViscosityMeasure = REAL;
END_TYPE;

TYPE IfcLabel = STRING(255);
END_TYPE;

TYPE IfcLanguageId = IfcIdentifier;
END_TYPE;

TYPE IfcLengthMeasure = REAL;
END_TYPE;

TYPE IfcLineIndex = LIST [2:?] OF IfcPositiveInteger;
END_TYPE;

TYPE IfcLinearForceMeasure = REAL;
END_TYPE;

TYPE IfcLinearMomentMeasure = REAL;
END_TYPE;

TYPE IfcLinearStiffnessMeasure = REAL;
END_TYPE;

TYPE IfcLinearVelocityMeasure = REAL;
END_TYPE;

TYPE IfcLogical = LOGICAL;
END_TYPE;

TYPE IfcLuminousFluxMeasure = REAL;
END_TYPE;

TYPE IfcLuminousIntensityDistributionMeasure = REAL;
END_TYPE;

TYPE IfcLuminousIntensityMeasure = REAL;
END_TYPE;

TYPE IfcMagneticFluxDensityMeasure = REAL;
END_TYPE;

TYPE IfcMagneticFluxMeasure = REAL;
```

---

```

END_TYPE;

TYPE IfcMassDensityMeasure = REAL;
END_TYPE;

TYPE IfcMassFlowRateMeasure = REAL;
END_TYPE;

TYPE IfcMassMeasure = REAL;
END_TYPE;

TYPE IfcMassPerLengthMeasure = REAL;
END_TYPE;

TYPE IfcModulusOfElasticityMeasure = REAL;
END_TYPE;

TYPE IfcModulusOfLinearSubgradeReactionMeasure = REAL;
END_TYPE;

TYPE IfcModulusOfRotationalSubgradeReactionMeasure = REAL;
END_TYPE;

TYPE IfcModulusOfSubgradeReactionMeasure = REAL;
END_TYPE;

TYPE IfcMoistureDiffusivityMeasure = REAL;
END_TYPE;

TYPE IfcMolecularWeightMeasure = REAL;
END_TYPE;

TYPE IfcMomentOfInertiaMeasure = REAL;
END_TYPE;

TYPE IfcMonetaryMeasure = REAL;
END_TYPE;

TYPE IfcMonthInYearNumber = INTEGER;
  WHERE
    ValidRange : {1 <= SELF <= 12};
END_TYPE;

TYPE IfcNonNegativeLengthMeasure = IfcLengthMeasure;

```

---

```
WHERE
    NotNegative : SELF >= 0. ;
END_TYPE;

TYPE IfcNormalisedRatioMeasure = IfcRatioMeasure;
WHERE
    WR1 : {0.0 <= SELF <= 1.0};
END_TYPE;

TYPE IfcNumericMeasure = NUMBER;
END_TYPE;

TYPE IfcPHMeasure = REAL;
WHERE
    WR21 : {0.0 <= SELF <= 14.0};
END_TYPE;

TYPE IfcParameterValue = REAL;
END_TYPE;

TYPE IfcPlanarForceMeasure = REAL;
END_TYPE;

TYPE IfcPlaneAngleMeasure = REAL;
END_TYPE;

TYPE IfcPositiveInteger = IfcInteger;
WHERE
    WR1 : SELF > 0;
END_TYPE;

TYPE IfcPositiveLengthMeasure = IfcLengthMeasure;
WHERE
    WR1 : SELF > 0. ;
END_TYPE;

TYPE IfcPositivePlaneAngleMeasure = IfcPlaneAngleMeasure;
WHERE
    WR1 : SELF > 0. ;
END_TYPE;

TYPE IfcPositiveRatioMeasure = IfcRatioMeasure;
WHERE
    WR1 : SELF > 0. ;
```

---

```
END_TYPE;

TYPE IfcPowerMeasure = REAL;
END_TYPE;

TYPE IfcPresentableText = STRING;
END_TYPE;

TYPE IfcPressureMeasure = REAL;
END_TYPE;

TYPE IfcPropertySetDefinitionSet = SET [1:?] OF IfcPropertySetDefinition;
END_TYPE;

TYPE IfcRadioActivityMeasure = REAL;
END_TYPE;

TYPE IfcRatioMeasure = REAL;
END_TYPE;

TYPE IfcReal = REAL;
END_TYPE;

TYPE IfcRotationalFrequencyMeasure = REAL;
END_TYPE;

TYPE IfcRotationalMassMeasure = REAL;
END_TYPE;

TYPE IfcRotationalStiffnessMeasure = REAL;
END_TYPE;

TYPE IfcSectionModulusMeasure = REAL;
END_TYPE;

TYPE IfcSectionalAreaIntegralMeasure = REAL;
END_TYPE;

TYPE IfcShearModulusMeasure = REAL;
END_TYPE;

TYPE IfcSolidAngleMeasure = REAL;
END_TYPE;
```



---

```

TYPE IfcSoundPowerLevelMeasure = REAL;
END_TYPE;

TYPE IfcSoundPowerMeasure = REAL;
END_TYPE;

TYPE IfcSoundPressureLevelMeasure = REAL;
END_TYPE;

TYPE IfcSoundPressureMeasure = REAL;
END_TYPE;

TYPE IfcSpecificHeatCapacityMeasure = REAL;
END_TYPE;

TYPE IfcSpecularExponent = REAL;
END_TYPE;

TYPE IfcSpecularRoughness = REAL;
WHERE
    WR1 : {0.0 <= SELF <= 1.0};
END_TYPE;

TYPE IfcTemperatureGradientMeasure = REAL;
END_TYPE;

TYPE IfcTemperatureRateOfChangeMeasure = REAL;
END_TYPE;

TYPE IfcText = STRING;
END_TYPE;

TYPE IfcTextAlignment = STRING;
WHERE
    WR1 : SELF IN ['left', 'right', 'center', 'justify'];
END_TYPE;

TYPE IfcTextDecoration = STRING;
WHERE
    WR1 : SELF IN ['none', 'underline', 'overline', 'line-through', 'blink'];
END_TYPE;

TYPE IfcTextFontName = STRING;
END_TYPE;

```

---

```
TYPE IfcTextTransformation = STRING;
  WHERE
    WR1 : SELF IN ['capitalize', 'uppercase', 'lowercase', 'none'];
END_TYPE;

TYPE IfcThermalAdmittanceMeasure = REAL;
END_TYPE;

TYPE IfcThermalConductivityMeasure = REAL;
END_TYPE;

TYPE IfcThermalExpansionCoefficientMeasure = REAL;
END_TYPE;

TYPE IfcThermalResistanceMeasure = REAL;
END_TYPE;

TYPE IfcThermalTransmittanceMeasure = REAL;
END_TYPE;

TYPE IfcThermodynamicTemperatureMeasure = REAL;
END_TYPE;

TYPE IfcTime = STRING;
END_TYPE;

TYPE IfcTimeMeasure = REAL;
END_TYPE;

TYPE IfcTimeStamp = INTEGER;
END_TYPE;

TYPE IfcTorqueMeasure = REAL;
END_TYPE;

TYPE IfcURIReference = STRING;
END_TYPE;

TYPE IfcVaporPermeabilityMeasure = REAL;
END_TYPE;

TYPE IfcVolumeMeasure = REAL;
END_TYPE;
```

---

```
TYPE IfcVolumetricFlowRateMeasure = REAL;  
END_TYPE;
```

```
TYPE IfcWarpingConstantMeasure = REAL;  
END_TYPE;
```

```
TYPE IfcWarpingMomentMeasure = REAL;  
END_TYPE;
```

```
TYPE IfcActionRequestTypeEnum = ENUMERATION OF  
    (EMAIL  
    , FAX  
    , PHONE  
    , POST  
    , VERBAL  
    , USERDEFINED  
    , NOTDEFINED);  
END_TYPE;
```

```
TYPE IfcActionSourceTypeEnum = ENUMERATION OF  
    (DEAD_LOAD_G  
    , COMPLETION_G1  
    , LIVE_LOAD_Q  
    , SNOW_S  
    , WIND_W  
    , PRESTRESSING_P  
    , SETTLEMENT_U  
    , TEMPERATURE_T  
    , EARTHQUAKE_E  
    , FIRE  
    , IMPULSE  
    , IMPACT  
    , TRANSPORT  
    , ERECTION  
    , PROPPING  
    , SYSTEM_IMPERFECTION  
    , SHRINKAGE  
    , CREEP  
    , LACK_OF_FIT  
    , BUOYANCY  
    , ICE  
    , CURRENT  
    , WAVE
```

---

```
, RAIN
, BRAKES
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcActionTypeEnum = ENUMERATION OF
(PERMANENT_G
, VARIABLE_Q
, EXTRAORDINARY_A
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcActuatorTypeEnum = ENUMERATION OF
(ELECTRICACTUATOR
, HANDOPERATEDACTUATOR
, HYDRAULICACTUATOR
, PNEUMATICACTUATOR
, THERMOSTATICACTUATOR
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcAddressTypeEnum = ENUMERATION OF
(OFFICE
, SITE
, HOME
, DISTRIBUTIONPOINT
, USERDEFINED);
END_TYPE;
```

```
TYPE IfcAirTerminalBoxTypeEnum = ENUMERATION OF
(CONSTANTFLOW
, VARIABLEFLOWPRESSUREDEPENDANT
, VARIABLEFLOWPRESSUREINDEPENDANT
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcAirTerminalTypeEnum = ENUMERATION OF
(DIFFUSER
, GRILLE
, LOUVRE
```

---

```
, REGISTER
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcAirToAirHeatRecoveryTypeEnum = ENUMERATION OF
(FIXEDPLATECOUNTERFLOWEXCHANGER
, FIXEDPLATECROSSFLOWEXCHANGER
, FIXEDPLATEPARALLELFLOWEXCHANGER
, ROTARYWHEEL
, RUNAROUNDCOILLOOP
, HEATPIPE
, TWINTOWERENTHALPYRECOVERYLOOPS
, THERMOSIPHONSEALEDTUBEHEATEXCHANGERS
, THERMOSIPHONCOILTYPEHEATEXCHANGERS
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcAlarmTypeEnum = ENUMERATION OF
(BELL
, BREAKGLASSBUTTON
, LIGHT
, MANUALPULLBOX
, SIREN
, WHISTLE
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcAnalysisModelTypeEnum = ENUMERATION OF
(IN_PLANE_LOADING_2D
, OUT_PLANE_LOADING_2D
, LOADING_3D
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcAnalysisTheoryTypeEnum = ENUMERATION OF
(FIRST_ORDER_THEORY
, SECOND_ORDER_THEORY
, THIRD_ORDER_THEORY
, FULL_NONLINEAR_THEORY
, USERDEFINED
```

---

```
    , NOTDEFINED);  
END_TYPE;
```

```
TYPE IfcArithmeticOperatorEnum = ENUMERATION OF  
    (ADD  
    , DIVIDE  
    , MULTIPLY  
    , SUBTRACT);  
END_TYPE;
```

```
TYPE IfcAssemblyPlaceEnum = ENUMERATION OF  
    (SITE  
    , FACTORY  
    , NOTDEFINED);  
END_TYPE;
```

```
TYPE IfcAudioVisualApplianceTypeEnum = ENUMERATION OF  
    (AMPLIFIER  
    , CAMERA  
    , DISPLAY  
    , MICROPHONE  
    , PLAYER  
    , PROJECTOR  
    , RECEIVER  
    , SPEAKER  
    , SWITCHER  
    , TELEPHONE  
    , TUNER  
    , USERDEFINED  
    , NOTDEFINED);  
END_TYPE;
```

```
TYPE IfcBSplineCurveForm = ENUMERATION OF  
    (POLYLINE_FORM  
    , CIRCULAR_ARC  
    , ELLIPTIC_ARC  
    , PARABOLIC_ARC  
    , HYPERBOLIC_ARC  
    , UNSPECIFIED);  
END_TYPE;
```

```
TYPE IfcBSplineSurfaceForm = ENUMERATION OF  
    (PLANE_SURF  
    , CYLINDRICAL_SURF
```

---

```
, CONICAL_SURF
, SPHERICAL_SURF
, TOROIDAL_SURF
, SURF_OF_REVOLUTION
, RULED_SURF
, GENERALISED_CONE
, QUADRIC_SURF
, SURF_OF_LINEAR_EXTRUSION
, UNSPECIFIED);
END_TYPE;
```

```
TYPE IfcBeamTypeEnum = ENUMERATION OF
    (BEAM
    , JOIST
    , HOLLOWCORE
    , LINTEL
    , SPANDREL
    , T_BEAM
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcBenchmarkEnum = ENUMERATION OF
    (GREATERTHAN
    , GREATERTHANOREQUALTO
    , LESSTHAN
    , LESSTHANOREQUALTO
    , EQUALTO
    , NOTEQUALTO
    , INCLUDES
    , NOTINCLUDES
    , INCLUDEDIN
    , NOTINCLUDEDIN);
END_TYPE;
```

```
TYPE IfcBoilerTypeEnum = ENUMERATION OF
    (WATER
    , STEAM
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcBooleanOperator = ENUMERATION OF
    (UNION
```

---

```
    , INTERSECTION
    , DIFFERENCE);
END_TYPE;
```

```
TYPE IfcBuildingElementPartTypeEnum = ENUMERATION OF
    (INSULATION
    , PRECASTPANEL
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcBuildingElementProxyTypeEnum = ENUMERATION OF
    (COMPLEX
    , ELEMENT
    , PARTIAL
    , PROVISIONFORVOID
    , PROVISIONFORSPACE
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcBuildingSystemTypeEnum = ENUMERATION OF
    (FENESTRATION
    , FOUNDATION
    , LOADBEARING
    , OUTERSHELL
    , SHADING
    , TRANSPORT
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcBurnerTypeEnum = ENUMERATION OF
    (USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcCableCarrierFittingTypeEnum = ENUMERATION OF
    (BEND
    , CROSS
    , REDUCER
    , TEE
    , USERDEFINED
    , NOTDEFINED);
```



---

END\_TYPE;

TYPE IfcCableCarrierSegmentTypeEnum = ENUMERATION OF  
    (CABLELADDERSEGMENT  
    , CABLETRAYSEGMENT  
    , CABLETRUNKINGSEGMENT  
    , CONDUITSEGMENT  
    , USERDEFINED  
    , NOTDEFINED);  
END\_TYPE;

TYPE IfcCableFittingTypeEnum = ENUMERATION OF  
    (CONNECTOR  
    , ENTRY  
    , EXIT  
    , JUNCTION  
    , TRANSITION  
    , USERDEFINED  
    , NOTDEFINED);  
END\_TYPE;

TYPE IfcCableSegmentTypeEnum = ENUMERATION OF  
    (BUSBARSEGMENT  
    , CABLESEGMENT  
    , CONDUCTORSEGMENT  
    , CORESEGMENT  
    , USERDEFINED  
    , NOTDEFINED);  
END\_TYPE;

TYPE IfcChangeActionEnum = ENUMERATION OF  
    (NOCHANGE  
    , MODIFIED  
    , ADDED  
    , DELETED  
    , NOTDEFINED);  
END\_TYPE;

TYPE IfcChillerTypeEnum = ENUMERATION OF  
    (AIRCOOLED  
    , WATERCOOLED  
    , HEATRECOVERY  
    , USERDEFINED  
    , NOTDEFINED);

---

END\_TYPE;

TYPE IfcChimneyTypeEnum = ENUMERATION OF  
    (USERDEFINED  
    , NOTDEFINED);  
END\_TYPE;

TYPE IfcCoilTypeEnum = ENUMERATION OF  
    (DXCOOLINGCOIL  
    , ELECTRICHEATINGCOIL  
    , GASHEATINGCOIL  
    , HYDRONICCOIL  
    , STEAMHEATINGCOIL  
    , WATERCOOLINGCOIL  
    , WATERHEATINGCOIL  
    , USERDEFINED  
    , NOTDEFINED);  
END\_TYPE;

TYPE IfcColumnTypeEnum = ENUMERATION OF  
    (COLUMN  
    , PILASTER  
    , USERDEFINED  
    , NOTDEFINED);  
END\_TYPE;

TYPE IfcCommunicationsApplianceTypeEnum = ENUMERATION OF  
    (ANTENNA  
    , COMPUTER  
    , FAX  
    , GATEWAY  
    , MODEM  
    , NETWORKAPPLIANCE  
    , NETWORKBRIDGE  
    , NETWORKHUB  
    , PRINTER  
    , REPEATER  
    , ROUTER  
    , SCANNER  
    , USERDEFINED  
    , NOTDEFINED);  
END\_TYPE;

TYPE IfcComplexPropertyTemplateTypeEnum = ENUMERATION OF

---

```

        (P_COMPLEX
        , Q_COMPLEX) ;
END_TYPE;

TYPE IfcCompressorTypeEnum = ENUMERATION OF
    (DYNAMIC
    , RECIPROCATING
    , ROTARY
    , SCROLL
    , TROCHOIDAL
    , SINGLESTAGE
    , BOOSTER
    , OPENTYPE
    , HERMETIC
    , SEMIHERMETIC
    , WELDEDSHELLHERMETIC
    , ROLLINGPISTON
    , ROTARYVANE
    , SINGLESCREW
    , TWINSCREW
    , USERDEFINED
    , NOTDEFINED) ;
END_TYPE;

TYPE IfcCondenserTypeEnum = ENUMERATION OF
    (AIRCOOLED
    , EVAPORATIVECOOLED
    , WATERCOOLED
    , WATERCOOLDBRAZEDPLATE
    , WATERCOOLEDSHELLCOIL
    , WATERCOOLEDSHELLTUBE
    , WATERCOOLEDTUBEINTUBE
    , USERDEFINED
    , NOTDEFINED) ;
END_TYPE;

TYPE IfcConnectionTypeEnum = ENUMERATION OF
    (ATPATH
    , ATSTART
    , ATEND
    , NOTDEFINED) ;
END_TYPE;

TYPE IfcConstraintEnum = ENUMERATION OF

```

---

```
(HARD
, SOFT
, ADVISORY
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcConstructionEquipmentResourceTypeEnum = ENUMERATION OF
(DEMOLISHING
, EARTHMOVING
, ERECTING
, HEATING
, LIGHTING
, PAVING
, PUMPING
, TRANSPORTING
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcConstructionMaterialResourceTypeEnum = ENUMERATION OF
(AGGREGATES
, CONCRETE
, DRYWALL
, FUEL
, GYPSUM
, MASONRY
, METAL
, PLASTIC
, WOOD
, NOTDEFINED
, USERDEFINED);
END_TYPE;
```

```
TYPE IfcConstructionProductResourceTypeEnum = ENUMERATION OF
(ASSEMBLY
, FORMWORK
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcControllerTypeEnum = ENUMERATION OF
(FLOATING
, PROGRAMMABLE
```

---

```

        , PROPORTIONAL
        , MULTIPOSITION
        , TWOPOSITION
        , USERDEFINED
        , NOTDEFINED);
END_TYPE;

TYPE IfcCooledBeamTypeEnum = ENUMERATION OF
    (ACTIVE
    , PASSIVE
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;

TYPE IfcCoolingTowerTypeEnum = ENUMERATION OF
    (NATURALDRAFT
    , MECHANICALINDUCEDDRAFT
    , MECHANICALFORCEDDRAFT
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;

TYPE IfcCostItemTypeEnum = ENUMERATION OF
    (USERDEFINED
    , NOTDEFINED);
END_TYPE;

TYPE IfcCostScheduleTypeEnum = ENUMERATION OF
    (BUDGET
    , COSTPLAN
    , ESTIMATE
    , TENDER
    , PRICEDBILLOFQUANTITIES
    , UNPRICEDBILLOFQUANTITIES
    , SCHEDULEOFRATES
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;

TYPE IfcCoveringTypeEnum = ENUMERATION OF
    (CEILING
    , FLOORING
    , CLADDING
    , ROOFING

```

---

```
, MOLDING
, SKIRTINGBOARD
, INSULATION
, MEMBRANE
, SLEEVING
, WRAPPING
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcCrewResourceTypeEnum = ENUMERATION OF
(OFFICE
, SITE
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcCurtainWallTypeEnum = ENUMERATION OF
(USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcCurveInterpolationEnum = ENUMERATION OF
(LINEAR
, LOG_LINEAR
, LOG_LOG
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcDamperTypeEnum = ENUMERATION OF
(BACKDRAFTDAMPER
, BALANCINGDAMPER
, BLASTDAMPER
, CONTROLDAMPER
, FIREDAMPER
, FIRESMOKEDAMPER
, FUMEHOODEXHAUST
, GRAVITYDAMPER
, GRAVITYRELIEFDAMPER
, RELIEFDAMPER
, SMOKEDAMPER
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

---

```
TYPE IfcDataOriginEnum = ENUMERATION OF
    (MEASURED
    , PREDICTED
    , SIMULATED
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcDerivedUnitEnum = ENUMERATION OF
    (ANGULARVELOCITYUNIT
    , AREADENSITYUNIT
    , COMPOUNDPLANEANGLEUNIT
    , DYNAMICVISCOSITYUNIT
    , HEATFLUXDENSITYUNIT
    , INTEGERCOUNTRATEUNIT
    , ISOTHERMALMOISTURECAPACITYUNIT
    , KINEMATICVISCOSITYUNIT
    , LINEARVELOCITYUNIT
    , MASSDENSITYUNIT
    , MASSFLOWRATEUNIT
    , MOISTUREDIFFFUSIVITYUNIT
    , MOLECULARWEIGHTUNIT
    , SPECIFICHEATCAPACITYUNIT
    , THERMALADMITTANCEUNIT
    , THERMALCONDUCTANCEUNIT
    , THERMALRESISTANCEUNIT
    , THERMALTRANSMITTANCEUNIT
    , VAPORPERMEABILITYUNIT
    , VOLUMETRICFLOWRATEUNIT
    , ROTATIONALFREQUENCYUNIT
    , TORQUEUNIT
    , MOMENTOFINERTIAUNIT
    , LINEARMOMENTUNIT
    , LINEARFORCEUNIT
    , PLANARFORCEUNIT
    , MODULUSOFELASTICITYUNIT
    , SHEARMODULUSUNIT
    , LINEARSTIFFNESSUNIT
    , ROTATIONALSTIFFNESSUNIT
    , MODULUSOF SUBGRADE REACTIONUNIT
    , ACCELERATIONUNIT
    , CURVATUREUNIT
    , HEATINGVALUEUNIT
```

---

```

    , IONCONCENTRATIONUNIT
    , LUMINOUSINTENSITYDISTRIBUTIONUNIT
    , MASSPERLENGTHUNIT
    , MODULUSOFFLINEARSUBGRADEREACTIONUNIT
    , MODULUSOFFROTATIONALSUBGRADEREACTIONUNIT
    , PHUNIT
    , ROTATIONALMASSUNIT
    , SECTIONAREAINTEGRALUNIT
    , SECTIONMODULUSUNIT
    , SOUNDPOWERLEVELUNIT
    , SOUNDPOWERUNIT
    , SOUNDPRESSURELEVELUNIT
    , SOUNDPRESSUREUNIT
    , TEMPERATUREGRADIENTUNIT
    , TEMPERATURERATEOFCHANGEUNIT
    , THERMALEXPANSIONCOEFFICIENTUNIT
    , WARPINGCONSTANTUNIT
    , WARPINGMOMENTUNIT
    , USERDEFINED) ;
END_TYPE;

```

```

TYPE IfcDirectionSenseEnum = ENUMERATION OF
    (POSITIVE
    , NEGATIVE) ;
END_TYPE;

```

```

TYPE IfcDiscreteAccessoryTypeEnum = ENUMERATION OF
    (ANCHORPLATE
    , BRACKET
    , SHOE
    , USERDEFINED
    , NOTDEFINED) ;
END_TYPE;

```

```

TYPE IfcDistributionChamberElementTypeEnum = ENUMERATION OF
    (FORMEDDUCT
    , INSPECTIONCHAMBER
    , INSPECTIONPIT
    , MANHOLE
    , METERCHAMBER
    , SUMP
    , TRENCH
    , VALVECHAMBER
    , USERDEFINED

```



---

```
    , NOTDEFINED) ;  
END_TYPE;
```

```
TYPE IfcDistributionPortTypeEnum = ENUMERATION OF  
    (CABLE  
    , CABLECARRIER  
    , DUCT  
    , PIPE  
    , USERDEFINED  
    , NOTDEFINED) ;  
END_TYPE;
```

```
TYPE IfcDistributionSystemEnum = ENUMERATION OF  
    (AIRCONDITIONING  
    , AUDIOVISUAL  
    , CHEMICAL  
    , CHILLEDWATER  
    , COMMUNICATION  
    , COMPRESSED AIR  
    , CONDENSERWATER  
    , CONTROL  
    , CONVEYING  
    , DATA  
    , DISPOSAL  
    , DOMESTIC COLDWATER  
    , DOMESTIC HOTWATER  
    , DRAINAGE  
    , EARTHING  
    , ELECTRICAL  
    , ELECTROACOUSTIC  
    , EXHAUST  
    , FIREPROTECTION  
    , FUEL  
    , GAS  
    , HAZARDOUS  
    , HEATING  
    , LIGHTING  
    , LIGHTNINGPROTECTION  
    , MUNICIPAL SOLIDWASTE  
    , OIL  
    , OPERATIONAL  
    , POWERGENERATION  
    , RAINWATER  
    , REFRIGERATION
```

---

```
, SECURITY
, SEWAGE
, SIGNAL
, STORMWATER
, TELEPHONE
, TV
, VACUUM
, VENT
, VENTILATION
, WASTEWATER
, WATERSUPPLY
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcDocumentConfidentialityEnum = ENUMERATION OF
(PUBLIC
, RESTRICTED
, CONFIDENTIAL
, PERSONAL
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcDocumentStatusEnum = ENUMERATION OF
(DRAFT
, FINALDRAFT
, FINAL
, REVISION
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcDoorPanelOperationEnum = ENUMERATION OF
(SWINGING
, DOUBLE_ACTING
, SLIDING
, FOLDING
, REVOLVING
, ROLLINGUP
, FIXEDPANEL
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

---

```
TYPE IfcDoorPanelPositionEnum = ENUMERATION OF
    (LEFT
    , MIDDLE
    , RIGHT
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcDoorStyleConstructionEnum = ENUMERATION OF
    (ALUMINIUM
    , HIGH_GRADE_STEEL
    , STEEL
    , WOOD
    , ALUMINIUM_WOOD
    , ALUMINIUM_PLASTIC
    , PLASTIC
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcDoorStyleOperationEnum = ENUMERATION OF
    (SINGLE_SWING_LEFT
    , SINGLE_SWING_RIGHT
    , DOUBLE_DOOR_SINGLE_SWING
    , DOUBLE_DOOR_SINGLE_SWING_OPPOSITE_LEFT
    , DOUBLE_DOOR_SINGLE_SWING_OPPOSITE_RIGHT
    , DOUBLE_SWING_LEFT
    , DOUBLE_SWING_RIGHT
    , DOUBLE_DOOR_DOUBLE_SWING
    , SLIDING_TO_LEFT
    , SLIDING_TO_RIGHT
    , DOUBLE_DOOR_SLIDING
    , FOLDING_TO_LEFT
    , FOLDING_TO_RIGHT
    , DOUBLE_DOOR_FOLDING
    , REVOLVING
    , ROLLINGUP
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcDoorTypeEnum = ENUMERATION OF
    (DOOR
    , GATE
    , TRAPDOOR
```

---

```

        , USERDEFINED
        , NOTDEFINED) ;
END_TYPE;

TYPE IfcDoorTypeEnum = ENUMERATION OF
    (SINGLE_SWING_LEFT
    , SINGLE_SWING_RIGHT
    , DOUBLE_DOOR_SINGLE_SWING
    , DOUBLE_DOOR_SINGLE_SWING_OPPOSITE_LEFT
    , DOUBLE_DOOR_SINGLE_SWING_OPPOSITE_RIGHT
    , DOUBLE_SWING_LEFT
    , DOUBLE_SWING_RIGHT
    , DOUBLE_DOOR_DOUBLE_SWING
    , SLIDING_TO_LEFT
    , SLIDING_TO_RIGHT
    , DOUBLE_DOOR_SLIDING
    , FOLDING_TO_LEFT
    , FOLDING_TO_RIGHT
    , DOUBLE_DOOR_FOLDING
    , REVOLVING
    , ROLLINGUP
    , SWING_FIXED_LEFT
    , SWING_FIXED_RIGHT
    , USERDEFINED
    , NOTDEFINED) ;
END_TYPE;

TYPE IfcDuctFittingTypeEnum = ENUMERATION OF
    (BEND
    , CONNECTOR
    , ENTRY
    , EXIT
    , JUNCTION
    , OBSTRUCTION
    , TRANSITION
    , USERDEFINED
    , NOTDEFINED) ;
END_TYPE;

TYPE IfcDuctSegmentTypeEnum = ENUMERATION OF
    (RIGIDSEGMENT
    , FLEXIBLESEGMENT
    , USERDEFINED
    , NOTDEFINED) ;

```

---

END\_TYPE;

TYPE IfcDuctSilencerTypeEnum = ENUMERATION OF  
    (FLATOVAL  
    , RECTANGULAR  
    , ROUND  
    , USERDEFINED  
    , NOTDEFINED);  
END\_TYPE;

TYPE IfcElectricApplianceTypeEnum = ENUMERATION OF  
    (DISHWASHER  
    , ELECTRICCOOKER  
    , FREESTANDINGELECTRICHEATER  
    , FREESTANDINGFAN  
    , FREESTANDINGWATERHEATER  
    , FREESTANDINGWATERCOOLER  
    , FREEZER  
    , FRIDGE\_FREEZER  
    , HANDDRYER  
    , KITCHENMACHINE  
    , MICROWAVE  
    , PHOTOCOPIER  
    , REFRIGERATOR  
    , TUMBLEDRYER  
    , VENDINGMACHINE  
    , WASHINGMACHINE  
    , USERDEFINED  
    , NOTDEFINED);  
END\_TYPE;

TYPE IfcElectricDistributionBoardTypeEnum = ENUMERATION OF  
    (CONSUMERUNIT  
    , DISTRIBUTIONBOARD  
    , MOTORCONTROLCENTRE  
    , SWITCHBOARD  
    , USERDEFINED  
    , NOTDEFINED);  
END\_TYPE;

TYPE IfcElectricFlowStorageDeviceTypeEnum = ENUMERATION OF  
    (BATTERY  
    , CAPACITORBANK  
    , HARMONICFILTER

---

```
    , INDUCTORBANK
    , UPS
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcElectricGeneratorTypeEnum = ENUMERATION OF
    (CHP
    , ENGINEGENERATOR
    , STANDALONE
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcElectricMotorTypeEnum = ENUMERATION OF
    (DC
    , INDUCTION
    , POLYPHASE
    , RELUCTANCESYNCHRONOUS
    , SYNCHRONOUS
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcElectricTimeControlTypeEnum = ENUMERATION OF
    (TIMECLOCK
    , TIMEDELAY
    , RELAY
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcElementAssemblyTypeEnum = ENUMERATION OF
    (ACCESSORY_ASSEMBLY
    , ARCH
    , BEAM_GRID
    , BRACED_FRAME
    , GIRDER
    , REINFORCEMENT_UNIT
    , RIGID_FRAME
    , SLAB_FIELD
    , TRUSS
    , USERDEFINED
    , NOTDEFINED);
```

---

END\_TYPE;

TYPE IfcElementCompositionEnum = ENUMERATION OF  
    (COMPLEX  
      , ELEMENT  
      , PARTIAL) ;  
END\_TYPE;

TYPE IfcEngineTypeEnum = ENUMERATION OF  
    (EXTERNALCOMBUSTION  
      , INTERNALCOMBUSTION  
      , USERDEFINED  
      , NOTDEFINED) ;  
END\_TYPE;

TYPE IfcEvaporativeCoolerTypeEnum = ENUMERATION OF  
    (DIRECTEVAPORATIVERANDOMMEDIAAIRCOOLER  
      , DIRECTEVAPORATIVERIGIDMEDIAAIRCOOLER  
      , DIRECTEVAPORATIVESLINGERSPACKAGEDAIRCOOLER  
      , DIRECTEVAPORATIVEPACKAGEDROTARYAIRCOOLER  
      , DIRECTEVAPORATIVEAIRWASHER  
      , INDIRECTEVAPORATIVEPACKAGEAIRCOOLER  
      , INDIRECTEVAPORATIVEWETCOIL  
      , INDIRECTEVAPORATIVECOOLINGTOWERORCOILCOOLER  
      , INDIRECTDIRECTCOMBINATION  
      , USERDEFINED  
      , NOTDEFINED) ;  
END\_TYPE;

TYPE IfcEvaporatorTypeEnum = ENUMERATION OF  
    (DIRECTEXPANSION  
      , DIRECTEXPANSIONSHELLANDTUBE  
      , DIRECTEXPANSIONTUBEINTUBE  
      , DIRECTEXPANSIONBRAZEDPLATE  
      , FLOODEDSHELLANDTUBE  
      , SHELLANDCOIL  
      , USERDEFINED  
      , NOTDEFINED) ;  
END\_TYPE;

TYPE IfcEventTriggerTypeEnum = ENUMERATION OF  
    (EVENTRULE  
      , EVENTMESSAGE  
      , EVENTTIME

---

```
, EVENTCOMPLEX
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcEventTypeEnum = ENUMERATION OF
(STARTEVENT
, ENDEVENT
, INTERMEDIATEEVENT
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcExternalSpatialElementTypeEnum = ENUMERATION OF
(EXTERNAL
, EXTERNAL_EARTH
, EXTERNAL_WATER
, EXTERNAL_FIRE
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcFanTypeEnum = ENUMERATION OF
(CENTRIFUGALFORWARDCURVED
, CENTRIFUGALRADIAL
, CENTRIFUGALBACKWARDINCLINEDCURVED
, CENTRIFUGALAIRFOIL
, TUBEAXIAL
, VANEAXIAL
, PROPELLORAXIAL
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcFastenerTypeEnum = ENUMERATION OF
(GLUE
, MORTAR
, WELD
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcFilterTypeEnum = ENUMERATION OF
(AIRPARTICLEFILTER
```



---

```
, COMPRESSED_AIR_FILTER
, ODOR_FILTER
, OIL_FILTER
, STRAINER
, WATER_FILTER
, USER_DEFINED
, NOT_DEFINED);
END_TYPE;
```

```
TYPE IfcFireSuppressionTerminalTypeEnum = ENUMERATION OF
(BREECHING_INLET
, FIRE_HYDRANT
, HOSE_REEL
, SPRINKLER
, SPRINKLER_DEFLECTOR
, USER_DEFINED
, NOT_DEFINED);
END_TYPE;
```

```
TYPE IfcFlowDirectionEnum = ENUMERATION OF
(SOURCE
, SINK
, SOURCE_AND_SINK
, NOT_DEFINED);
END_TYPE;
```

```
TYPE IfcFlowInstrumentTypeEnum = ENUMERATION OF
(PRESSURE_GAUGE
, THERMOMETER
, AMMETER
, FREQUENCY_METER
, POWER_FACTOR_METER
, PHASE_ANGLE_METER
, VOLT_METER_PEAK
, VOLT_METER_RMS
, USER_DEFINED
, NOT_DEFINED);
END_TYPE;
```

```
TYPE IfcFlowMeterTypeEnum = ENUMERATION OF
(ENERGY_METER
, GAS_METER
, OIL_METER
, WATER_METER
```

---

```
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcFootingTypeEnum = ENUMERATION OF
(CAISSON_FOUNDATION
, FOOTING_BEAM
, PAD_FOOTING
, PILE_CAP
, STRIP_FOOTING
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcFurnitureTypeEnum = ENUMERATION OF
(CHAIR
, TABLE
, DESK
, BED
, FILECABINET
, SHELF
, SOFA
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcGeographicElementTypeEnum = ENUMERATION OF
(TERRAIN
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcGeometricProjectionEnum = ENUMERATION OF
(GRAPH_VIEW
, SKETCH_VIEW
, MODEL_VIEW
, PLAN_VIEW
, REFLECTED_PLAN_VIEW
, SECTION_VIEW
, ELEVATION_VIEW
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

---

```
TYPE IfcGlobalOrLocalEnum = ENUMERATION OF
    (GLOBAL_COORDS
    , LOCAL_COORDS);
END_TYPE;
```

```
TYPE IfcGridTypeEnum = ENUMERATION OF
    (RECTANGULAR
    , RADIAL
    , TRIANGULAR
    , IRREGULAR
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcHeatExchangerTypeEnum = ENUMERATION OF
    (PLATE
    , SHELLANDTUBE
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcHumidifierTypeEnum = ENUMERATION OF
    (STEAMINJECTION
    , ADIABATICAIRWASHER
    , ADIABATICPAN
    , ADIABATICWETTEDELEMENT
    , ADIABATICATOMIZING
    , ADIABATICULTRASONIC
    , ADIABATICRIGIDMEDIA
    , ADIABATICCOMPRESSEDPAIRNOZZLE
    , ASSISTEDELECTRIC
    , ASSISTEDNATURALGAS
    , ASSISTEDPROPANE
    , ASSISTEDBUTANE
    , ASSISTEDSTEAM
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcInterceptorTypeEnum = ENUMERATION OF
    (CYCLONIC
    , GREASE
    , OIL
    , PETROL
```

---

```

        , USERDEFINED
        , NOTDEFINED) ;
END_TYPE;

TYPE IfcInternalOrExternalEnum = ENUMERATION OF
    (INTERNAL
    , EXTERNAL
    , EXTERNAL_EARTH
    , EXTERNAL_WATER
    , EXTERNAL_FIRE
    , NOTDEFINED) ;
END_TYPE;

TYPE IfcInventoryTypeEnum = ENUMERATION OF
    (ASSETINVENTORY
    , SPACEINVENTORY
    , FURNITUREINVENTORY
    , USERDEFINED
    , NOTDEFINED) ;
END_TYPE;

TYPE IfcJunctionBoxTypeEnum = ENUMERATION OF
    (DATA
    , POWER
    , USERDEFINED
    , NOTDEFINED) ;
END_TYPE;

TYPE IfcKnotType = ENUMERATION OF
    (UNIFORM_KNOTS
    , QUASI_UNIFORM_KNOTS
    , PIECEWISE_BEZIER_KNOTS
    , UNSPECIFIED) ;
END_TYPE;

TYPE IfcLaborResourceTypeEnum = ENUMERATION OF
    (ADMINISTRATION
    , CARPENTRY
    , CLEANING
    , CONCRETE
    , DRYWALL
    , ELECTRIC
    , FINISHING
    , FLOORING

```

---

```
, GENERAL
, HVAC
, LANDSCAPING
, MASONRY
, PAINTING
, PAVING
, PLUMBING
, ROOFING
, SITEGRADING
, STEELWORK
, SURVEYING
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcLampTypeEnum = ENUMERATION OF
  (COMPACTFLUORESCENT
  , FLUORESCENT
  , HALOGEN
  , HIGHPRESSUREMERCURY
  , HIGHPRESSURESODIUM
  , LED
  , METALHALIDE
  , OLED
  , TUNGSTENFILAMENT
  , USERDEFINED
  , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcLayerSetDirectionEnum = ENUMERATION OF
  (AXIS1
  , AXIS2
  , AXIS3);
END_TYPE;
```

```
TYPE IfcLightDistributionCurveEnum = ENUMERATION OF
  (TYPE_A
  , TYPE_B
  , TYPE_C
  , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcLightEmissionSourceEnum = ENUMERATION OF
  (COMPACTFLUORESCENT
```

---

```
, FLUORESCENT
, HIGHPRESSUREMERCURY
, HIGHPRESSURESODIUM
, LIGHTEMITTINGDIODE
, LOWPRESSURESODIUM
, LOWVOLTAGEHALOGEN
, MAINVOLTAGEHALOGEN
, METALHALIDE
, TUNGSTENFILAMENT
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcLightFixtureTypeEnum = ENUMERATION OF
(POINTSOURCE
, DIRECTIONSOURCE
, SECURITYLIGHTING
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcLoadGroupTypeEnum = ENUMERATION OF
(LOAD_GROUP
, LOAD_CASE
, LOAD_COMBINATION
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcLogicalOperatorEnum = ENUMERATION OF
(LOGICALAND
, LOGICALOR
, LOGICALXOR
, LOGICALNOTAND
, LOGICALNOTOR);
END_TYPE;
```

```
TYPE IfcMechanicalFastenerTypeEnum = ENUMERATION OF
(ANCHORBOLT
, BOLT
, DOWEL
, NAIL
, NAILPLATE
, RIVET
, SCREW
```

---

```
, SHEARCONNECTOR
, STAPLE
, STUDSHEARCONNECTOR
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcMedicalDeviceTypeEnum = ENUMERATION OF
    (AIRSTATION
    , FEEDAIRUNIT
    , OXYGENGENERATOR
    , OXYGENPLANT
    , VACUUMSTATION
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcMemberTypeEnum = ENUMERATION OF
    (BRACE
    , CHORD
    , COLLAR
    , MEMBER
    , MULLION
    , PLATE
    , POST
    , PURLIN
    , RAFTER
    , STRINGER
    , STRUT
    , STUD
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcMotorConnectionTypeEnum = ENUMERATION OF
    (BELTDRIVE
    , COUPLING
    , DIRECTDRIVE
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcNullStyle = ENUMERATION OF
    (NULL);
```

---

END\_TYPE;

TYPE IfcObjectTypeEnum = ENUMERATION OF  
    (PRODUCT  
    , PROCESS  
    , CONTROL  
    , RESOURCE  
    , ACTOR  
    , GROUP  
    , PROJECT  
    , NOTDEFINED);  
END\_TYPE;

TYPE IfcObjectiveEnum = ENUMERATION OF  
    (CODECOMPLIANCE  
    , CODEWAIVER  
    , DESIGNINTENT  
    , EXTERNAL  
    , HEALTHANDSAFETY  
    , MERGECONFLICT  
    , MODELVIEW  
    , PARAMETER  
    , REQUIREMENT  
    , SPECIFICATION  
    , TRIGGERCONDITION  
    , USERDEFINED  
    , NOTDEFINED);  
END\_TYPE;

TYPE IfcOccupantTypeEnum = ENUMERATION OF  
    (ASSIGNEE  
    , ASSIGNOR  
    , LESSEE  
    , LESSOR  
    , LETTINGAGENT  
    , OWNER  
    , TENANT  
    , USERDEFINED  
    , NOTDEFINED);  
END\_TYPE;

TYPE IfcOpeningElementTypeEnum = ENUMERATION OF  
    (OPENING  
    , RECESS



---

```
        , USERDEFINED
        , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcOutletTypeEnum = ENUMERATION OF
    (AUDIOVISUALOUTLET
    , COMMUNICATIONSOUTLET
    , POWEROUTLET
    , DATAOUTLET
    , TELEPHONEOUTLET
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcPerformanceHistoryTypeEnum = ENUMERATION OF
    (USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcPermeableCoveringOperationEnum = ENUMERATION OF
    (GRILL
    , LOUVER
    , SCREEN
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcPermitTypeEnum = ENUMERATION OF
    (ACCESS
    , BUILDING
    , WORK
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcPhysicalOrVirtualEnum = ENUMERATION OF
    (PHYSICAL
    , VIRTUAL
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcPileConstructionEnum = ENUMERATION OF
    (CAST_IN_PLACE
    , COMPOSITE
```

---

```
, PRECAST_CONCRETE
, PREFAB_STEEL
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcPileTypeEnum = ENUMERATION OF
(BORED
, DRIVEN
, JETGROUTING
, COHESION
, FRICTION
, SUPPORT
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcPipeFittingTypeEnum = ENUMERATION OF
(BEND
, CONNECTOR
, ENTRY
, EXIT
, JUNCTION
, OBSTRUCTION
, TRANSITION
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcPipeSegmentTypeEnum = ENUMERATION OF
(CULVERT
, FLEXIBLESEGMENT
, RIGIDSEGMENT
, GUTTER
, SPOOL
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcPlateTypeEnum = ENUMERATION OF
(CURTAIN_PANEL
, SHEET
, USERDEFINED
, NOTDEFINED);
```

---

END\_TYPE;

TYPE IfcPreferredSurfaceCurveRepresentation = ENUMERATION OF  
    (CURVE3D  
    , PCURVE\_S1  
    , PCURVE\_S2);  
END\_TYPE;

TYPE IfcProcedureTypeEnum = ENUMERATION OF  
    (ADVICE\_CAUTION  
    , ADVICE\_NOTE  
    , ADVICE\_WARNING  
    , CALIBRATION  
    , DIAGNOSTIC  
    , SHUTDOWN  
    , STARTUP  
    , USERDEFINED  
    , NOTDEFINED);  
END\_TYPE;

TYPE IfcProfileTypeEnum = ENUMERATION OF  
    (CURVE  
    , AREA);  
END\_TYPE;

TYPE IfcProjectOrderTypeEnum = ENUMERATION OF  
    (CHANGEORDER  
    , MAINTENANCEWORKORDER  
    , MOVEORDER  
    , PURCHASEORDER  
    , WORKORDER  
    , USERDEFINED  
    , NOTDEFINED);  
END\_TYPE;

TYPE IfcProjectedOrTrueLengthEnum = ENUMERATION OF  
    (PROJECTED\_LENGTH  
    , TRUE\_LENGTH);  
END\_TYPE;

TYPE IfcProjectionElementTypeEnum = ENUMERATION OF  
    (USERDEFINED  
    , NOTDEFINED);  
END\_TYPE;

---

```
TYPE IfcPropertySetTemplateTypeEnum = ENUMERATION OF
    (PSET_TYPEDRIVENONLY
    , PSET_TYPEDRIVENOVERRIDE
    , PSET_OCCURRENCEDRIVEN
    , PSET_PERFORMANCEDRIVEN
    , QTO_TYPEDRIVENONLY
    , QTO_TYPEDRIVENOVERRIDE
    , QTO_OCCURRENCEDRIVEN
    , NOTDEFINED) ;
END_TYPE;
```

```
TYPE IfcProtectiveDeviceTrippingUnitTypeEnum = ENUMERATION OF
    (ELECTRONIC
    , ELECTROMAGNETIC
    , RESIDUALCURRENT
    , THERMAL
    , USERDEFINED
    , NOTDEFINED) ;
END_TYPE;
```

```
TYPE IfcProtectiveDeviceTypeEnum = ENUMERATION OF
    (CIRCUITBREAKER
    , EARTHLEAKAGECIRCUITBREAKER
    , EARTHINGSWITCH
    , FUSEDISCONNECTOR
    , RESIDUALCURRENTCIRCUITBREAKER
    , RESIDUALCURRENTSWITCH
    , VARISTOR
    , USERDEFINED
    , NOTDEFINED) ;
END_TYPE;
```

```
TYPE IfcPumpTypeEnum = ENUMERATION OF
    (CIRCULATOR
    , ENDSUCTION
    , SPLITCASE
    , SUBMERSIBLEPUMP
    , SUMPPUMP
    , VERTICALINLINE
    , VERTICALTURBINE
    , USERDEFINED
    , NOTDEFINED) ;
END_TYPE;
```

---

```
TYPE IfcRailingTypeEnum = ENUMERATION OF
    (HANDRAIL
    , GUARDRAIL
    , BALUSTRADE
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcRampFlightTypeEnum = ENUMERATION OF
    (STRAIGHT
    , SPIRAL
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcRampTypeEnum = ENUMERATION OF
    (STRAIGHT_RUN_RAMP
    , TWO_STRAIGHT_RUN_RAMP
    , QUARTER_TURN_RAMP
    , TWO_QUARTER_TURN_RAMP
    , HALF_TURN_RAMP
    , SPIRAL_RAMP
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcRecurrenceTypeEnum = ENUMERATION OF
    (DAILY
    , WEEKLY
    , MONTHLY_BY_DAY_OF_MONTH
    , MONTHLY_BY_POSITION
    , BY_DAY_COUNT
    , BY_WEEKDAY_COUNT
    , YEARLY_BY_DAY_OF_MONTH
    , YEARLY_BY_POSITION);
END_TYPE;
```

```
TYPE IfcReflectanceMethodEnum = ENUMERATION OF
    (BLINN
    , FLAT
    , GLASS
    , MATT
    , METAL
```

---

```
, MIRROR
, PHONG
, PLASTIC
, STRAUSS
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcReinforcingBarRoleEnum = ENUMERATION OF
    (MAIN
    , SHEAR
    , LIGATURE
    , STUD
    , PUNCHING
    , EDGE
    , RING
    , ANCHORING
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcReinforcingBarSurfaceEnum = ENUMERATION OF
    (PLAIN
    , TEXTURED);
END_TYPE;
```

```
TYPE IfcReinforcingBarTypeEnum = ENUMERATION OF
    (ANCHORING
    , EDGE
    , LIGATURE
    , MAIN
    , PUNCHING
    , RING
    , SHEAR
    , STUD
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcReinforcingMeshTypeEnum = ENUMERATION OF
    (USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcRoleEnum = ENUMERATION OF
```

---

```

    (SUPPLIER
    , MANUFACTURER
    , CONTRACTOR
    , SUBCONTRACTOR
    , ARCHITECT
    , STRUCTURALENGINEER
    , COSTENGINEER
    , CLIENT
    , BUILDINGOWNER
    , BUILDINGOPERATOR
    , MECHANICALENGINEER
    , ELECTRICALENGINEER
    , PROJECTMANAGER
    , FACILITIESMANAGER
    , CIVILENGINEER
    , COMMISSIONINGENGINEER
    , ENGINEER
    , OWNER
    , CONSULTANT
    , CONSTRUCTIONMANAGER
    , FIELDCONSTRUCTIONMANAGER
    , RESELLER
    , USERDEFINED) ;
END_TYPE;

```

```

TYPE IfcRoofTypeEnum = ENUMERATION OF
    (FLAT_ROOF
    , SHED_ROOF
    , GABLE_ROOF
    , HIP_ROOF
    , HIPPED_GABLE_ROOF
    , GAMBREL_ROOF
    , MANSARD_ROOF
    , BARREL_ROOF
    , RAINBOW_ROOF
    , BUTTERFLY_ROOF
    , PAVILION_ROOF
    , DOME_ROOF
    , FREEFORM
    , USERDEFINED
    , NOTDEFINED) ;
END_TYPE;

```

```

TYPE IfcSIPrefix = ENUMERATION OF

```

---

```
(EXA
, PETA
, TERA
, GIGA
, MEGA
, KILO
, HECTO
, DECA
, DECI
, CENTI
, MILLI
, MICRO
, NANO
, PICO
, FEMTO
, ATTO);
END_TYPE;
```

```
TYPE IfcSIUnitName = ENUMERATION OF
```

```
(AMPERE
, BECQUEREL
, CANDELA
, COULOMB
, CUBIC_METRE
, DEGREE_CELSIUS
, FARAD
, GRAM
, GRAY
, HENRY
, HERTZ
, JOULE
, KELVIN
, LUMEN
, LUX
, METRE
, MOLE
, NEWTON
, OHM
, PASCAL
, RADIAN
, SECOND
, SIEMENS
, SIEVERT
, SQUARE_METRE
```



---

```
, STERADIAN
, TESLA
, VOLT
, WATT
, WEBER);
END_TYPE;
```

```
TYPE IfcSanitaryTerminalTypeEnum = ENUMERATION OF
    (BATH
    , BIDET
    , CISTERN
    , SHOWER
    , SINK
    , SANITARYFOUNTAIN
    , TOILETPAN
    , URINAL
    , WASHHANDBASIN
    , WCSEAT
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcSectionTypeEnum = ENUMERATION OF
    (UNIFORM
    , TAPERED);
END_TYPE;
```

```
TYPE IfcSensorTypeEnum = ENUMERATION OF
    (COSENSOR
    , CO2SENSOR
    , CONDUCTANCESENSOR
    , CONTACTSENSOR
    , FIRESSENSOR
    , FLOWSSENSOR
    , FROSTSENSOR
    , GASSENSOR
    , HEATSENSOR
    , HUMIDITYSENSOR
    , IDENTIFIERSSENSOR
    , IONCONCENTRATIONSENSOR
    , LEVELSENSOR
    , LIGHTSENSOR
    , MOISTURESENSOR
    , MOVEMENTSENSOR
```

---

```
, PHSENSOR
, PRESSURESENSOR
, RADIATIONSENSOR
, RADIOACTIVITYSENSOR
, SMOKESENSOR
, SOUNDSSENSOR
, TEMPERATURESENSOR
, WINDSENSOR
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcSequenceEnum = ENUMERATION OF
  (START_START
  , START_FINISH
  , FINISH_START
  , FINISH_FINISH
  , USERDEFINED
  , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcShadingDeviceTypeEnum = ENUMERATION OF
  (JALOUSIE
  , SHUTTER
  , AWNING
  , USERDEFINED
  , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcSimplePropertyTemplateTypeEnum = ENUMERATION OF
  (P_SINGLEVALUE
  , P_ENUMERATEDVALUE
  , P_BOUNDEDVALUE
  , P_LISTVALUE
  , P_TABLEVALUE
  , P_REFERENCEVALUE
  , Q_LENGTH
  , Q_AREA
  , Q_VOLUME
  , Q_COUNT
  , Q_WEIGHT
  , Q_TIME);
END_TYPE;
```

---

```
TYPE IfcSlabTypeEnum = ENUMERATION OF
    (FLOOR
    , ROOF
    , LANDING
    , BASESLAB
    , USERDEFINED
    , NOTDEFINED) ;
END_TYPE;
```

```
TYPE IfcSolarDeviceTypeEnum = ENUMERATION OF
    (SOLARCOLLECTOR
    , SOLARPANEL
    , USERDEFINED
    , NOTDEFINED) ;
END_TYPE;
```

```
TYPE IfcSpaceHeaterTypeEnum = ENUMERATION OF
    (CONVECTOR
    , RADIATOR
    , USERDEFINED
    , NOTDEFINED) ;
END_TYPE;
```

```
TYPE IfcSpaceTypeEnum = ENUMERATION OF
    (SPACE
    , PARKING
    , GFA
    , INTERNAL
    , EXTERNAL
    , USERDEFINED
    , NOTDEFINED) ;
END_TYPE;
```

```
TYPE IfcSpatialZoneTypeEnum = ENUMERATION OF
    (CONSTRUCTION
    , FIRESAFETY
    , LIGHTING
    , OCCUPANCY
    , SECURITY
    , THERMAL
    , TRANSPORT
    , VENTILATION
    , USERDEFINED
    , NOTDEFINED) ;
```

---

END\_TYPE;

TYPE IfcStackTerminalTypeEnum = ENUMERATION OF  
    (BIRDCAGE  
    , COWL  
    , RAINWATERHOPPER  
    , USERDEFINED  
    , NOTDEFINED);  
END\_TYPE;

TYPE IfcStairFlightTypeEnum = ENUMERATION OF  
    (STRAIGHT  
    , WINDER  
    , SPIRAL  
    , CURVED  
    , FREEFORM  
    , USERDEFINED  
    , NOTDEFINED);  
END\_TYPE;

TYPE IfcStairTypeEnum = ENUMERATION OF  
    (STRAIGHT\_RUN\_STAIR  
    , TWO\_STRAIGHT\_RUN\_STAIR  
    , QUARTER\_WINDING\_STAIR  
    , QUARTER\_TURN\_STAIR  
    , HALF\_WINDING\_STAIR  
    , HALF\_TURN\_STAIR  
    , TWO\_QUARTER\_WINDING\_STAIR  
    , TWO\_QUARTER\_TURN\_STAIR  
    , THREE\_QUARTER\_WINDING\_STAIR  
    , THREE\_QUARTER\_TURN\_STAIR  
    , SPIRAL\_STAIR  
    , DOUBLE\_RETURN\_STAIR  
    , CURVED\_RUN\_STAIR  
    , TWO\_CURVED\_RUN\_STAIR  
    , USERDEFINED  
    , NOTDEFINED);  
END\_TYPE;

TYPE IfcStateEnum = ENUMERATION OF  
    (READWRITE  
    , READONLY  
    , LOCKED  
    , READWRITELOCKED

---

```
    , READONLYLOCKED) ;  
END_TYPE;
```

```
TYPE IfcStructuralCurveActivityTypeEnum = ENUMERATION OF  
    (CONST  
    , LINEAR  
    , POLYGONAL  
    , EQUIDISTANT  
    , SINUS  
    , PARABOLA  
    , DISCRETE  
    , USERDEFINED  
    , NOTDEFINED) ;  
END_TYPE;
```

```
TYPE IfcStructuralCurveMemberTypeEnum = ENUMERATION OF  
    (RIGID_JOINED_MEMBER  
    , PIN_JOINED_MEMBER  
    , CABLE  
    , TENSION_MEMBER  
    , COMPRESSION_MEMBER  
    , USERDEFINED  
    , NOTDEFINED) ;  
END_TYPE;
```

```
TYPE IfcStructuralSurfaceActivityTypeEnum = ENUMERATION OF  
    (CONST  
    , BILINEAR  
    , DISCRETE  
    , ISOCONTOUR  
    , USERDEFINED  
    , NOTDEFINED) ;  
END_TYPE;
```

```
TYPE IfcStructuralSurfaceMemberTypeEnum = ENUMERATION OF  
    (BENDING_ELEMENT  
    , MEMBRANE_ELEMENT  
    , SHELL  
    , USERDEFINED  
    , NOTDEFINED) ;  
END_TYPE;
```

```
TYPE IfcSubContractResourceTypeEnum = ENUMERATION OF  
    (PURCHASE
```

---

```
    , WORK
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcSurfaceFeatureTypeEnum = ENUMERATION OF
    (MARK
    , TAG
    , TREATMENT
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcSurfaceSide = ENUMERATION OF
    (POSITIVE
    , NEGATIVE
    , BOTH);
END_TYPE;
```

```
TYPE IfcSwitchingDeviceTypeEnum = ENUMERATION OF
    (CONTACTOR
    , DIMMERSWITCH
    , EMERGENCYSTOP
    , KEYPAD
    , MOMENTARYSWITCH
    , SELECTORSWITCH
    , STARTER
    , SWITCHDISCONNECTOR
    , TOGGLESWITCH
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcSystemFurnitureElementTypeEnum = ENUMERATION OF
    (PANEL
    , WORKSURFACE
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcTankTypeEnum = ENUMERATION OF
    (BASIN
    , BREAKPRESSURE
    , EXPANSION
```

---

```
    , FEEDANDEXPANSION
    , PRESSUREVESSEL
    , STORAGE
    , VESSEL
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcTaskDurationEnum = ENUMERATION OF
    (ELAPSEDTIME
    , WORKTIME
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcTaskTypeEnum = ENUMERATION OF
    (ATTENDANCE
    , CONSTRUCTION
    , DEMOLITION
    , DISMANTLE
    , DISPOSAL
    , INSTALLATION
    , LOGISTIC
    , MAINTENANCE
    , MOVE
    , OPERATION
    , REMOVAL
    , RENOVATION
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcTendonAnchorTypeEnum = ENUMERATION OF
    (COUPLER
    , FIXED_END
    , TENSIONING_END
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcTendonTypeEnum = ENUMERATION OF
    (BAR
    , COATED
    , STRAND
    , WIRE
```

---

```
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcTextPath = ENUMERATION OF
    (LEFT
    , RIGHT
    , UP
    , DOWN);
END_TYPE;
```

```
TYPE IfcTimeSeriesDataTypeEnum = ENUMERATION OF
    (CONTINUOUS
    , DISCRETE
    , DISCRETEBINARY
    , PIECEWISEBINARY
    , PIECEWISECONSTANT
    , PIECEWISECONTINUOUS
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcTransformerTypeEnum = ENUMERATION OF
    (CURRENT
    , FREQUENCY
    , INVERTER
    , RECTIFIER
    , VOLTAGE
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcTransitionCode = ENUMERATION OF
    (DISCONTINUOUS
    , CONTINUOUS
    , CONTSAMEGRADIENT
    , CONTSAMEGRADIENTSAMECURVATURE);
END_TYPE;
```

```
TYPE IfcTransportElementTypeEnum = ENUMERATION OF
    (ELEVATOR
    , ESCALATOR
    , MOVINGWALKWAY
    , CRANEWAY
    , LIFTINGGEAR
```



---

```

        , USERDEFINED
        , NOTDEFINED) ;
END_TYPE;

TYPE IfcTrimmingPreference = ENUMERATION OF
    (CARTESIAN
    , PARAMETER
    , UNSPECIFIED) ;
END_TYPE;

TYPE IfcTubeBundleTypeEnum = ENUMERATION OF
    (FINNED
    , USERDEFINED
    , NOTDEFINED) ;
END_TYPE;

TYPE IfcUnitEnum = ENUMERATION OF
    (ABSORBEDDOSEUNIT
    , AMOUNTOFSUBSTANCEUNIT
    , AREAUNIT
    , DOSEEQUIVALENTUNIT
    , ELECTRICCAPACITANCEUNIT
    , ELECTRICCHARGEUNIT
    , ELECTRICCONDUCTANCEUNIT
    , ELECTRICCURRENTUNIT
    , ELECTRICRESISTANCEUNIT
    , ELECTRICVOLTAGEUNIT
    , ENERGYUNIT
    , FORCEUNIT
    , FREQUENCYUNIT
    , ILLUMINANCEUNIT
    , INDUCTANCEUNIT
    , LENGTHUNIT
    , LUMINOUSFLUXUNIT
    , LUMINOUSINTENSITYUNIT
    , MAGNETICFLUXDENSITYUNIT
    , MAGNETICFLUXUNIT
    , MASSUNIT
    , PLANEANGLEUNIT
    , POWERUNIT
    , PRESSUREUNIT
    , RADIOACTIVITYUNIT
    , SOLIDANGLEUNIT
    , THERMODYNAMICTEMPERATUREUNIT

```

---

```
, TIMEUNIT  
, VOLUMEUNIT  
, USERDEFINED);  
END_TYPE;
```

```
TYPE IfcUnitaryControlElementTypeEnum = ENUMERATION OF  
  (ALARMPANEL  
  , CONTROLPANEL  
  , GASDETECTIONPANEL  
  , INDICATORPANEL  
  , MIMICPANEL  
  , HUMIDISTAT  
  , THERMOSTAT  
  , WEATHERSTATION  
  , USERDEFINED  
  , NOTDEFINED);  
END_TYPE;
```

```
TYPE IfcUnitaryEquipmentTypeEnum = ENUMERATION OF  
  (AIRHANDLER  
  , AIRCONDITIONINGUNIT  
  , DEHUMIDIFIER  
  , SPLITSYSTEM  
  , ROOFTOPUNIT  
  , USERDEFINED  
  , NOTDEFINED);  
END_TYPE;
```

```
TYPE IfcValveTypeEnum = ENUMERATION OF  
  (AIRRELEASE  
  , ANTIVACUUM  
  , CHANGEOVER  
  , CHECK  
  , COMMISSIONING  
  , DIVERTING  
  , DRAWOFFCOCK  
  , DOUBLECHECK  
  , DOUBLEREGULATING  
  , FAUCET  
  , FLUSHING  
  , GASCOCK  
  , GASTAP  
  , ISOLATING  
  , MIXING
```

---

```
, PRESSUREREDUCING
, PRESSURERELIEF
, REGULATING
, SAFETYCUTOFF
, STEAMTRAP
, STOPCOCK
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcVibrationIsolatorTypeEnum = ENUMERATION OF
    (COMPRESSION
    , SPRING
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcVoidingFeatureTypeEnum = ENUMERATION OF
    (CUTOUT
    , NOTCH
    , HOLE
    , MITER
    , CHAMFER
    , EDGE
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcWallTypeEnum = ENUMERATION OF
    (MOVABLE
    , PARAPET
    , PARTITIONING
    , PLUMBINGWALL
    , SHEAR
    , SOLIDWALL
    , STANDARD
    , POLYGONAL
    , ELEMENTEDWALL
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;
```

```
TYPE IfcWasteTerminalTypeEnum = ENUMERATION OF
    (FLOORTRAP
```

---

```
, FLOORWASTE
, GULLYSUMP
, GULLYTRAP
, ROOFDRAIN
, WASTEDISPOSALUNIT
, WASTETRAPH
, USERDEFINED
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcWindowPanelOperationEnum = ENUMERATION OF
(SIDEHUNGRIGHTHAND
, SIDEHUNGLEFTHAND
, TILTANDTURNRIGHTHAND
, TILTANDTURNLEFTHAND
, TOPHUNG
, BOTTOMHUNG
, PIVOTHORIZONTAL
, PIVOTVERTICAL
, SLIDINGHORIZONTAL
, SLIDINGVERTICAL
, REMOVABLECASEMENT
, FIXEDCASEMENT
, OTHEROPERATION
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcWindowPanelPositionEnum = ENUMERATION OF
(LEFT
, MIDDLE
, RIGHT
, BOTTOM
, TOP
, NOTDEFINED);
END_TYPE;
```

```
TYPE IfcWindowStyleConstructionEnum = ENUMERATION OF
(ALUMINIUM
, HIGH_GRADE_STEEL
, STEEL
, WOOD
, ALUMINIUM_WOOD
, PLASTIC
, OTHER_CONSTRUCTION
```

---

```
    , NOTDEFINED) ;  
END_TYPE;
```

```
TYPE IfcWindowStyleOperationEnum = ENUMERATION OF  
    (SINGLE_PANEL  
    , DOUBLE_PANEL_VERTICAL  
    , DOUBLE_PANEL_HORIZONTAL  
    , TRIPLE_PANEL_VERTICAL  
    , TRIPLE_PANEL_BOTTOM  
    , TRIPLE_PANEL_TOP  
    , TRIPLE_PANEL_LEFT  
    , TRIPLE_PANEL_RIGHT  
    , TRIPLE_PANEL_HORIZONTAL  
    , USERDEFINED  
    , NOTDEFINED) ;  
END_TYPE;
```

```
TYPE IfcWindowTypeEnum = ENUMERATION OF  
    (WINDOW  
    , SKYLIGHT  
    , LIGHTDOME  
    , USERDEFINED  
    , NOTDEFINED) ;  
END_TYPE;
```

```
TYPE IfcWindowTypePartitioningEnum = ENUMERATION OF  
    (SINGLE_PANEL  
    , DOUBLE_PANEL_VERTICAL  
    , DOUBLE_PANEL_HORIZONTAL  
    , TRIPLE_PANEL_VERTICAL  
    , TRIPLE_PANEL_BOTTOM  
    , TRIPLE_PANEL_TOP  
    , TRIPLE_PANEL_LEFT  
    , TRIPLE_PANEL_RIGHT  
    , TRIPLE_PANEL_HORIZONTAL  
    , USERDEFINED  
    , NOTDEFINED) ;  
END_TYPE;
```

```
TYPE IfcWorkCalendarTypeEnum = ENUMERATION OF  
    (FIRSTSHIFT  
    , SECONDSHIFT  
    , THIRDSHIFT  
    , USERDEFINED
```

---

```

        , NOTDEFINED);
END_TYPE;

TYPE IfcWorkPlanTypeEnum = ENUMERATION OF
    (ACTUAL
    , BASELINE
    , PLANNED
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;

TYPE IfcWorkScheduleTypeEnum = ENUMERATION OF
    (ACTUAL
    , BASELINE
    , PLANNED
    , USERDEFINED
    , NOTDEFINED);
END_TYPE;

TYPE IfcActorSelect = SELECT
    (IfcOrganization
    , IfcPerson
    , IfcPersonAndOrganization);
END_TYPE;

TYPE IfcAppliedValueSelect = SELECT
    (IfcMeasureWithUnit
    , IfcReference
    , IfcValue);
END_TYPE;

TYPE IfcAxis2Placement = SELECT
    (IfcAxis2Placement2D
    , IfcAxis2Placement3D);
END_TYPE;

TYPE IfcBendingParameterSelect = SELECT
    (IfcLengthMeasure
    , IfcPlaneAngleMeasure);
END_TYPE;

TYPE IfcBooleanOperand = SELECT
    (IfcBooleanResult
    , IfcCsgPrimitive3D

```

---

```

        , IfcHalfSpaceSolid
        , IfcSolidModel
        , IfcTessellatedFaceSet);
END_TYPE;

TYPE IfcClassificationReferenceSelect = SELECT
    (IfcClassification
    , IfcClassificationReference);
END_TYPE;

TYPE IfcClassificationSelect = SELECT
    (IfcClassification
    , IfcClassificationReference);
END_TYPE;

TYPE IfcColour = SELECT
    (IfcColourSpecification
    , IfcPreDefinedColour);
END_TYPE;

TYPE IfcColourOrFactor = SELECT
    (IfcColourRgb
    , IfcNormalisedRatioMeasure);
END_TYPE;

TYPE IfcCoordinateReferenceSystemSelect = SELECT
    (IfcCoordinateReferenceSystem
    , IfcGeometricRepresentationContext);
END_TYPE;

TYPE IfcCsgSelect = SELECT
    (IfcBooleanResult
    , IfcCsgPrimitive3D);
END_TYPE;

TYPE IfcCurveFontOrScaledCurveFontSelect = SELECT
    (IfcCurveStyleFontAndScaling
    , IfcCurveStyleFontSelect);
END_TYPE;

TYPE IfcCurveOnSurface = SELECT
    (IfcCompositeCurveOnSurface
    , IfcPcurve
    , IfcSurfaceCurve);

```

---

END\_TYPE;

TYPE IfcCurveOrEdgeCurve = SELECT  
    (IfcBoundedCurve  
      , IfcEdgeCurve);  
END\_TYPE;

TYPE IfcCurveStyleFontSelect = SELECT  
    (IfcCurveStyleFont  
      , IfcPreDefinedCurveFont);  
END\_TYPE;

TYPE IfcDefinitionSelect = SELECT  
    (IfcObjectDefinition  
      , IfcPropertyDefinition);  
END\_TYPE;

TYPE IfcDerivedMeasureValue = SELECT  
    (IfcAbsorbedDoseMeasure  
      , IfcAccelerationMeasure  
      , IfcAngularVelocityMeasure  
      , IfcAreaDensityMeasure  
      , IfcCompoundPlaneAngleMeasure  
      , IfcCurvatureMeasure  
      , IfcDoseEquivalentMeasure  
      , IfcDynamicViscosityMeasure  
      , IfcElectricCapacitanceMeasure  
      , IfcElectricChargeMeasure  
      , IfcElectricConductanceMeasure  
      , IfcElectricResistanceMeasure  
      , IfcElectricVoltageMeasure  
      , IfcEnergyMeasure  
      , IfcForceMeasure  
      , IfcFrequencyMeasure  
      , IfcHeatFluxDensityMeasure  
      , IfcHeatingValueMeasure  
      , IfcIlluminanceMeasure  
      , IfcInductanceMeasure  
      , IfcIntegerCountRateMeasure  
      , IfcIonConcentrationMeasure  
      , IfcIsothermalMoistureCapacityMeasure  
      , IfcKinematicViscosityMeasure  
      , IfcLinearForceMeasure  
      , IfcLinearMomentMeasure



---

, IfcLinearStiffnessMeasure  
, IfcLinearVelocityMeasure  
, IfcLuminousFluxMeasure  
, IfcLuminousIntensityDistributionMeasure  
, IfcMagneticFluxDensityMeasure  
, IfcMagneticFluxMeasure  
, IfcMassDensityMeasure  
, IfcMassFlowRateMeasure  
, IfcMassPerLengthMeasure  
, IfcModulusOfElasticityMeasure  
, IfcModulusOfLinearSubgradeReactionMeasure  
, IfcModulusOfRotationalSubgradeReactionMeasure  
, IfcModulusOfSubgradeReactionMeasure  
, IfcMoistureDiffusivityMeasure  
, IfcMolecularWeightMeasure  
, IfcMomentOfInertiaMeasure  
, IfcMonetaryMeasure  
, IfcPHMeasure  
, IfcPlanarForceMeasure  
, IfcPowerMeasure  
, IfcPressureMeasure  
, IfcRadioActivityMeasure  
, IfcRotationalFrequencyMeasure  
, IfcRotationalMassMeasure  
, IfcRotationalStiffnessMeasure  
, IfcSectionModulusMeasure  
, IfcSectionalAreaIntegralMeasure  
, IfcShearModulusMeasure  
, IfcSoundPowerLevelMeasure  
, IfcSoundPowerMeasure  
, IfcSoundPressureLevelMeasure  
, IfcSoundPressureMeasure  
, IfcSpecificHeatCapacityMeasure  
, IfcTemperatureGradientMeasure  
, IfcTemperatureRateOfChangeMeasure  
, IfcThermalAdmittanceMeasure  
, IfcThermalConductivityMeasure  
, IfcThermalExpansionCoefficientMeasure  
, IfcThermalResistanceMeasure  
, IfcThermalTransmittanceMeasure  
, IfcTorqueMeasure  
, IfcVaporPermeabilityMeasure  
, IfcVolumetricFlowRateMeasure  
, IfcWarpingConstantMeasure

---

```
    , IfcWarpingMomentMeasure);  
END_TYPE;
```

```
TYPE IfcDocumentSelect = SELECT  
    (IfcDocumentInformation  
    , IfcDocumentReference);  
END_TYPE;
```

```
TYPE IfcFillStyleSelect = SELECT  
    (IfcColour  
    , IfcExternallyDefinedHatchStyle  
    , IfcFillAreaStyleHatching  
    , IfcFillAreaStyleTiles);  
END_TYPE;
```

```
TYPE IfcGeometricSetSelect = SELECT  
    (IfcCurve  
    , IfcPoint  
    , IfcSurface);  
END_TYPE;
```

```
TYPE IfcGridPlacementDirectionSelect = SELECT  
    (IfcDirection  
    , IfcVirtualGridIntersection);  
END_TYPE;
```

```
TYPE IfcHatchLineDistanceSelect = SELECT  
    (IfcPositiveLengthMeasure  
    , IfcVector);  
END_TYPE;
```

```
TYPE IfcLayeredItem = SELECT  
    (IfcRepresentation  
    , IfcRepresentationItem);  
END_TYPE;
```

```
TYPE IfcLibrarySelect = SELECT  
    (IfcLibraryInformation  
    , IfcLibraryReference);  
END_TYPE;
```

```
TYPE IfcLightDistributionDataSourceSelect = SELECT  
    (IfcExternalReference  
    , IfcLightIntensityDistribution);
```

---

END\_TYPE;

```
TYPE IfcMaterialSelect = SELECT
    (IfcMaterialDefinition
    , IfcMaterialList
    , IfcMaterialUsageDefinition);
END_TYPE;
```

```
TYPE IfcMeasureValue = SELECT
    (IfcAmountOfSubstanceMeasure
    , IfcAreaMeasure
    , IfcComplexNumber
    , IfcContextDependentMeasure
    , IfcCountMeasure
    , IfcDescriptiveMeasure
    , IfcElectricCurrentMeasure
    , IfcLengthMeasure
    , IfcLuminousIntensityMeasure
    , IfcMassMeasure
    , IfcNonNegativeLengthMeasure
    , IfcNormalisedRatioMeasure
    , IfcNumericMeasure
    , IfcParameterValue
    , IfcPlaneAngleMeasure
    , IfcPositiveLengthMeasure
    , IfcPositivePlaneAngleMeasure
    , IfcPositiveRatioMeasure
    , IfcRatioMeasure
    , IfcSolidAngleMeasure
    , IfcThermodynamicTemperatureMeasure
    , IfcTimeMeasure
    , IfcVolumeMeasure);
END_TYPE;
```

```
TYPE IfcMetricValueSelect = SELECT
    (IfcAppliedValue
    , IfcMeasureWithUnit
    , IfcReference
    , IfcTable
    , IfcTimeSeries
    , IfcValue);
END_TYPE;
```

```
TYPE IfcModulusOfRotationalSubgradeReactionSelect = SELECT
```

---

```

        (IfcBoolean
        , IfcModulusOfRotationalSubgradeReactionMeasure);
END_TYPE;

TYPE IfcModulusOfSubgradeReactionSelect = SELECT
    (IfcBoolean
    , IfcModulusOfSubgradeReactionMeasure);
END_TYPE;

TYPE IfcModulusOfTranslationalSubgradeReactionSelect = SELECT
    (IfcBoolean
    , IfcModulusOfLinearSubgradeReactionMeasure);
END_TYPE;

TYPE IfcObjectReferenceSelect = SELECT
    (IfcAddress
    , IfcAppliedValue
    , IfcExternalReference
    , IfcMaterialDefinition
    , IfcOrganization
    , IfcPerson
    , IfcPersonAndOrganization
    , IfcTable
    , IfcTimeSeries);
END_TYPE;

TYPE IfcPointOrVertexPoint = SELECT
    (IfcPoint
    , IfcVertexPoint);
END_TYPE;

TYPE IfcPresentationStyleSelect = SELECT
    (IfcCurveStyle
    , IfcFillAreaStyle
    , IfcNullStyle
    , IfcSurfaceStyle
    , IfcTextStyle);
END_TYPE;

TYPE IfcProcessSelect = SELECT
    (IfcProcess
    , IfcTypeProcess);
END_TYPE;

```

---

```
TYPE IfcProductRepresentationSelect = SELECT
    (IfcProductDefinitionShape
    , IfcRepresentationMap);
END_TYPE;
```

```
TYPE IfcProductSelect = SELECT
    (IfcProduct
    , IfcTypeProduct);
END_TYPE;
```

```
TYPE IfcPropertySetDefinitionSelect = SELECT
    (IfcPropertySetDefinition
    , IfcPropertySetDefinitionSet);
END_TYPE;
```

```
TYPE IfcResourceObjectSelect = SELECT
    (IfcActorRole
    , IfcAppliedValue
    , IfcApproval
    , IfcConstraint
    , IfcContextDependentUnit
    , IfcConversionBasedUnit
    , IfcExternalInformation
    , IfcExternalReference
    , IfcMaterialDefinition
    , IfcOrganization
    , IfcPerson
    , IfcPersonAndOrganization
    , IfcPhysicalQuantity
    , IfcProfileDef
    , IfcPropertyAbstraction
    , IfcTimeSeries);
END_TYPE;
```

```
TYPE IfcResourceSelect = SELECT
    (IfcResource
    , IfcTypeResource);
END_TYPE;
```

```
TYPE IfcRotationalStiffnessSelect = SELECT
    (IfcBoolean
    , IfcRotationalStiffnessMeasure);
END_TYPE;
```

---

```
TYPE IfcSegmentIndexSelect = SELECT
    (IfcArcIndex
    , IfcLineIndex);
END_TYPE;
```

```
TYPE IfcShell = SELECT
    (IfcClosedShell
    , IfcOpenShell);
END_TYPE;
```

```
TYPE IfcSimpleValue = SELECT
    (IfcBinary
    , IfcBoolean
    , IfcDate
    , IfcDateTime
    , IfcDuration
    , IfcIdentifier
    , IfcInteger
    , IfcLabel
    , IfcLogical
    , IfcPositiveInteger
    , IfcReal
    , IfcText
    , IfcTime
    , IfcTimeStamp);
END_TYPE;
```

```
TYPE IfcSizeSelect = SELECT
    (IfcDescriptiveMeasure
    , IfcLengthMeasure
    , IfcNormalisedRatioMeasure
    , IfcPositiveLengthMeasure
    , IfcPositiveRatioMeasure
    , IfcRatioMeasure);
END_TYPE;
```

```
TYPE IfcSolidOrShell = SELECT
    (IfcClosedShell
    , IfcSolidModel);
END_TYPE;
```

```
TYPE IfcSpaceBoundarySelect = SELECT
    (IfcExternalSpatialElement
    , IfcSpace);
```

---

END\_TYPE;

TYPE IfcSpecularHighlightSelect = SELECT  
    (IfcSpecularExponent  
      , IfcSpecularRoughness);  
END\_TYPE;

TYPE IfcStructuralActivityAssignmentSelect = SELECT  
    (IfcElement  
      , IfcStructuralItem);  
END\_TYPE;

TYPE IfcStyleAssignmentSelect = SELECT  
    (IfcPresentationStyle  
      , IfcPresentationStyleAssignment);  
END\_TYPE;

TYPE IfcSurfaceOrFaceSurface = SELECT  
    (IfcFaceBasedSurfaceModel  
      , IfcFaceSurface  
      , IfcSurface);  
END\_TYPE;

TYPE IfcSurfaceStyleElementSelect = SELECT  
    (IfcExternallyDefinedSurfaceStyle  
      , IfcSurfaceStyleLighting  
      , IfcSurfaceStyleRefraction  
      , IfcSurfaceStyleShading  
      , IfcSurfaceStyleWithTextures);  
END\_TYPE;

TYPE IfcTextFontSelect = SELECT  
    (IfcExternallyDefinedTextFont  
      , IfcPreDefinedTextFont);  
END\_TYPE;

TYPE IfcTimeOrRatioSelect = SELECT  
    (IfcDuration  
      , IfcRatioMeasure);  
END\_TYPE;

TYPE IfcTranslationalStiffnessSelect = SELECT  
    (IfcBoolean  
      , IfcLinearStiffnessMeasure);

---

```

END_TYPE;

TYPE IfcTrimmingSelect = SELECT
    (IfcCartesianPoint
     , IfcParameterValue);
END_TYPE;

TYPE IfcUnit = SELECT
    (IfcDerivedUnit
     , IfcMonetaryUnit
     , IfcNamedUnit);
END_TYPE;

TYPE IfcValue = SELECT
    (IfcDerivedMeasureValue
     , IfcMeasureValue
     , IfcSimpleValue);
END_TYPE;

TYPE IfcVectorOrDirection = SELECT
    (IfcDirection
     , IfcVector);
END_TYPE;

TYPE IfcWarpingStiffnessSelect = SELECT
    (IfcBoolean
     , IfcWarpingMomentMeasure);
END_TYPE;

ENTITY IfcActionRequest
    SUBTYPE OF (IfcControl);
    PredefinedType : OPTIONAL IfcActionRequestTypeEnum;
    Status : OPTIONAL IfcLabel;
    LongDescription : OPTIONAL IfcText;
END_ENTITY;

ENTITY IfcActor
    SUPERTYPE OF (ONEOF
        (IfcOccupant))
    SUBTYPE OF (IfcObject);
    TheActor : IfcActorSelect;
    INVERSE
        IsActingUpon : SET [0:?] OF IfcRelAssignsToActor FOR RelatingActor;
END_ENTITY;

```



---

```

ENTITY IfcActorRole;
    Role : IfcRoleEnum;
    UserDefinedRole : OPTIONAL IfcLabel;
    Description : OPTIONAL IfcText;
INVERSE
    HasExternalReference : SET [0:?] OF IfcExternalReferenceRelationship FOR
RelatedResourceObjects;
WHERE
    WR1 : (Role <> IfcRoleEnum.USERDEFINED) OR
((Role = IfcRoleEnum.USERDEFINED) AND
    EXISTS(SELF.UserDefinedRole));
END_ENTITY;

```

```

ENTITY IfcActuator
    SUBTYPE OF (IfcDistributionControlElement);
    PredefinedType : OPTIONAL IfcActuatorTypeEnum;
WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
(PredefinedType <> IfcActuatorTypeEnum.USERDEFINED) OR
((PredefinedType = IfcActuatorTypeEnum.USERDEFINED) AND EXISTS
(SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
('IFC4.IFCACTUATORTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcActuatorType
    SUBTYPE OF (IfcDistributionControlElementType);
    PredefinedType : IfcActuatorTypeEnum;
WHERE
    CorrectPredefinedType : (PredefinedType <> IfcActuatorTypeEnum.USERDEFINED)
OR
((PredefinedType = IfcActuatorTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcAddress
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcPostalAddress
        , IfcTelecomAddress));
    Purpose : OPTIONAL IfcAddressTypeEnum;
    Description : OPTIONAL IfcText;
    UserDefinedPurpose : OPTIONAL IfcLabel;
INVERSE

```

---

```

    OfPerson : SET [0:?] OF IfcPerson FOR Addresses;
    OfOrganization : SET [0:?] OF IfcOrganization FOR Addresses;
WHERE
    WR1 : (NOT(EXISTS(Purpose))) OR
((Purpose <> IfcAddressTypeEnum.USERDEFINED) OR
((Purpose = IfcAddressTypeEnum.USERDEFINED) AND
    EXISTS(SELF.UserDefinedPurpose)));
END_ENTITY;

ENTITY IfcAdvancedBrep
    SUPERTYPE OF (ONEOF
        (IfcAdvancedBrepWithVoids))
    SUBTYPE OF (IfcManifoldSolidBrep);
WHERE
    HasAdvancedFaces : SIZEOF(QUERY (Afs < *
SELF\IfcManifoldSolidBrep.Outer.CfsFaces |
    (NOT (' IFC4. IFCADVANCEDFACE' IN TYPEOF(Afs)))
)) = 0;
END_ENTITY;

ENTITY IfcAdvancedBrepWithVoids
    SUBTYPE OF (IfcAdvancedBrep);
    Voids : SET [1:?] OF IfcClosedShell;
WHERE
    VoidsHaveAdvancedFaces : SIZEOF (QUERY (Vsh < * Voids |
    SIZEOF (QUERY (Afs < * Vsh.CfsFaces |
    (NOT (' IFC4. IFCADVANCEDFACE' IN TYPEOF(Afs)))
    )) = 0
    )) = 0;
END_ENTITY;

ENTITY IfcAdvancedFace
    SUBTYPE OF (IfcFaceSurface);
WHERE
    ApplicableSurface : SIZEOF (
[' IFC4. IFCELEMENTARYSURFACE',
    ' IFC4. IFCSWEPTSURFACE',
    ' IFC4. IFCBSPLINESURFACE' ] *
    TYPEOF(SELF\IfcFaceSurface.FaceSurface)) = 1;
    RequiresEdgeCurve : SIZEOF(QUERY (ElpFbnds < *
    QUERY (Bnds < * SELF\IfcFace.Bounds |
    ' IFC4. IFCEDGELOOP' IN TYPEOF(Bnds.Bound)) |
    NOT (SIZEOF (QUERY (Oe < * ElpFbnds.Bound\IfcEdgeLoop.EdgeList |
    NOT(' IFC4. IFCEDGECURVE' IN

```

---

```

        TYPEOF (Oe\IfcOrientedEdge. EdgeElement)
    ))) = 0
))) = 0;
    ApplicableEdgeCurves : SIZEOF (QUERY (ElpFbnds <*
QUERY (Bnds <* SELF\IfcFace. Bounds |
    ' IFC4. IFCEDGELOOP' IN TYPEOF (Bnds. Bound)) |
    NOT (SIZEOF (QUERY (Oe <* ElpFbnds. Bound\IfcEdgeLoop. EdgeList |
        NOT (SIZEOF ([ ' IFC4. IFCLINE',
                        ' IFC4. IFCCONIC',
                        ' IFC4. IFCPOLYLINE',
                        ' IFC4. IFCBSPLINECURVE' ] *
        TYPEOF (Oe\IfcOrientedEdge. EdgeElement\IfcEdgeCurve. EdgeGeometry))) = 1 )
    )) = 0
))) = 0;
END_ENTITY;

```

```

ENTITY IfcAirTerminal
    SUBTYPE OF (IfcFlowTerminal);
    PredefinedType : OPTIONAL IfcAirTerminalTypeEnum;
    WHERE
        CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
        (PredefinedType <> IfcAirTerminalTypeEnum. USERDEFINED) OR
        ((PredefinedType = IfcAirTerminalTypeEnum. USERDEFINED) AND EXISTS
        (SELF\IfcObject. ObjectType));
        CorrectTypeAssigned : (SIZEOF (IsTypedBy) = 0) OR
        (' IFC4. IFCAIRTERMINALTYPE' IN
        TYPEOF (SELF\IfcObject. IsTypedBy[1]. RelatingType));
END_ENTITY;

```

```

ENTITY IfcAirTerminalBox
    SUBTYPE OF (IfcFlowController);
    PredefinedType : OPTIONAL IfcAirTerminalBoxTypeEnum;
    WHERE
        CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
        (PredefinedType <> IfcAirTerminalBoxTypeEnum. USERDEFINED) OR
        ((PredefinedType = IfcAirTerminalBoxTypeEnum. USERDEFINED) AND EXISTS
        (SELF\IfcObject. ObjectType));
        CorrectTypeAssigned : (SIZEOF (IsTypedBy) = 0) OR
        (' IFC4. IFCAIRTERMINALBOXTYPE' IN
        TYPEOF (SELF\IfcObject. IsTypedBy[1]. RelatingType));
END_ENTITY;

```

```

ENTITY IfcAirTerminalBoxType
    SUBTYPE OF (IfcFlowControllerType);

```

---

```

    PredefinedType : IfcAirTerminalBoxTypeEnum;
WHERE
    CorrectPredefinedType : (PredefinedType <>
IfcAirTerminalBoxTypeEnum.USERDEFINED) OR
((PredefinedType = IfcAirTerminalBoxTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcAirTerminalType
SUBTYPE OF (IfcFlowTerminalType);
    PredefinedType : IfcAirTerminalTypeEnum;
WHERE
    CorrectPredefinedType : (PredefinedType <> IfcAirTerminalTypeEnum.USERDEFINED)
OR
((PredefinedType = IfcAirTerminalTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcAirToAirHeatRecovery
SUBTYPE OF (IfcEnergyConversionDevice);
    PredefinedType : OPTIONAL IfcAirToAirHeatRecoveryTypeEnum;
WHERE
    CorrectPredefinedType : NOT (EXISTS(PredefinedType)) OR
(PredefinedType <> IfcAirToAirHeatRecoveryTypeEnum.USERDEFINED) OR
((PredefinedType = IfcAirToAirHeatRecoveryTypeEnum.USERDEFINED) AND EXISTS
(SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
('IFC4.IFCAIRTOAIRHEATRECOVERYTYPE' IN
TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcAirToAirHeatRecoveryType
SUBTYPE OF (IfcEnergyConversionDeviceType);
    PredefinedType : IfcAirToAirHeatRecoveryTypeEnum;
WHERE
    CorrectPredefinedType : (PredefinedType <>
IfcAirToAirHeatRecoveryTypeEnum.USERDEFINED) OR
((PredefinedType = IfcAirToAirHeatRecoveryTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcAlarm
SUBTYPE OF (IfcDistributionControlElement);
    PredefinedType : OPTIONAL IfcAlarmTypeEnum;

```

---

```

WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcAlarmTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcAlarmTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCALARMTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcAlarmType
    SUBTYPE OF (IfcDistributionControlElementType);
    PredefinedType : IfcAlarmTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <> IfcAlarmTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcAlarmTypeEnum.USERDEFINED) AND
        EXISTS(SELF\IfcElementType.ElementType));
    END_ENTITY;

ENTITY IfcAnnotation
    SUBTYPE OF (IfcProduct);
    INVERSE
        ContainedInStructure : SET [0:1] OF IfcRelContainedInSpatialStructure FOR
        RelatedElements;
    END_ENTITY;

ENTITY IfcAnnotationFillArea
    SUBTYPE OF (IfcGeometricRepresentationItem);
    OuterBoundary : IfcCurve;
    InnerBoundaries : OPTIONAL SET [1:?] OF IfcCurve;
    END_ENTITY;

ENTITY IfcApplication;
    ApplicationDeveloper : IfcOrganization;
    Version : IfcLabel;
    ApplicationFullName : IfcLabel;
    ApplicationIdentifier : IfcIdentifier;
    UNIQUE
        UR1 : ApplicationIdentifier;
        UR2 : ApplicationFullName, Version;
    END_ENTITY;

ENTITY IfcAppliedValue
    SUPERTYPE OF (ONEOF
        (IfcCostValue));

```

---

```

    Name : OPTIONAL IfcLabel;
    Description : OPTIONAL IfcText;
    AppliedValue : OPTIONAL IfcAppliedValueSelect;
    UnitBasis : OPTIONAL IfcMeasureWithUnit;
    ApplicableDate : OPTIONAL IfcDate;
    FixedUntilDate : OPTIONAL IfcDate;
    Category : OPTIONAL IfcLabel;
    Condition : OPTIONAL IfcLabel;
    ArithmeticOperator : OPTIONAL IfcArithmeticOperatorEnum;
    Components : OPTIONAL LIST [1:?] OF IfcAppliedValue;
INVERSE
    HasExternalReference : SET [0:?] OF IfcExternalReferenceRelationship FOR
RelatedResourceObjects;
END_ENTITY;

ENTITY IfcApproval;
    Identifier : OPTIONAL IfcIdentifier;
    Name : OPTIONAL IfcLabel;
    Description : OPTIONAL IfcText;
    TimeOfApproval : OPTIONAL IfcDateTime;
    Status : OPTIONAL IfcLabel;
    Level : OPTIONAL IfcLabel;
    Qualifier : OPTIONAL IfcText;
    RequestingApproval : OPTIONAL IfcActorSelect;
    GivingApproval : OPTIONAL IfcActorSelect;
INVERSE
    HasExternalReferences : SET [0:?] OF IfcExternalReferenceRelationship FOR
RelatedResourceObjects;
    ApprovedObjects : SET [0:?] OF IfcRelAssociatesApproval FOR RelatingApproval;
    ApprovedResources : SET [0:?] OF IfcResourceApprovalRelationship FOR
RelatingApproval;
    IsRelatedWith : SET [0:?] OF IfcApprovalRelationship FOR RelatedApprovals;
    Relates : SET [0:?] OF IfcApprovalRelationship FOR RelatingApproval;
WHERE
    HasIdentifierOrName : EXISTS (Identifier) OR EXISTS (Name);
END_ENTITY;

ENTITY IfcApprovalRelationship
    SUBTYPE OF (IfcResourceLevelRelationship);
    RelatingApproval : IfcApproval;
    RelatedApprovals : SET [1:?] OF IfcApproval;
END_ENTITY;

ENTITY IfcArbitraryClosedProfileDef

```

---

```

SUPERTYPE OF (ONEOF
  (IfcArbitraryProfileDefWithVoids))
SUBTYPE OF (IfcProfileDef);
  OuterCurve : IfcCurve;
WHERE
  WR1 : OuterCurve.Dim = 2;
  WR2 : NOT(' IFC4.IFCLINE' IN TYPEOF(OuterCurve));
  WR3 : NOT(' IFC4.IFCOFFSETCURVE2D' IN TYPEOF(OuterCurve));
END_ENTITY;

ENTITY IfcArbitraryOpenProfileDef
  SUPERTYPE OF (ONEOF
    (IfcCenterLineProfileDef))
  SUBTYPE OF (IfcProfileDef);
    Curve : IfcBoundedCurve;
  WHERE
    WR11 : (' IFC4.IFCCENTERLINEPROFILEDEF' IN TYPEOF(SELF)) OR
    (SELF\IfcProfileDef.ProfileType = IfcProfileTypeEnum.CURVE);
    WR12 : Curve.Dim = 2;
  END_ENTITY;

ENTITY IfcArbitraryProfileDefWithVoids
  SUBTYPE OF (IfcArbitraryClosedProfileDef);
    InnerCurves : SET [1:?] OF IfcCurve;
  WHERE
    WR1 : SELF\IfcProfileDef.ProfileType = AREA;
    WR2 : SIZEOF(QUERY(temp < * InnerCurves | temp.Dim < 2)) = 0;
    WR3 : SIZEOF(QUERY(temp < * InnerCurves | ' IFC4.IFCLINE' IN TYPEOF(temp))) = 0;
  END_ENTITY;

ENTITY IfcAsset
  SUBTYPE OF (IfcGroup);
    Identification : OPTIONAL IfcIdentifier;
    OriginalValue : OPTIONAL IfcCostValue;
    CurrentValue : OPTIONAL IfcCostValue;
    TotalReplacementCost : OPTIONAL IfcCostValue;
    Owner : OPTIONAL IfcActorSelect;
    User : OPTIONAL IfcActorSelect;
    ResponsiblePerson : OPTIONAL IfcPerson;
    IncorporationDate : OPTIONAL IfcDate;
    DepreciatedValue : OPTIONAL IfcCostValue;
  END_ENTITY;

ENTITY IfcAsymmetricIShapeProfileDef

```

---

```

SUBTYPE OF (IfcParameterizedProfileDef);
  BottomFlangeWidth : IfcPositiveLengthMeasure;
  OverallDepth : IfcPositiveLengthMeasure;
  WebThickness : IfcPositiveLengthMeasure;
  BottomFlangeThickness : IfcPositiveLengthMeasure;
  BottomFlangeFilletRadius : OPTIONAL IfcNonNegativeLengthMeasure;
  TopFlangeWidth : IfcPositiveLengthMeasure;
  TopFlangeThickness : OPTIONAL IfcPositiveLengthMeasure;
  TopFlangeFilletRadius : OPTIONAL IfcNonNegativeLengthMeasure;
  BottomFlangeEdgeRadius : OPTIONAL IfcNonNegativeLengthMeasure;
  BottomFlangeSlope : OPTIONAL IfcPlaneAngleMeasure;
  TopFlangeEdgeRadius : OPTIONAL IfcNonNegativeLengthMeasure;
  TopFlangeSlope : OPTIONAL IfcPlaneAngleMeasure;
WHERE
  ValidFlangeThickness : NOT (EXISTS(TopFlangeThickness)) OR
  ((BottomFlangeThickness + TopFlangeThickness) < OverallDepth);
  ValidWebThickness : (WebThickness < BottomFlangeWidth) AND (WebThickness <
TopFlangeWidth);
  ValidBottomFilletRadius : (NOT (EXISTS(BottomFlangeFilletRadius))) OR
  (BottomFlangeFilletRadius <= (BottomFlangeWidth - WebThickness)/2.);
  ValidTopFilletRadius : (NOT (EXISTS(TopFlangeFilletRadius))) OR
  (TopFlangeFilletRadius <= (TopFlangeWidth - WebThickness)/2.);
END_ENTITY;

```

```

ENTITY IfcAudioVisualAppliance
  SUBTYPE OF (IfcFlowTerminal);
    PredefinedType : OPTIONAL IfcAudioVisualApplianceTypeEnum;
WHERE
  CorrectPredefinedType : NOT (EXISTS(PredefinedType)) OR
  (PredefinedType <> IfcAudioVisualApplianceTypeEnum.USERDEFINED) OR
  ((PredefinedType = IfcAudioVisualApplianceTypeEnum.USERDEFINED) AND EXISTS
  (SELF\IfcObject.ObjectType));
  CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
  (' IFC4. IFCAUDIOVISUALAPPLIANCETYPE' IN
  TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcAudioVisualApplianceType
  SUBTYPE OF (IfcFlowTerminalType);
    PredefinedType : IfcAudioVisualApplianceTypeEnum;
WHERE
  CorrectPredefinedType : (PredefinedType <>
IfcAudioVisualApplianceTypeEnum.USERDEFINED) OR
  ((PredefinedType = IfcAudioVisualApplianceTypeEnum.USERDEFINED) AND

```



---

```

EXISTS (SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcAxis1Placement
  SUBTYPE OF (IfcPlacement);
    Axis : OPTIONAL IfcDirection;
  DERIVE
    Z : IfcDirection := NVL (IfcNormalise(Axis), IfcRepresentationItem() ||
IfcGeometricRepresentationItem () || IfcDirection([0.0,0.0,1.0]));
  WHERE
    AxisIs3D : (NOT (EXISTS (Axis))) OR (Axis.Dim = 3);
    LocationIs3D : SELF\IfcPlacement.Location.Dim = 3;
END_ENTITY;

ENTITY IfcAxis2Placement2D
  SUBTYPE OF (IfcPlacement);
    RefDirection : OPTIONAL IfcDirection;
  DERIVE
    P : LIST [2:2] OF IfcDirection := IfcBuild2Axes(RefDirection);
  WHERE
    RefDirIs2D : (NOT (EXISTS (RefDirection))) OR (RefDirection.Dim = 2);
    LocationIs2D : SELF\IfcPlacement.Location.Dim = 2;
END_ENTITY;

ENTITY IfcAxis2Placement3D
  SUBTYPE OF (IfcPlacement);
    Axis : OPTIONAL IfcDirection;
    RefDirection : OPTIONAL IfcDirection;
  DERIVE
    P : LIST [3:3] OF IfcDirection := IfcBuildAxes(Axis, RefDirection);
  WHERE
    LocationIs3D : SELF\IfcPlacement.Location.Dim = 3;
    AxisIs3D : (NOT (EXISTS (Axis))) OR (Axis.Dim = 3);
    RefDirIs3D : (NOT (EXISTS (RefDirection))) OR (RefDirection.Dim = 3);
    AxisToRefDirPosition : (NOT (EXISTS (Axis))) OR (NOT (EXISTS (RefDirection)))
OR (IfcCrossProduct(Axis, RefDirection).Magnitude > 0.0);
    AxisAndRefDirProvision : NOT ((EXISTS (Axis)) XOR (EXISTS (RefDirection)));
END_ENTITY;

ENTITY IfcBSplineCurve
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcBSplineCurveWithKnots))
  SUBTYPE OF (IfcBoundedCurve);
    Degree : IfcInteger;

```

---

```

    ControlPointsList : LIST [2:?] OF IfcCartesianPoint;
    CurveForm : IfcBSplineCurveForm;
    ClosedCurve : IfcLogical;
    SelfIntersect : IfcLogical;
DERIVE
    UpperIndexOnControlPoints : IfcInteger := (SIZEOF(ControlPointsList) - 1);
    ControlPoints : ARRAY [0:UpperIndexOnControlPoints] OF IfcCartesianPoint :=
IfcListToArray(ControlPointsList, 0, UpperIndexOnControlPoints);
WHERE
    SameDim : SIZEOF(QUERY(Temp <* ControlPointsList |
    Temp.Dim <> ControlPointsList[1].Dim))
= 0;
END_ENTITY;

ENTITY IfcBSplineCurveWithKnots
    SUPERTYPE OF (ONEOF
        (IfcRationalBSplineCurveWithKnots))
    SUBTYPE OF (IfcBSplineCurve);
    KnotMultiplicities : LIST [2:?] OF IfcInteger;
    Knots : LIST [2:?] OF IfcParameterValue;
    KnotSpec : IfcKnotType;
DERIVE
    UpperIndexOnKnots : IfcInteger := SIZEOF(Knots);
WHERE
    ConsistentBSpline : IfcConstraintsParamBSpline(Degree, UpperIndexOnKnots,
UpperIndexOnControlPoints, KnotMultiplicities, Knots);
    CorrespondingKnotLists : SIZEOF(KnotMultiplicities) = UpperIndexOnKnots;
END_ENTITY;

ENTITY IfcBSplineSurface
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcBSplineSurfaceWithKnots))
    SUBTYPE OF (IfcBoundedSurface);
    UDegree : IfcInteger;
    VDegree : IfcInteger;
    ControlPointsList : LIST [2:?] OF LIST [2:?] OF IfcCartesianPoint;
    SurfaceForm : IfcBSplineSurfaceForm;
    UClosed : IfcLogical;
    VClosed : IfcLogical;
    SelfIntersect : IfcLogical;
DERIVE
    UUpper : IfcInteger := SIZEOF(ControlPointsList) - 1;
    VUpper : IfcInteger := SIZEOF(ControlPointsList[1]) - 1;
    ControlPoints : ARRAY [0:UUpper] OF ARRAY [0:VUpper] OF IfcCartesianPoint :=

```

---

```

IfcMakeArrayOfArray(ControlPointsList,
0, UUpper, 0, VUpper);
END_ENTITY;

ENTITY IfcBSplineSurfaceWithKnots
  SUPERTYPE OF (ONEOF
    (IfcRationalBSplineSurfaceWithKnots))
  SUBTYPE OF (IfcBSplineSurface);
    UMult multiplicities : LIST [2:?] OF IfcInteger;
    VMult multiplicities : LIST [2:?] OF IfcInteger;
    UKnots : LIST [2:?] OF IfcParameterValue;
    VKnots : LIST [2:?] OF IfcParameterValue;
    KnotSpec : IfcKnotType;
  DERIVE
    KnotVUpper : IfcInteger := SIZEOF(VKnots);
    KnotUUpper : IfcInteger := SIZEOF(UKnots);
  WHERE
    UDirectionConstraints : IfcConstraintsParamBSpline (
      SELF\IfcBSplineSurface.UDegree, KnotUUpper,
      SELF\IfcBSplineSurface.UUpper, UMult multiplicities, UKnots);
    VDirectionConstraints : IfcConstraintsParamBSpline (
      SELF\IfcBSplineSurface.VDegree, KnotVUpper,
      SELF\IfcBSplineSurface.VUpper, VMult multiplicities, VKnots);
    CorrespondingULists : SIZEOF(UMult multiplicities) = KnotUUpper;
    CorrespondingVLists : SIZEOF(VMult multiplicities) = KnotVUpper;
END_ENTITY;

ENTITY IfcBeam
  SUPERTYPE OF (ONEOF
    (IfcBeamStandardCase))
  SUBTYPE OF (IfcBuildingElement);
    PredefinedType : OPTIONAL IfcBeamTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcBeamTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcBeamTypeEnum.USERDEFINED) AND EXISTS
      (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCBEAMTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcBeamStandardCase
  SUBTYPE OF (IfcBeam);
  WHERE

```

---

```

        HasMaterialProfileSetUsage      :      SIZEOF      (QUERY(temp      < *      USEDIN(SELF,
' IFC4. IFCRELASSOCIATES.RELATEDOBJECTS' ) |
                (' IFC4. IFCRELASSOCIATESMATERIAL' IN TYPEOF(temp)) AND
                (' IFC4. IFCMATERIALPROFILESETUSAGE'                                     IN
TYPEOF(temp.RelatingMaterial))
                )) = 1;
END_ENTITY;

ENTITY IfcBeamType
SUBTYPE OF (IfcBuildingElementType);
    PredefinedType : IfcBeamTypeEnum;
WHERE
    CorrectPredefinedType : (PredefinedType <> IfcBeamTypeEnum.USERDEFINED) OR
((PredefinedType      =      IfcBeamTypeEnum.USERDEFINED)      AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcBlobTexture
SUBTYPE OF (IfcSurfaceTexture);
    RasterFormat : IfcIdentifier;
    RasterCode : IfcBinary;
WHERE
    SupportedRasterFormat : SELF.RasterFormat IN ['BMP', 'JPG', 'GIF', 'PNG'];
    RasterCodeByteStream : BLENGTH(RasterCode) MOD 8 = 0;
END_ENTITY;

ENTITY IfcBlock
SUBTYPE OF (IfcCsgPrimitive3D);
    XLength : IfcPositiveLengthMeasure;
    YLength : IfcPositiveLengthMeasure;
    ZLength : IfcPositiveLengthMeasure;
END_ENTITY;

ENTITY IfcBoiler
SUBTYPE OF (IfcEnergyConversionDevice);
    PredefinedType : OPTIONAL IfcBoilerTypeEnum;
WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
(PredefinedType <> IfcBoilerTypeEnum.USERDEFINED) OR
((PredefinedType      =      IfcBoilerTypeEnum.USERDEFINED)      AND      EXISTS
(SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
(' IFC4. IFCBOILERTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

---

```

ENTITY IfcBoilerType
  SUBTYPE OF (IfcEnergyConversionDeviceType);
    PredefinedType : IfcBoilerTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcBoilerTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcBoilerTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcBooleanClippingResult
  SUBTYPE OF (IfcBooleanResult);
  WHERE
    FirstOperandType : (' IFC4. IFC4SWEPTAREASOLID' IN TYPEOF(FirstOperand)) OR
    (' IFC4. IFC4SWEPTDISCSOLID' IN TYPEOF(FirstOperand)) OR
    (' IFC4. IFC4BOOLEANCLIPPINGRESULT' IN TYPEOF(FirstOperand));
    SecondOperandType : (' IFC4. IFC4HALFSPACESOLID' IN TYPEOF(SecondOperand));
    OperatorType : Operator = DIFFERENCE;
END_ENTITY;

```

```

ENTITY IfcBooleanResult
  SUPERTYPE OF (ONEOF
    (IfcBooleanClippingResult))
  SUBTYPE OF (IfcGeometricRepresentationItem);
    Operator : IfcBooleanOperator;
    FirstOperand : IfcBooleanOperand;
    SecondOperand : IfcBooleanOperand;
  DERIVE
    Dim : IfcDimensionCount := FirstOperand.Dim;
  WHERE
    SameDim : FirstOperand.Dim = SecondOperand.Dim;
    FirstOperandClosed : NOT(' IFC4. IFCTESSELLATEDFACESET' IN TYPEOF(FirstOperand))
OR (EXISTS(FirstOperand.Closed) AND FirstOperand.Closed);
    SecondOperandClosed : NOT(' IFC4. IFCTESSELLATEDFACESET' IN
TYPEOF(SecondOperand)) OR (EXISTS(SecondOperand.Closed) AND
SecondOperand.Closed);
END_ENTITY;

```

```

ENTITY IfcBoundaryCondition
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcBoundaryEdgeCondition
    , IfcBoundaryFaceCondition
    , IfcBoundaryNodeCondition));
  Name : OPTIONAL IfcLabel;

```

---

```

END_ENTITY;

ENTITY IfcBoundaryCurve
  SUPERTYPE OF (ONEOF
    (IfcOuterBoundaryCurve))
  SUBTYPE OF (IfcCompositeCurveOnSurface);
  WHERE
    IsClosed : SELF\IfcCompositeCurve.ClosedCurve;
END_ENTITY;

ENTITY IfcBoundaryEdgeCondition
  SUBTYPE OF (IfcBoundaryCondition);
  TranslationalStiffnessByLengthX : OPTIONAL
IfcModulusOfTranslationalSubgradeReactionSelect;
  TranslationalStiffnessByLengthY : OPTIONAL
IfcModulusOfTranslationalSubgradeReactionSelect;
  TranslationalStiffnessByLengthZ : OPTIONAL
IfcModulusOfTranslationalSubgradeReactionSelect;
  RotationalStiffnessByLengthX : OPTIONAL
IfcModulusOfRotationalSubgradeReactionSelect;
  RotationalStiffnessByLengthY : OPTIONAL
IfcModulusOfRotationalSubgradeReactionSelect;
  RotationalStiffnessByLengthZ : OPTIONAL
IfcModulusOfRotationalSubgradeReactionSelect;
END_ENTITY;

ENTITY IfcBoundaryFaceCondition
  SUBTYPE OF (IfcBoundaryCondition);
  TranslationalStiffnessByAreaX : OPTIONAL IfcModulusOfSubgradeReactionSelect;
  TranslationalStiffnessByAreaY : OPTIONAL IfcModulusOfSubgradeReactionSelect;
  TranslationalStiffnessByAreaZ : OPTIONAL IfcModulusOfSubgradeReactionSelect;
END_ENTITY;

ENTITY IfcBoundaryNodeCondition
  SUPERTYPE OF (ONEOF
    (IfcBoundaryNodeConditionWarping))
  SUBTYPE OF (IfcBoundaryCondition);
  TranslationalStiffnessX : OPTIONAL IfcTranslationalStiffnessSelect;
  TranslationalStiffnessY : OPTIONAL IfcTranslationalStiffnessSelect;
  TranslationalStiffnessZ : OPTIONAL IfcTranslationalStiffnessSelect;
  RotationalStiffnessX : OPTIONAL IfcRotationalStiffnessSelect;
  RotationalStiffnessY : OPTIONAL IfcRotationalStiffnessSelect;
  RotationalStiffnessZ : OPTIONAL IfcRotationalStiffnessSelect;
END_ENTITY;

```

---

```

ENTITY IfcBoundaryNodeConditionWarping
  SUBTYPE OF (IfcBoundaryNodeCondition);
    WarpingStiffness : OPTIONAL IfcWarpingStiffnessSelect;
END_ENTITY;

ENTITY IfcBoundedCurve
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcBSplineCurve
      , IfcCompositeCurve
      , IfcIndexedPolyCurve
      , IfcPolyline
      , IfcTrimmedCurve))
  SUBTYPE OF (IfcCurve);
END_ENTITY;

ENTITY IfcBoundedSurface
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcBSplineSurface
      , IfcCurveBoundedPlane
      , IfcCurveBoundedSurface
      , IfcRectangularTrimmedSurface))
  SUBTYPE OF (IfcSurface);
END_ENTITY;

ENTITY IfcBoundingBox
  SUBTYPE OF (IfcGeometricRepresentationItem);
    Corner : IfcCartesianPoint;
    XDim : IfcPositiveLengthMeasure;
    YDim : IfcPositiveLengthMeasure;
    ZDim : IfcPositiveLengthMeasure;
  DERIVE
    Dim : IfcDimensionCount := 3;
END_ENTITY;

ENTITY IfcBoxedHalfSpace
  SUBTYPE OF (IfcHalfSpaceSolid);
    Enclosure : IfcBoundingBox;
  WHERE
    UnboundedSurface : NOT (' IFC4. IFCCURVEBOUNDEDPLANE' IN
      TYPEOF (SELF\IfcHalfSpaceSolid.BaseSurface));
END_ENTITY;

ENTITY IfcBuilding

```

---

```

SUBTYPE OF (IfcSpatialStructureElement);
    ElevationOfRefHeight : OPTIONAL IfcLengthMeasure;
    ElevationOfTerrain : OPTIONAL IfcLengthMeasure;
    BuildingAddress : OPTIONAL IfcPostalAddress;
END_ENTITY;

ENTITY IfcBuildingElement
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcBeam
        , IfcBuildingElementProxy
        , IfcChimney
        , IfcColumn
        , IfcCovering
        , IfcCurtainWall
        , IfcDoor
        , IfcFooting
        , IfcMember
        , IfcPile
        , IfcPlate
        , IfcRailing
        , IfcRamp
        , IfcRampFlight
        , IfcRoof
        , IfcShadingDevice
        , IfcSlab
        , IfcStair
        , IfcStairFlight
        , IfcWall
        , IfcWindow))
    SUBTYPE OF (IfcElement);
    WHERE
        MaxOneMaterialAssociation      :      SIZEOF      (QUERY(temp      <*>
SELF\IfcObjectDefinition.HasAssociations |
    ' IFC4. IFCRELASSOCIATESMATERIAL' IN TYPEOF(temp)
    )) <= 1;
END_ENTITY;

ENTITY IfcBuildingElementPart
    SUBTYPE OF (IfcElementComponent);
    PredefinedType : OPTIONAL IfcBuildingElementPartTypeEnum;
    WHERE
        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
        (PredefinedType <> IfcBuildingElementPartTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcBuildingElementPartTypeEnum.USERDEFINED) AND EXISTS

```



---

```

(SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
(' IFC4. IFCBUILDINGELEMENTPARTTYPE'                                IN
TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcBuildingElementPartType
    SUBTYPE OF (IfcElementComponentType);
    PredefinedType : IfcBuildingElementPartTypeEnum;
    WHERE
        CorrectPredefinedType          :          (PredefinedType          <>
IfcBuildingElementPartTypeEnum.USERDEFINED) OR
((PredefinedType = IfcBuildingElementPartTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcBuildingElementProxy
    SUBTYPE OF (IfcBuildingElement);
    PredefinedType : OPTIONAL IfcBuildingElementProxyTypeEnum;
    WHERE
        HasObjectName : EXISTS(SELF\IfcRoot.Name);
        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
(PredefinedType <> IfcBuildingElementProxyTypeEnum.USERDEFINED) OR
((PredefinedType = IfcBuildingElementProxyTypeEnum.USERDEFINED) AND EXISTS
(SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
(' IFC4. IFCBUILDINGELEMENTPROXYTYPE'                                IN
TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcBuildingElementProxyType
    SUBTYPE OF (IfcBuildingElementType);
    PredefinedType : IfcBuildingElementProxyTypeEnum;
    WHERE
        CorrectPredefinedType          :          (PredefinedType          <>
IfcBuildingElementProxyTypeEnum.USERDEFINED) OR
((PredefinedType = IfcBuildingElementProxyTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcBuildingElementType
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcBeamType
        , IfcBuildingElementProxyType

```

---

```

    , IfcChimneyType
    , IfcColumnType
    , IfcCoveringType
    , IfcCurtainWallType
    , IfcDoorType
    , IfcFootingType
    , IfcMemberType
    , IfcPileType
    , IfcPlateType
    , IfcRailingType
    , IfcRampFlightType
    , IfcRampType
    , IfcRoofType
    , IfcShadingDeviceType
    , IfcSlabType
    , IfcStairFlightType
    , IfcStairType
    , IfcWallType
    , IfcWindowType))
SUBTYPE OF (IfcElementType);
END_ENTITY;

ENTITY IfcBuildingStorey
SUBTYPE OF (IfcSpatialStructureElement);
    Elevation : OPTIONAL IfcLengthMeasure;
END_ENTITY;

ENTITY IfcBuildingSystem
SUBTYPE OF (IfcSystem);
    PredefinedType : OPTIONAL IfcBuildingSystemTypeEnum;
    LongName : OPTIONAL IfcLabel;
END_ENTITY;

ENTITY IfcBurner
SUBTYPE OF (IfcEnergyConversionDevice);
    PredefinedType : OPTIONAL IfcBurnerTypeEnum;
WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcBurnerTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcBurnerTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCBURNERTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

---

```

ENTITY IfcBurnerType
  SUBTYPE OF (IfcEnergyConversionDeviceType);
    PredefinedType : IfcBurnerTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcBurnerTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcBurnerTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcCShapeProfileDef
  SUBTYPE OF (IfcParameterizedProfileDef);
    Depth : IfcPositiveLengthMeasure;
    Width : IfcPositiveLengthMeasure;
    WallThickness : IfcPositiveLengthMeasure;
    Girth : IfcPositiveLengthMeasure;
    InternalFilletRadius : OPTIONAL IfcNonNegativeLengthMeasure;
  WHERE
    ValidGirth : Girth < (Depth / 2.);
    ValidInternalFilletRadius : NOT(EXISTS(InternalFilletRadius)) OR
    ((InternalFilletRadius <= Width/2. - WallThickness) AND (InternalFilletRadius <=
    Depth/2. - WallThickness));
    ValidWallThickness : (WallThickness < Width/2.) AND (WallThickness < Depth/2.);
END_ENTITY;

```

```

ENTITY IfcCableCarrierFitting
  SUBTYPE OF (IfcFlowFitting);
    PredefinedType : OPTIONAL IfcCableCarrierFittingTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcCableCarrierFittingTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcCableCarrierFittingTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCCABLECARRIERFITTINGTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcCableCarrierFittingType
  SUBTYPE OF (IfcFlowFittingType);
    PredefinedType : IfcCableCarrierFittingTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <>
    IfcCableCarrierFittingTypeEnum.USERDEFINED) OR

```

---

```

((PredefinedType      =      IfcCableCarrierFittingTypeEnum.USERDEFINED)      AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcCableCarrierSegment
  SUBTYPE OF (IfcFlowSegment);
    PredefinedType : OPTIONAL IfcCableCarrierSegmentTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcCableCarrierSegmentTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcCableCarrierSegmentTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    (' IFC4. IFCCABLECARRIERSEGMENTTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcCableCarrierSegmentType
  SUBTYPE OF (IfcFlowSegmentType);
    PredefinedType : IfcCableCarrierSegmentTypeEnum;
  WHERE
    CorrectPredefinedType      :      (PredefinedType      <>
IfcCableCarrierSegmentTypeEnum.USERDEFINED) OR
    ((PredefinedType      =      IfcCableCarrierSegmentTypeEnum.USERDEFINED)      AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcCableFitting
  SUBTYPE OF (IfcFlowFitting);
    PredefinedType : OPTIONAL IfcCableFittingTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcCableFittingTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcCableFittingTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    (' IFC4. IFCCABLEFITTINGTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcCableFittingType
  SUBTYPE OF (IfcFlowFittingType);
    PredefinedType : IfcCableFittingTypeEnum;
  WHERE

```

---

```

        CorrectPredefinedType          :          (PredefinedType          <>
IfcCableFittingTypeEnum.USERDEFINED) OR
((PredefinedType          =          IfcCableFittingTypeEnum.USERDEFINED)          AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcCableSegment
  SUBTYPE OF (IfcFlowSegment);
    PredefinedType : OPTIONAL IfcCableSegmentTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcCableSegmentTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcCableSegmentTypeEnum.USERDEFINED)          AND          EXISTS
(SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    (' IFC4. IFCCABLESEGMENTTYPE'          IN
TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcCableSegmentType
  SUBTYPE OF (IfcFlowSegmentType);
    PredefinedType : IfcCableSegmentTypeEnum;
  WHERE
    CorrectPredefinedType          :          (PredefinedType          <>
IfcCableSegmentTypeEnum.USERDEFINED) OR
((PredefinedType          =          IfcCableSegmentTypeEnum.USERDEFINED)          AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcCartesianPoint
  SUBTYPE OF (IfcPoint);
    Coordinates : LIST [1:3] OF IfcLengthMeasure;
  DERIVE
    Dim : IfcDimensionCount := HIINDEX(Coordinates);
  WHERE
    CP2Dor3D : HIINDEX(Coordinates) >= 2;
END_ENTITY;

```

```

ENTITY IfcCartesianPointList
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcCartesianPointList2D
    , IfcCartesianPointList3D))
  SUBTYPE OF (IfcGeometricRepresentationItem);
  DERIVE

```

---

```

    Dim : IfcDimensionCount := IfcPointListDim(SELF);
END_ENTITY;

ENTITY IfcCartesianPointList2D
  SUBTYPE OF (IfcCartesianPointList);
    CoordList : LIST [1:?] OF LIST [2:2] OF IfcLengthMeasure;
END_ENTITY;

ENTITY IfcCartesianPointList3D
  SUBTYPE OF (IfcCartesianPointList);
    CoordList : LIST [1:?] OF LIST [3:3] OF IfcLengthMeasure;
END_ENTITY;

ENTITY IfcCartesianTransformationOperator
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcCartesianTransformationOperator2D
    , IfcCartesianTransformationOperator3D))
  SUBTYPE OF (IfcGeometricRepresentationItem);
    Axis1 : OPTIONAL IfcDirection;
    Axis2 : OPTIONAL IfcDirection;
    LocalOrigin : IfcCartesianPoint;
    Scale : OPTIONAL IfcReal;
  DERIVE
    Scl : IfcReal := NVL(Scale, 1.0);
    Dim : IfcDimensionCount := LocalOrigin.Dim;
  WHERE
    ScaleGreaterZero : Scl > 0.0;
END_ENTITY;

ENTITY IfcCartesianTransformationOperator2D
  SUPERTYPE OF (ONEOF
    (IfcCartesianTransformationOperator2DnonUniform))
  SUBTYPE OF (IfcCartesianTransformationOperator);
  DERIVE
    U      :      LIST      [2:2]      OF      IfcDirection      :=
IfcBaseAxis(2, SELF\IfcCartesianTransformationOperator.Axis1,
SELF\IfcCartesianTransformationOperator.Axis2, ?);
  WHERE
    DimEqual2 : SELF\IfcCartesianTransformationOperator.Dim = 2;
    Axis1Is2D : NOT (EXISTS (SELF\IfcCartesianTransformationOperator.Axis1)) OR
(SELF\IfcCartesianTransformationOperator.Axis1.Dim = 2);
    Axis2Is2D : NOT (EXISTS (SELF\IfcCartesianTransformationOperator.Axis2)) OR
(SELF\IfcCartesianTransformationOperator.Axis2.Dim = 2);
END_ENTITY;

```

---

```

ENTITY IfcCartesianTransformationOperator2DnonUniform
  SUBTYPE OF (IfcCartesianTransformationOperator2D);
    Scale2 : OPTIONAL IfcReal;
  DERIVE
    Sc12 : IfcReal := NVL(Scale2, SELF\IfcCartesianTransformationOperator.Sc1);
  WHERE
    Scale2GreaterZero : Sc12 > 0.0;
END_ENTITY;

```

```

ENTITY IfcCartesianTransformationOperator3D
  SUPERTYPE OF (ONEOF
    (IfcCartesianTransformationOperator3DnonUniform))
  SUBTYPE OF (IfcCartesianTransformationOperator);
    Axis3 : OPTIONAL IfcDirection;
  DERIVE
    U      :      LIST      [3:3]      OF      IfcDirection      :=
IfcBaseAxis(3, SELF\IfcCartesianTransformationOperator.Axis1,
SELF\IfcCartesianTransformationOperator.Axis2, Axis3);
  WHERE
    DimIs3D : SELF\IfcCartesianTransformationOperator.Dim = 3;
    Axis1Is3D : NOT (EXISTS(SELF\IfcCartesianTransformationOperator.Axis1)) OR
(SELF\IfcCartesianTransformationOperator.Axis1.Dim = 3);
    Axis2Is3D : NOT (EXISTS(SELF\IfcCartesianTransformationOperator.Axis2)) OR
(SELF\IfcCartesianTransformationOperator.Axis2.Dim = 3);
    Axis3Is3D : NOT (EXISTS(Axis3)) OR (Axis3.Dim = 3);
END_ENTITY;

```

```

ENTITY IfcCartesianTransformationOperator3DnonUniform
  SUBTYPE OF (IfcCartesianTransformationOperator3D);
    Scale2 : OPTIONAL IfcReal;
    Scale3 : OPTIONAL IfcReal;
  DERIVE
    Sc12 : IfcReal := NVL(Scale2, SELF\IfcCartesianTransformationOperator.Sc1);
    Sc13 : IfcReal := NVL(Scale3, SELF\IfcCartesianTransformationOperator.Sc1);
  WHERE
    Scale2GreaterZero : Sc12 > 0.0;
    Scale3GreaterZero : Sc13 > 0.0;
END_ENTITY;

```

```

ENTITY IfcCenterLineProfileDef
  SUBTYPE OF (IfcArbitraryOpenProfileDef);
    Thickness : IfcPositiveLengthMeasure;
END_ENTITY;

```

---

```

ENTITY IfcChiller
  SUBTYPE OF (IfcEnergyConversionDevice);
    PredefinedType : OPTIONAL IfcChillerTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcChillerTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcChillerTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCCHILLERTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcChillerType
  SUBTYPE OF (IfcEnergyConversionDeviceType);
    PredefinedType : IfcChillerTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcChillerTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcChillerTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcChimney
  SUBTYPE OF (IfcBuildingElement);
    PredefinedType : OPTIONAL IfcChimneyTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcChimneyTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcChimneyTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCCHIMNEYTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcChimneyType
  SUBTYPE OF (IfcBuildingElementType);
    PredefinedType : IfcChimneyTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcChimneyTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcChimneyTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcCircle

```



---

```

SUBTYPE OF (IfcConic);
    Radius : IfcPositiveLengthMeasure;
END_ENTITY;

ENTITY IfcCircleHollowProfileDef
    SUBTYPE OF (IfcCircleProfileDef);
        WallThickness : IfcPositiveLengthMeasure;
    WHERE
        WR1 : WallThickness < SELF\IfcCircleProfileDef.Radius;
END_ENTITY;

ENTITY IfcCircleProfileDef
    SUPERTYPE OF (ONEOF
        (IfcCircleHollowProfileDef))
    SUBTYPE OF (IfcParameterizedProfileDef);
        Radius : IfcPositiveLengthMeasure;
END_ENTITY;

ENTITY IfcCivilElement
    SUBTYPE OF (IfcElement);
END_ENTITY;

ENTITY IfcCivilElementType
    SUBTYPE OF (IfcElementType);
END_ENTITY;

ENTITY IfcClassification
    SUBTYPE OF (IfcExternalInformation);
        Source : OPTIONAL IfcLabel;
        Edition : OPTIONAL IfcLabel;
        EditionDate : OPTIONAL IfcDate;
        Name : IfcLabel;
        Description : OPTIONAL IfcText;
        Location : OPTIONAL IfcURIReference;
        ReferenceTokens : OPTIONAL LIST [1:?] OF IfcIdentifier;
    INVERSE
        ClassificationForObjects : SET [0:?] OF IfcRelAssociatesClassification FOR
        RelatingClassification;
        HasReferences : SET [0:?] OF IfcClassificationReference FOR ReferencedSource;
END_ENTITY;

ENTITY IfcClassificationReference
    SUBTYPE OF (IfcExternalReference);
        ReferencedSource : OPTIONAL IfcClassificationReferenceSelect;

```

---

```

        Description : OPTIONAL IfcText;
        Sort : OPTIONAL IfcIdentifier;
    INVERSE
        ClassificationRefForObjects : SET [0:?] OF IfcRelAssociatesClassification FOR
    RelatingClassification;
        HasReferences : SET [0:?] OF IfcClassificationReference FOR ReferencedSource;
    END_ENTITY;

ENTITY IfcClosedShell
    SUBTYPE OF (IfcConnectedFaceSet);
    END_ENTITY;

ENTITY IfcCoil
    SUBTYPE OF (IfcEnergyConversionDevice);
        PredefinedType : OPTIONAL IfcCoilTypeEnum;
    WHERE
        CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
        (PredefinedType <> IfcCoilTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcCoilTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject.ObjectType));
        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
        ('IFC4.IFCCOILTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
    END_ENTITY;

ENTITY IfcCoilType
    SUBTYPE OF (IfcEnergyConversionDeviceType);
        PredefinedType : IfcCoilTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <> IfcCoilTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcCoilTypeEnum.USERDEFINED) AND
        EXISTS (SELF\IfcElementType.ElementType));
    END_ENTITY;

ENTITY IfcColourRgb
    SUBTYPE OF (IfcColourSpecification);
        Red : IfcNormalisedRatioMeasure;
        Green : IfcNormalisedRatioMeasure;
        Blue : IfcNormalisedRatioMeasure;
    END_ENTITY;

ENTITY IfcColourRgbList
    SUBTYPE OF (IfcPresentationItem);
        ColourList : LIST [1:?] OF LIST [3:3] OF IfcNormalisedRatioMeasure;
    END_ENTITY;

```

---

```

ENTITY IfcColourSpecification
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcColourRgb))
  SUBTYPE OF (IfcPresentationItem);
  Name : OPTIONAL IfcLabel;
END_ENTITY;

ENTITY IfcColumn
  SUPERTYPE OF (ONEOF
    (IfcColumnStandardCase))
  SUBTYPE OF (IfcBuildingElement);
  PredefinedType : OPTIONAL IfcColumnTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcColumnTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcColumnTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCCOLUMNTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcColumnStandardCase
  SUBTYPE OF (IfcColumn);
  WHERE
    HasMaterialProfileSetUsage : SIZEOF (QUERY(temp < * USEDIN(SELF,
'IFC4.IFCRELASSOCIATES.RELATEDOBJECTS') |
      ('IFC4.IFCRELASSOCIATESMATERIAL' IN TYPEOF(temp)) AND
      ('IFC4.IFCMATERIALPROFILESETUSAGE' IN
      TYPEOF(temp.RelatingMaterial))
      )) = 1;
END_ENTITY;

ENTITY IfcColumnType
  SUBTYPE OF (IfcBuildingElementType);
  PredefinedType : IfcColumnTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcColumnTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcColumnTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcCommunicationsAppliance
  SUBTYPE OF (IfcFlowTerminal);

```

---

```

    PredefinedType : OPTIONAL IfcCommunicationsApplianceTypeEnum;
WHERE
    CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
    (PredefinedType <> IfcCommunicationsApplianceTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcCommunicationsApplianceTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCCOMMUNICATIONSAPPLIANCETYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcCommunicationsApplianceType
    SUBTYPE OF (IfcFlowTerminalType);
    PredefinedType : IfcCommunicationsApplianceTypeEnum;
WHERE
    CorrectPredefinedType : (PredefinedType <>
    IfcCommunicationsApplianceTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcCommunicationsApplianceTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcComplexProperty
    SUBTYPE OF (IfcProperty);
    UsageName : IfcIdentifier;
    HasProperties : SET [1:?] OF IfcProperty;
WHERE
    WR21 : SIZEOF(QUERY(temp <* HasProperties | SELF :=: temp)) = 0;
    WR22 : IfcUniquePropertyName(HasProperties);
END_ENTITY;

```

```

ENTITY IfcComplexPropertyTemplate
    SUBTYPE OF (IfcPropertyTemplate);
    UsageName : OPTIONAL IfcLabel;
    TemplateType : OPTIONAL IfcComplexPropertyTemplateTypeEnum;
    HasPropertyTemplates : OPTIONAL SET [1:?] OF IfcPropertyTemplate;
WHERE
    UniquePropertyNames : IfcUniquePropertyTemplateName(HasPropertyTemplates);
    NoSelfReference : SIZEOF(QUERY(temp <* HasPropertyTemplates | SELF :=: temp))
    = 0;
END_ENTITY;

```

```

ENTITY IfcCompositeCurve
    SUPERTYPE OF (ONEOF
    (IfcCompositeCurveOnSurface))

```

---

```

SUBTYPE OF (IfcBoundedCurve);
  Segments : LIST [1:?] OF IfcCompositeCurveSegment;
  SelfIntersect : IfcLogical;
DERIVE
  NSegments : IfcInteger := SIZEOF(Segments);
  ClosedCurve : IfcLogical := Segments[NSegments].Transition <> Discontinuous;
WHERE
  CurveContinuous : ((NOT ClosedCurve) AND (SIZEOF(QUERY(Temp < * Segments |
Temp.Transition = Discontinuous)) = 1)) OR ((ClosedCurve) AND (SIZEOF(QUERY(Temp
< * Segments | Temp.Transition = Discontinuous)) = 0));
  SameDim : SIZEOF( QUERY( Temp < * Segments | Temp.Dim <> Segments[1].Dim)) = 0;
END_ENTITY;

```

```

ENTITY IfcCompositeCurveOnSurface
  SUPERTYPE OF (ONEOF
    (IfcBoundaryCurve))
  SUBTYPE OF (IfcCompositeCurve);
  DERIVE
    BasisSurface : SET [0:1] OF IfcSurface := IfcGetBasisSurface(SELF);
  WHERE
    SameSurface : SIZEOF(BasisSurface) > 0;
END_ENTITY;

```

```

ENTITY IfcCompositeCurveSegment
  SUPERTYPE OF (ONEOF
    (IfcReparametrisedCompositeCurveSegment))
  SUBTYPE OF (IfcGeometricRepresentationItem);
  Transition : IfcTransitionCode;
  SameSense : IfcBoolean;
  ParentCurve : IfcCurve;
  DERIVE
    Dim : IfcDimensionCount := ParentCurve.Dim;
  INVERSE
    UsingCurves : SET [1:?] OF IfcCompositeCurve FOR Segments;
  WHERE
    ParentIsBoundedCurve : (' IFC4. IFCBOUNDED_CURVE' IN TYPEOF(ParentCurve));
END_ENTITY;

```

```

ENTITY IfcCompositeProfileDef
  SUBTYPE OF (IfcProfileDef);
  Profiles : SET [2:?] OF IfcProfileDef;
  Label : OPTIONAL IfcLabel;
  WHERE
    InvariantProfileType : SIZEOF(QUERY(temp < * Profiles | temp.ProfileType <>

```

---

```

Profiles[1].ProfileType)) = 0;
    NoRecursion : SIZEOF(QUERY(temp <* Profiles | ' IFC4. IFCCOMPOSITEPROFILEDEF' IN
    TYPEOF(temp))) = 0;
END_ENTITY;

```

```

ENTITY IfcCompressor
    SUBTYPE OF (IfcFlowMovingDevice);
    PredefinedType : OPTIONAL IfcCompressorTypeEnum;
    WHERE
        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
        (PredefinedType <> IfcCompressorTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcCompressorTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject.ObjectType));
        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
        (' IFC4. IFCCOMPRESSORTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcCompressorType
    SUBTYPE OF (IfcFlowMovingDeviceType);
    PredefinedType : IfcCompressorTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <> IfcCompressorTypeEnum.USERDEFINED)
    OR
        ((PredefinedType = IfcCompressorTypeEnum.USERDEFINED) AND
        EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcCondenser
    SUBTYPE OF (IfcEnergyConversionDevice);
    PredefinedType : OPTIONAL IfcCondenserTypeEnum;
    WHERE
        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
        (PredefinedType <> IfcCondenserTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcCondenserTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject.ObjectType));
        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
        (' IFC4. IFCCONDENSERTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcCondenserType
    SUBTYPE OF (IfcEnergyConversionDeviceType);
    PredefinedType : IfcCondenserTypeEnum;
    WHERE

```

---

```

        CorrectPredefinedType : (PredefinedType <> IfcCondenserTypeEnum.USERDEFINED)
OR
((PredefinedType          =          IfcCondenserTypeEnum.USERDEFINED)          AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcConic
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcCircle
    , IfcEllipse))
  SUBTYPE OF (IfcCurve);
  Position : IfcAxis2Placement;
END_ENTITY;

```

```

ENTITY IfcConnectedFaceSet
  SUPERTYPE OF (ONEOF
    (IfcClosedShell
    , IfcOpenShell))
  SUBTYPE OF (IfcTopologicalRepresentationItem);
  CfsFaces : SET [1:?] OF IfcFace;
END_ENTITY;

```

```

ENTITY IfcConnectionCurveGeometry
  SUBTYPE OF (IfcConnectionGeometry);
  CurveOnRelatingElement : IfcCurveOrEdgeCurve;
  CurveOnRelatedElement : OPTIONAL IfcCurveOrEdgeCurve;
END_ENTITY;

```

```

ENTITY IfcConnectionGeometry
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcConnectionCurveGeometry
    , IfcConnectionPointGeometry
    , IfcConnectionSurfaceGeometry
    , IfcConnectionVolumeGeometry));
END_ENTITY;

```

```

ENTITY IfcConnectionPointEccentricity
  SUBTYPE OF (IfcConnectionPointGeometry);
  EccentricityInX : OPTIONAL IfcLengthMeasure;
  EccentricityInY : OPTIONAL IfcLengthMeasure;
  EccentricityInZ : OPTIONAL IfcLengthMeasure;
END_ENTITY;

```

```

ENTITY IfcConnectionPointGeometry

```

---

```

SUPERTYPE OF (ONEOF
    (IfcConnectionPointEccentricity))
SUBTYPE OF (IfcConnectionGeometry);
    PointOnRelatingElement : IfcPointOrVertexPoint;
    PointOnRelatedElement : OPTIONAL IfcPointOrVertexPoint;
END_ENTITY;

ENTITY IfcConnectionSurfaceGeometry
    SUBTYPE OF (IfcConnectionGeometry);
        SurfaceOnRelatingElement : IfcSurfaceOrFaceSurface;
        SurfaceOnRelatedElement : OPTIONAL IfcSurfaceOrFaceSurface;
END_ENTITY;

ENTITY IfcConnectionVolumeGeometry
    SUBTYPE OF (IfcConnectionGeometry);
        VolumeOnRelatingElement : IfcSolidOrShell;
        VolumeOnRelatedElement : OPTIONAL IfcSolidOrShell;
END_ENTITY;

ENTITY IfcConstraint
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcMetric
            , IfcObjective));
    Name : IfcLabel;
    Description : OPTIONAL IfcText;
    ConstraintGrade : IfcConstraintEnum;
    ConstraintSource : OPTIONAL IfcLabel;
    CreatingActor : OPTIONAL IfcActorSelect;
    CreationTime : OPTIONAL IfcDateTime;
    UserDefinedGrade : OPTIONAL IfcLabel;
    INVERSE
        HasExternalReferences : SET [0:?] OF IfcExternalReferenceRelationship FOR
RelatedResourceObjects;
        PropertiesForConstraint : SET [0:?] OF IfcResourceConstraintRelationship FOR
RelatingConstraint;
    WHERE
        WR11 : (ConstraintGrade <> IfcConstraintEnum.USERDEFINED) OR
        ((ConstraintGrade = IfcConstraintEnum.USERDEFINED) AND
EXISTS(SELF\IfcConstraint.UserDefinedGrade));
END_ENTITY;

ENTITY IfcConstructionEquipmentResource
    SUBTYPE OF (IfcConstructionResource);
        PredefinedType : OPTIONAL IfcConstructionEquipmentResourceTypeEnum;

```



---

```

WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcConstructionEquipmentResourceTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcConstructionEquipmentResourceTypeEnum.USERDEFINED) AND
EXISTS (SELF\IfcObject.ObjectType));
END_ENTITY;

ENTITY IfcConstructionEquipmentResourceType
    SUBTYPE OF (IfcConstructionResourceType);
    PredefinedType : IfcConstructionEquipmentResourceTypeEnum;
    WHERE
        CorrectPredefinedType          :          (PredefinedType          <>
IfcConstructionEquipmentResourceTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcConstructionEquipmentResourceTypeEnum.USERDEFINED) AND
EXISTS (SELF\IfcTypeResource.ResourceType));
    END_ENTITY;

ENTITY IfcConstructionMaterialResource
    SUBTYPE OF (IfcConstructionResource);
    PredefinedType : OPTIONAL IfcConstructionMaterialResourceTypeEnum;
    WHERE
        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
        (PredefinedType <> IfcConstructionMaterialResourceTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcConstructionMaterialResourceTypeEnum.USERDEFINED) AND
EXISTS (SELF\IfcObject.ObjectType));
    END_ENTITY;

ENTITY IfcConstructionMaterialResourceType
    SUBTYPE OF (IfcConstructionResourceType);
    PredefinedType : IfcConstructionMaterialResourceTypeEnum;
    WHERE
        CorrectPredefinedType          :          (PredefinedType          <>
IfcConstructionMaterialResourceTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcConstructionMaterialResourceTypeEnum.USERDEFINED) AND
EXISTS (SELF\IfcTypeResource.ResourceType));
    END_ENTITY;

ENTITY IfcConstructionProductResource
    SUBTYPE OF (IfcConstructionResource);
    PredefinedType : OPTIONAL IfcConstructionProductResourceTypeEnum;
    WHERE
        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
        (PredefinedType <> IfcConstructionProductResourceTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcConstructionProductResourceTypeEnum.USERDEFINED) AND

```

---

```

EXISTS (SELF\IfcObject.ObjectType));
END_ENTITY;

ENTITY IfcConstructionProductResourceType
  SUBTYPE OF (IfcConstructionResourceType);
    PredefinedType : IfcConstructionProductResourceTypeEnum;
  WHERE
    CorrectPredefinedType          :          (PredefinedType          <>
IfcConstructionProductResourceTypeEnum.USERDEFINED) OR
((PredefinedType = IfcConstructionProductResourceTypeEnum.USERDEFINED) AND
EXISTS (SELF\IfcTypeResource.ResourceType));
END_ENTITY;

ENTITY IfcConstructionResource
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcConstructionEquipmentResource
    , IfcConstructionMaterialResource
    , IfcConstructionProductResource
    , IfcCrewResource
    , IfcLaborResource
    , IfcSubContractResource))
  SUBTYPE OF (IfcResource);
    Usage : OPTIONAL IfcResourceTime;
    BaseCosts : OPTIONAL LIST [1:?] OF IfcAppliedValue;
    BaseQuantity : OPTIONAL IfcPhysicalQuantity;
END_ENTITY;

ENTITY IfcConstructionResourceType
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcConstructionEquipmentResourceType
    , IfcConstructionMaterialResourceType
    , IfcConstructionProductResourceType
    , IfcCrewResourceType
    , IfcLaborResourceType
    , IfcSubContractResourceType))
  SUBTYPE OF (IfcTypeResource);
    BaseCosts : OPTIONAL LIST [1:?] OF IfcAppliedValue;
    BaseQuantity : OPTIONAL IfcPhysicalQuantity;
END_ENTITY;

ENTITY IfcContext
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcProject
    , IfcProjectLibrary))

```

---

```

SUBTYPE OF (IfcObjectDefinition);
  ObjectType : OPTIONAL IfcLabel;
  LongName : OPTIONAL IfcLabel;
  Phase : OPTIONAL IfcLabel;
  RepresentationContexts : OPTIONAL SET [1:?] OF IfcRepresentationContext;
  UnitsInContext : OPTIONAL IfcUnitAssignment;
INVERSE
  IsDefinedBy : SET [0:?] OF IfcRelDefinesByProperties FOR RelatedObjects;
  Declares : SET [0:?] OF IfcRelDeclares FOR RelatingContext;
END_ENTITY;

ENTITY IfcContextDependentUnit
  SUBTYPE OF (IfcNamedUnit);
    Name : IfcLabel;
  INVERSE
    HasExternalReference : SET [0:?] OF IfcExternalReferenceRelationship FOR
RelatedResourceObjects;
END_ENTITY;

ENTITY IfcControl
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcActionRequest
      , IfcCostItem
      , IfcCostSchedule
      , IfcPerformanceHistory
      , IfcPermit
      , IfcProjectOrder
      , IfcWorkCalendar
      , IfcWorkControl))
  SUBTYPE OF (IfcObject);
    Identification : OPTIONAL IfcIdentifier;
  INVERSE
    Controls : SET [0:?] OF IfcRelAssignsToControl FOR RelatingControl;
END_ENTITY;

ENTITY IfcController
  SUBTYPE OF (IfcDistributionControlElement);
    PredefinedType : OPTIONAL IfcControllerTypeEnum;
  WHERE
    CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
    (PredefinedType <> IfcControllerTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcControllerTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR

```

---

```
(' IFC4. IFCCONTROLLERTYPE' IN TYPEOF(SELF\IfcObject. IsTypedBy[1].RelatingType));  
END_ENTITY;
```

```
ENTITY IfcControllerType  
  SUBTYPE OF (IfcDistributionControlElementType);  
    PredefinedType : IfcControllerTypeEnum;  
  WHERE  
    CorrectPredefinedType : (PredefinedType <> IfcControllerTypeEnum.USERDEFINED)  
OR  
  ((PredefinedType = IfcControllerTypeEnum.USERDEFINED) AND  
  EXISTS(SELF\IfcElementType.ElementType));  
END_ENTITY;
```

```
ENTITY IfcConversionBasedUnit  
  SUPERTYPE OF (ONEOF  
    (IfcConversionBasedUnitWithOffset))  
  SUBTYPE OF (IfcNamedUnit);  
    Name : IfcLabel;  
    ConversionFactor : IfcMeasureWithUnit;  
  INVERSE  
    HasExternalReference : SET [0:?] OF IfcExternalReferenceRelationship FOR  
  RelatedResourceObjects;  
END_ENTITY;
```

```
ENTITY IfcConversionBasedUnitWithOffset  
  SUBTYPE OF (IfcConversionBasedUnit);  
    ConversionOffset : IfcReal;  
END_ENTITY;
```

```
ENTITY IfcCooledBeam  
  SUBTYPE OF (IfcEnergyConversionDevice);  
    PredefinedType : OPTIONAL IfcCooledBeamTypeEnum;  
  WHERE  
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR  
    (PredefinedType <> IfcCooledBeamTypeEnum.USERDEFINED) OR  
    ((PredefinedType = IfcCooledBeamTypeEnum.USERDEFINED) AND EXISTS  
    (SELF\IfcObject.ObjectType));  
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR  
    (' IFC4. IFCCOOLEDBEAMTYPE' IN  
    TYPEOF(SELF\IfcObject. IsTypedBy[1].RelatingType));  
END_ENTITY;
```

```
ENTITY IfcCooledBeamType  
  SUBTYPE OF (IfcEnergyConversionDeviceType);
```

---

```

    PredefinedType : IfcCooledBeamTypeEnum;
WHERE
    CorrectPredefinedType : (PredefinedType <> IfcCooledBeamTypeEnum.USERDEFINED)
OR
((PredefinedType = IfcCooledBeamTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcCoolingTower
SUBTYPE OF (IfcEnergyConversionDevice);
    PredefinedType : OPTIONAL IfcCoolingTowerTypeEnum;
WHERE
    CorrectPredefinedType : NOT (EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcCoolingTowerTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcCoolingTowerTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCCOOLINGTOWERTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcCoolingTowerType
SUBTYPE OF (IfcEnergyConversionDeviceType);
    PredefinedType : IfcCoolingTowerTypeEnum;
WHERE
    CorrectPredefinedType : (PredefinedType <>
    IfcCoolingTowerTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcCoolingTowerTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcCoordinateOperation
ABSTRACT SUPERTYPE OF (ONEOF
    (IfcMapConversion));
    SourceCRS : IfcCoordinateReferenceSystemSelect;
    TargetCRS : IfcCoordinateReferenceSystem;
END_ENTITY;

ENTITY IfcCoordinateReferenceSystem
ABSTRACT SUPERTYPE OF (ONEOF
    (IfcProjectedCRS));
    Name : IfcLabel;
    Description : OPTIONAL IfcText;
    GeodeticDatum : OPTIONAL IfcIdentifier;

```

---

```

    VerticalDatum : OPTIONAL IfcIdentifier;
INVERSE
    HasCoordinateOperation : SET [0:1] OF IfcCoordinateOperation FOR SourceCRS;
END_ENTITY;

ENTITY IfcCostItem
    SUBTYPE OF (IfcControl);
    PredefinedType : OPTIONAL IfcCostItemTypeEnum;
    CostValues : OPTIONAL LIST [1:?] OF IfcCostValue;
    CostQuantities : OPTIONAL LIST [1:?] OF IfcPhysicalQuantity;
END_ENTITY;

ENTITY IfcCostSchedule
    SUBTYPE OF (IfcControl);
    PredefinedType : OPTIONAL IfcCostScheduleTypeEnum;
    Status : OPTIONAL IfcLabel;
    SubmittedOn : OPTIONAL IfcDateTime;
    UpdateDate : OPTIONAL IfcDateTime;
END_ENTITY;

ENTITY IfcCostValue
    SUBTYPE OF (IfcAppliedValue);
END_ENTITY;

ENTITY IfcCovering
    SUBTYPE OF (IfcBuildingElement);
    PredefinedType : OPTIONAL IfcCoveringTypeEnum;
INVERSE
    CoversSpaces : SET [0:1] OF IfcRelCoversSpaces FOR RelatedCoverings;
    CoversElements : SET [0:1] OF IfcRelCoversBldgElements FOR RelatedCoverings;
WHERE
    CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
    (PredefinedType <> IfcCoveringTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcCoveringTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCCOVERINGTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcCoveringType
    SUBTYPE OF (IfcBuildingElementType);
    PredefinedType : IfcCoveringTypeEnum;
WHERE
    CorrectPredefinedType : (PredefinedType <> IfcCoveringTypeEnum.USERDEFINED)

```

---

```

OR
((PredefinedType          =          IfcCoveringTypeEnum. USERDEFINED)          AND
 EXISTS(SELF\IfcElementType. ElementType));
END_ENTITY;

```

```

ENTITY IfcCrewResource
  SUBTYPE OF (IfcConstructionResource);
    PredefinedType : OPTIONAL IfcCrewResourceTypeEnum;
  WHERE
    CorrectPredefinedType : NOT (EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcCrewResourceTypeEnum. USERDEFINED) OR
    ((PredefinedType = IfcCrewResourceTypeEnum. USERDEFINED) AND EXISTS
    (SELF\IfcObject. ObjectType));
END_ENTITY;

```

```

ENTITY IfcCrewResourceType
  SUBTYPE OF (IfcConstructionResourceType);
    PredefinedType : IfcCrewResourceTypeEnum;
  WHERE
    CorrectPredefinedType          :          (PredefinedType          <>
IfcCrewResourceTypeEnum. USERDEFINED) OR
    ((PredefinedType          =          IfcCrewResourceTypeEnum. USERDEFINED)          AND
 EXISTS(SELF\IfcTypeResource. ResourceType));
END_ENTITY;

```

```

ENTITY IfcCsgPrimitive3D
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcBlock
    , IfcRectangularPyramid
    , IfcRightCircularCone
    , IfcRightCircularCylinder
    , IfcSphere))
  SUBTYPE OF (IfcGeometricRepresentationItem);
    Position : IfcAxis2Placement3D;
  DERIVE
    Dim : IfcDimensionCount := 3;
END_ENTITY;

```

```

ENTITY IfcCsgSolid
  SUBTYPE OF (IfcSolidModel);
    TreeRootExpression : IfcCsgSelect;
END_ENTITY;

```

```

ENTITY IfcCurrencyRelationship

```

---

```

SUBTYPE OF (IfcResourceLevelRelationship);
    RelatingMonetaryUnit : IfcMonetaryUnit;
    RelatedMonetaryUnit : IfcMonetaryUnit;
    ExchangeRate : IfcPositiveRatioMeasure;
    RateDateTime : OPTIONAL IfcDateTime;
    RateSource : OPTIONAL IfcLibraryInformation;
END_ENTITY;

ENTITY IfcCurtainWall
    SUBTYPE OF (IfcBuildingElement);
        PredefinedType : OPTIONAL IfcCurtainWallTypeEnum;
    WHERE
        CorrectPredefinedType : NOT (EXISTS(PredefinedType)) OR
        (PredefinedType <> IfcCurtainWallTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcCurtainWallTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject.ObjectType));
        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
        ('IFC4.IFCCURTAINWALLTYPE' IN
        TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
    END_ENTITY;

ENTITY IfcCurtainWallType
    SUBTYPE OF (IfcBuildingElementType);
        PredefinedType : IfcCurtainWallTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <> IfcCurtainWallTypeEnum.USERDEFINED)
    OR
        ((PredefinedType = IfcCurtainWallTypeEnum.USERDEFINED) AND
        EXISTS(SELF\IfcElementType.ElementType));
    END_ENTITY;

ENTITY IfcCurve
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcBoundedCurve
        , IfcConic
        , IfcLine
        , IfcOffsetCurve2D
        , IfcOffsetCurve3D
        , IfcPcurve
        , IfcSurfaceCurve))
    SUBTYPE OF (IfcGeometricRepresentationItem);
    DERIVE
        Dim : IfcDimensionCount := IfcCurveDim(SELF);
    END_ENTITY;

```



---

```

ENTITY IfcCurveBoundedPlane
  SUBTYPE OF (IfcBoundedSurface);
    BasisSurface : IfcPlane;
    OuterBoundary : IfcCurve;
    InnerBoundaries : SET [0:?] OF IfcCurve;
END_ENTITY;

ENTITY IfcCurveBoundedSurface
  SUBTYPE OF (IfcBoundedSurface);
    BasisSurface : IfcSurface;
    Boundaries : SET [1:?] OF IfcBoundaryCurve;
    ImplicitOuter : IfcBoolean;
END_ENTITY;

ENTITY IfcCurveStyle
  SUBTYPE OF (IfcPresentationStyle);
    CurveFont : OPTIONAL IfcCurveFontOrScaledCurveFontSelect;
    CurveWidth : OPTIONAL IfcSizeSelect;
    CurveColour : OPTIONAL IfcColour;
    ModelOrDraughting : OPTIONAL IfcBoolean;
  WHERE
    MeasureOfWidth : (NOT(EXISTS(CurveWidth))) OR
    ('IFC4.IFCPOSITIVELENGTHMEASURE' IN TYPEOF(CurveWidth)) OR
    (('IFC4.IFCDESCRIPTIVEMEASURE' IN TYPEOF(CurveWidth)) AND
    (CurveWidth = 'by layer'));
    IdentifiableCurveStyle : EXISTS(CurveFont) OR EXISTS(CurveWidth) OR
    EXISTS(CurveColour);
END_ENTITY;

ENTITY IfcCurveStyleFont
  SUBTYPE OF (IfcPresentationItem);
    Name : OPTIONAL IfcLabel;
    PatternList : LIST [1:?] OF IfcCurveStyleFontPattern;
END_ENTITY;

ENTITY IfcCurveStyleFontAndScaling
  SUBTYPE OF (IfcPresentationItem);
    Name : OPTIONAL IfcLabel;
    CurveFont : IfcCurveStyleFontSelect;
    CurveFontScaling : IfcPositiveRatioMeasure;
END_ENTITY;

ENTITY IfcCurveStyleFontPattern

```

---

```

SUBTYPE OF (IfcPresentationItem);
    VisibleSegmentLength : IfcLengthMeasure;
    InvisibleSegmentLength : IfcPositiveLengthMeasure;
WHERE
    VisibleLengthGreaterEqualZero : VisibleSegmentLength >= 0.;
END_ENTITY;

ENTITY IfcCylindricalSurface
    SUBTYPE OF (IfcElementarySurface);
        Radius : IfcPositiveLengthMeasure;
END_ENTITY;

ENTITY IfcDamper
    SUBTYPE OF (IfcFlowController);
        PredefinedType : OPTIONAL IfcDamperTypeEnum;
WHERE
    CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
    (PredefinedType <> IfcDamperTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcDamperTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCDAMPERTYPE' IN TYPEOF (SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcDamperType
    SUBTYPE OF (IfcFlowControllerType);
        PredefinedType : IfcDamperTypeEnum;
WHERE
    CorrectPredefinedType : (PredefinedType <> IfcDamperTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcDamperTypeEnum.USERDEFINED) AND
    EXISTS (SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcDerivedProfileDef
    SUPERTYPE OF (ONEOF
        (IfcMirroredProfileDef))
    SUBTYPE OF (IfcProfileDef);
        ParentProfile : IfcProfileDef;
        Operator : IfcCartesianTransformationOperator2D;
        Label : OPTIONAL IfcLabel;
WHERE
    InvariantProfileType : SELF\IfcProfileDef.ProfileType =
    ParentProfile.ProfileType;
END_ENTITY;

```

---

```

ENTITY IfcDerivedUnit;
    Elements : SET [1:?] OF IfcDerivedUnitElement;
    UnitType : IfcDerivedUnitEnum;
    UserDefinedType : OPTIONAL IfcLabel;
    DERIVE
        Dimensions : IfcDimensionalExponents :=
IfcDerivedDimensionalExponents(Elements);
    WHERE
        WR1 : (SIZEOF (Elements) > 1) OR ((SIZEOF (Elements) = 1) AND
(Elements[1].Exponent <> 1));
        WR2 : (UnitType <> IfcDerivedUnitEnum.USERDEFINED) OR
((UnitType = IfcDerivedUnitEnum.USERDEFINED) AND
(EXISTS(SELF.UserDefinedType)));
END_ENTITY;

ENTITY IfcDerivedUnitElement;
    Unit : IfcNamedUnit;
    Exponent : INTEGER;
END_ENTITY;

ENTITY IfcDimensionalExponents;
    LengthExponent : INTEGER;
    MassExponent : INTEGER;
    TimeExponent : INTEGER;
    ElectricCurrentExponent : INTEGER;
    ThermodynamicTemperatureExponent : INTEGER;
    AmountOfSubstanceExponent : INTEGER;
    LuminousIntensityExponent : INTEGER;
END_ENTITY;

ENTITY IfcDirection
    SUBTYPE OF (IfcGeometricRepresentationItem);
    DirectionRatios : LIST [2:3] OF IfcReal;
    DERIVE
        Dim : IfcDimensionCount := HIINDEX(DirectionRatios);
    WHERE
        MagnitudeGreaterZero : SIZEOF(QUERY(Tmp < * DirectionRatios | Tmp <> 0.0)) > 0;
END_ENTITY;

ENTITY IfcDiscreteAccessory
    SUBTYPE OF (IfcElementComponent);
    PredefinedType : OPTIONAL IfcDiscreteAccessoryTypeEnum;
    WHERE

```

---

```

        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
(PredefinedType <> IfcDiscreteAccessoryTypeEnum.USERDEFINED) OR
((PredefinedType = IfcDiscreteAccessoryTypeEnum.USERDEFINED) AND EXISTS
(SELF\IfcObject.ObjectType));
        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
(' IFC4. IFCDISCRETEACCESSORYTYPE' IN
TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcDiscreteAccessoryType
SUBTYPE OF (IfcElementComponentType);
    PredefinedType : IfcDiscreteAccessoryTypeEnum;
WHERE
    CorrectPredefinedType : (PredefinedType <>
IfcDiscreteAccessoryTypeEnum.USERDEFINED) OR
((PredefinedType = IfcDiscreteAccessoryTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcDistributionChamberElement
SUBTYPE OF (IfcDistributionFlowElement);
    PredefinedType : OPTIONAL IfcDistributionChamberElementTypeEnum;
WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
(PredefinedType <> IfcDistributionChamberElementTypeEnum.USERDEFINED) OR
((PredefinedType = IfcDistributionChamberElementTypeEnum.USERDEFINED) AND EXISTS
(SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
(' IFC4. IFCDISTRIBUTIONCHAMBERELEMENTTYPE' IN
TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcDistributionChamberElementType
SUBTYPE OF (IfcDistributionFlowElementType);
    PredefinedType : IfcDistributionChamberElementTypeEnum;
WHERE
    CorrectPredefinedType : (PredefinedType <>
IfcDistributionChamberElementTypeEnum.USERDEFINED) OR
((PredefinedType = IfcDistributionChamberElementTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcDistributionCircuit
SUBTYPE OF (IfcDistributionSystem);

```

---

```

END_ENTITY;

ENTITY IfcDistributionControlElement
  SUPERTYPE OF (ONEOF
    (IfcActuator
    , IfcAlarm
    , IfcController
    , IfcFlowInstrument
    , IfcProtectiveDeviceTrippingUnit
    , IfcSensor
    , IfcUnitaryControlElement))
  SUBTYPE OF (IfcDistributionElement);
  INVERSE
    AssignedToFlowElement : SET [0:1] OF IfcRelFlowControlElements FOR
RelatedControlElements;
END_ENTITY;

```

```

ENTITY IfcDistributionControlElementType
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcActuatorType
    , IfcAlarmType
    , IfcControllerType
    , IfcFlowInstrumentType
    , IfcProtectiveDeviceTrippingUnitType
    , IfcSensorType
    , IfcUnitaryControlElementType))
  SUBTYPE OF (IfcDistributionElementType);
END_ENTITY;

```

```

ENTITY IfcDistributionElement
  SUPERTYPE OF (ONEOF
    (IfcDistributionControlElement
    , IfcDistributionFlowElement))
  SUBTYPE OF (IfcElement);
  INVERSE
    HasPorts : SET [0:?] OF IfcRelConnectsPortToElement FOR RelatedElement;
END_ENTITY;

```

```

ENTITY IfcDistributionElementType
  SUPERTYPE OF (ONEOF
    (IfcDistributionControlElementType
    , IfcDistributionFlowElementType))
  SUBTYPE OF (IfcElementType);
END_ENTITY;

```

---

```

ENTITY IfcDistributionFlowElement
  SUPERTYPE OF (ONEOF
    (IfcDistributionChamberElement
      , IfcEnergyConversionDevice
      , IfcFlowController
      , IfcFlowFitting
      , IfcFlowMovingDevice
      , IfcFlowSegment
      , IfcFlowStorageDevice
      , IfcFlowTerminal
      , IfcFlowTreatmentDevice))
  SUBTYPE OF (IfcDistributionElement);
  INVERSE
    HasControlElements : SET [0:1] OF IfcRelFlowControlElements FOR
    RelatingFlowElement;
END_ENTITY;

```

```

ENTITY IfcDistributionFlowElementType
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcDistributionChamberElementType
      , IfcEnergyConversionDeviceType
      , IfcFlowControllerType
      , IfcFlowFittingType
      , IfcFlowMovingDeviceType
      , IfcFlowSegmentType
      , IfcFlowStorageDeviceType
      , IfcFlowTerminalType
      , IfcFlowTreatmentDeviceType))
  SUBTYPE OF (IfcDistributionElementType);
END_ENTITY;

```

```

ENTITY IfcDistributionPort
  SUBTYPE OF (IfcPort);
  FlowDirection : OPTIONAL IfcFlowDirectionEnum;
  PredefinedType : OPTIONAL IfcDistributionPortTypeEnum;
  SystemType : OPTIONAL IfcDistributionSystemEnum;
END_ENTITY;

```

```

ENTITY IfcDistributionSystem
  SUPERTYPE OF (ONEOF
    (IfcDistributionCircuit))
  SUBTYPE OF (IfcSystem);
  LongName : OPTIONAL IfcLabel;

```

---

```
    PredefinedType : OPTIONAL IfcDistributionSystemEnum;
END_ENTITY;
```

```
ENTITY IfcDocumentInformation
  SUBTYPE OF (IfcExternalInformation);
    Identification : IfcIdentifier;
    Name : IfcLabel;
    Description : OPTIONAL IfcText;
    Location : OPTIONAL IfcURIReference;
    Purpose : OPTIONAL IfcText;
    IntendedUse : OPTIONAL IfcText;
    Scope : OPTIONAL IfcText;
    Revision : OPTIONAL IfcLabel;
    DocumentOwner : OPTIONAL IfcActorSelect;
    Editors : OPTIONAL SET [1:?] OF IfcActorSelect;
    CreationTime : OPTIONAL IfcDateTime;
    LastRevisionTime : OPTIONAL IfcDateTime;
    ElectronicFormat : OPTIONAL IfcIdentifier;
    ValidFrom : OPTIONAL IfcDate;
    ValidUntil : OPTIONAL IfcDate;
    Confidentiality : OPTIONAL IfcDocumentConfidentialityEnum;
    Status : OPTIONAL IfcDocumentStatusEnum;
  INVERSE
    DocumentInfoForObjects : SET [0:?] OF IfcRelAssociatesDocument FOR
    RelatingDocument;
    HasDocumentReferences : SET [0:?] OF IfcDocumentReference FOR
    ReferencedDocument;
    IsPointedTo : SET [0:?] OF IfcDocumentInformationRelationship FOR
    RelatedDocuments;
    IsPointer : SET [0:1] OF IfcDocumentInformationRelationship FOR
    RelatingDocument;
END_ENTITY;
```

```
ENTITY IfcDocumentInformationRelationship
  SUBTYPE OF (IfcResourceLevelRelationship);
    RelatingDocument : IfcDocumentInformation;
    RelatedDocuments : SET [1:?] OF IfcDocumentInformation;
    RelationshipType : OPTIONAL IfcLabel;
END_ENTITY;
```

```
ENTITY IfcDocumentReference
  SUBTYPE OF (IfcExternalReference);
    Description : OPTIONAL IfcText;
    ReferencedDocument : OPTIONAL IfcDocumentInformation;
```

---

```

INVERSE
    DocumentRefForObjects : SET [0:?] OF IfcRelAssociatesDocument FOR
RelatingDocument;
WHERE
    WR1 : EXISTS(Name) XOR EXISTS(ReferencedDocument);
END_ENTITY;

ENTITY IfcDoor
    SUPERTYPE OF (ONEOF
        (IfcDoorStandardCase))
    SUBTYPE OF (IfcBuildingElement);
    OverallHeight : OPTIONAL IfcPositiveLengthMeasure;
    OverallWidth : OPTIONAL IfcPositiveLengthMeasure;
    PredefinedType : OPTIONAL IfcDoorTypeEnum;
    OperationType : OPTIONAL IfcDoorTypeOperationEnum;
    UserDefinedOperationType : OPTIONAL IfcLabel;
WHERE
    CorrectStyleAssigned : (SIZEOF(IsTypedBy) = 0)
OR ('IFC4.IFCDOORTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcDoorLiningProperties
    SUBTYPE OF (IfcPreDefinedPropertySet);
    LiningDepth : OPTIONAL IfcPositiveLengthMeasure;
    LiningThickness : OPTIONAL IfcNonNegativeLengthMeasure;
    ThresholdDepth : OPTIONAL IfcPositiveLengthMeasure;
    ThresholdThickness : OPTIONAL IfcNonNegativeLengthMeasure;
    TransomThickness : OPTIONAL IfcNonNegativeLengthMeasure;
    TransomOffset : OPTIONAL IfcLengthMeasure;
    LiningOffset : OPTIONAL IfcLengthMeasure;
    ThresholdOffset : OPTIONAL IfcLengthMeasure;
    CasingThickness : OPTIONAL IfcPositiveLengthMeasure;
    CasingDepth : OPTIONAL IfcPositiveLengthMeasure;
    ShapeAspectStyle : OPTIONAL IfcShapeAspect;
    LiningToPanelOffsetX : OPTIONAL IfcLengthMeasure;
    LiningToPanelOffsetY : OPTIONAL IfcLengthMeasure;
WHERE
    WR31 : NOT(EXISTS(LiningDepth) AND NOT(EXISTS(LiningThickness)));
    WR32 : NOT(EXISTS(ThresholdDepth) AND NOT(EXISTS(ThresholdThickness)));
    WR33 : (EXISTS(TransomOffset) AND EXISTS(TransomThickness)) XOR
(NOT(EXISTS(TransomOffset)) AND NOT(EXISTS(TransomThickness)));
    WR34 : (EXISTS(CasingDepth) AND EXISTS(CasingThickness)) XOR
(NOT(EXISTS(CasingDepth)) AND NOT(EXISTS(CasingThickness)));
    WR35 : (EXISTS(SELF\IfcPropertySetDefinition.DefinesType[1]))

```



---

```

AND
(
  (' IFC4. IFCDOORSTYLE' IN TYPEOF(SELF\IfcPropertySetDefinition.DefinesType[1]))
  OR
  (' IFC4. IFCDOORSTYLE' IN TYPEOF(SELF\IfcPropertySetDefinition.DefinesType[1]))
);
END_ENTITY;

```

```

ENTITY IfcDoorPanelProperties
  SUBTYPE OF (IfcPreDefinedPropertySet);
    PanelDepth : OPTIONAL IfcPositiveLengthMeasure;
    PanelOperation : IfcDoorPanelOperationEnum;
    PanelWidth : OPTIONAL IfcNormalisedRatioMeasure;
    PanelPosition : IfcDoorPanelPositionEnum;
    ShapeAspectStyle : OPTIONAL IfcShapeAspect;
  WHERE
    ApplicableToType : (EXISTS(SELF\IfcPropertySetDefinition.DefinesType[1]))
AND
(
  (' IFC4. IFCDOORSTYLE' IN TYPEOF(SELF\IfcPropertySetDefinition.DefinesType[1]))
  OR
  (' IFC4. IFCDOORSTYLE' IN TYPEOF(SELF\IfcPropertySetDefinition.DefinesType[1]))
);
END_ENTITY;

```

```

ENTITY IfcDoorStandardCase
  SUBTYPE OF (IfcDoor);
END_ENTITY;

```

```

ENTITY IfcDoorStyle
  SUBTYPE OF (IfcTypeProduct);
    OperationType : IfcDoorStyleOperationEnum;
    ConstructionType : IfcDoorStyleConstructionEnum;
    ParameterTakesPrecedence : IfcBoolean;
    Sizeable : IfcBoolean;
END_ENTITY;

```

```

ENTITY IfcDoorType
  SUBTYPE OF (IfcBuildingElementType);
    PredefinedType : IfcDoorTypeEnum;
    OperationType : IfcDoorTypeOperationEnum;
    ParameterTakesPrecedence : OPTIONAL IfcBoolean;
    UserDefinedOperationType : OPTIONAL IfcLabel;
  WHERE

```

---

```

        CorrectPredefinedType : (PredefinedType <> IfcDoorTypeEnum.USERDEFINED) OR
((PredefinedType = IfcDoorTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcDraughtingPreDefinedColour
SUBTYPE OF (IfcPreDefinedColour);
WHERE
    PreDefinedColourNames : SELF\IfcPreDefinedItem.Name IN
['black','red','green','blue','yellow',
'magenta','cyan','white','by layer'];
END_ENTITY;

```

```

ENTITY IfcDraughtingPreDefinedCurveFont
SUBTYPE OF (IfcPreDefinedCurveFont);
WHERE
    PreDefinedCurveFontNames : SELF\IfcPredefinedItem.Name IN
['continuous',
'chain',
'chain double dash',
'dashed',
'dotted',
'by layer'];
END_ENTITY;

```

```

ENTITY IfcDuctFitting
SUBTYPE OF (IfcFlowFitting);
    PredefinedType : OPTIONAL IfcDuctFittingTypeEnum;
WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
(PredefinedType <> IfcDuctFittingTypeEnum.USERDEFINED) OR
((PredefinedType = IfcDuctFittingTypeEnum.USERDEFINED) AND EXISTS
(SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
('IFC4.IFCDUCTFITTINGTYPE' IN
TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcDuctFittingType
SUBTYPE OF (IfcFlowFittingType);
    PredefinedType : IfcDuctFittingTypeEnum;
WHERE
    CorrectPredefinedType : (PredefinedType <> IfcDuctFittingTypeEnum.USERDEFINED)
OR

```

---

```

((PredefinedType = IfcDuctFittingTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcDuctSegment
  SUBTYPE OF (IfcFlowSegment);
  PredefinedType : OPTIONAL IfcDuctSegmentTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcDuctSegmentTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcDuctSegmentTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    (' IFC4. IFCDUCTSEGMENTTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcDuctSegmentType
  SUBTYPE OF (IfcFlowSegmentType);
  PredefinedType : IfcDuctSegmentTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcDuctSegmentTypeEnum.USERDEFINED)
OR
    ((PredefinedType = IfcDuctSegmentTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcDuctSilencer
  SUBTYPE OF (IfcFlowTreatmentDevice);
  PredefinedType : OPTIONAL IfcDuctSilencerTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcDuctSilencerTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcDuctSilencerTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    (' IFC4. IFCDUCTSILENCERTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcDuctSilencerType
  SUBTYPE OF (IfcFlowTreatmentDeviceType);
  PredefinedType : IfcDuctSilencerTypeEnum;
  WHERE

```

---

```

        CorrectPredefinedType          :          (PredefinedType          <>
IfcDuctSilencerTypeEnum.USERDEFINED) OR
((PredefinedType          =          IfcDuctSilencerTypeEnum.USERDEFINED)          AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcEdge
    SUPERTYPE OF (ONEOF
        (IfcEdgeCurve
        , IfcOrientedEdge
        , IfcSubedge))
    SUBTYPE OF (IfcTopologicalRepresentationItem);
        EdgeStart : IfcVertex;
        EdgeEnd : IfcVertex;
END_ENTITY;

ENTITY IfcEdgeCurve
    SUBTYPE OF (IfcEdge);
        EdgeGeometry : IfcCurve;
        SameSense : IfcBoolean;
END_ENTITY;

ENTITY IfcEdgeLoop
    SUBTYPE OF (IfcLoop);
        EdgeList : LIST [1:?] OF IfcOrientedEdge;
    DERIVE
        Ne : IfcInteger := SIZEOF(EdgeList);
    WHERE
        IsClosed : (EdgeList[1].EdgeStart) :=: (EdgeList[Ne].EdgeEnd);
        IsContinuous : IfcLoopHeadToTail(SELF);
END_ENTITY;

ENTITY IfcElectricAppliance
    SUBTYPE OF (IfcFlowTerminal);
        PredefinedType : OPTIONAL IfcElectricApplianceTypeEnum;
    WHERE
        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
        (PredefinedType <> IfcElectricApplianceTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcElectricApplianceTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject.ObjectType));
        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
        ('IFC4.IFCELECTRICAPPLIANCETYPE' IN
        TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

---

```

ENTITY IfcElectricApplianceType
  SUBTYPE OF (IfcFlowTerminalType);
    PredefinedType : IfcElectricApplianceTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <>
IfcElectricApplianceTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcElectricApplianceTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcElectricDistributionBoard
  SUBTYPE OF (IfcFlowController);
    PredefinedType : OPTIONAL IfcElectricDistributionBoardTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcElectricDistributionBoardTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcElectricDistributionBoardTypeEnum.USERDEFINED) AND EXISTS
(SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    (' IFC4. IFCELECTRICDISTRIBUTIONBOARDTYPE' IN
TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcElectricDistributionBoardType
  SUBTYPE OF (IfcFlowControllerType);
    PredefinedType : IfcElectricDistributionBoardTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <>
IfcElectricDistributionBoardTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcElectricDistributionBoardTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcElectricFlowStorageDevice
  SUBTYPE OF (IfcFlowStorageDevice);
    PredefinedType : OPTIONAL IfcElectricFlowStorageDeviceTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcElectricFlowStorageDeviceTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcElectricFlowStorageDeviceTypeEnum.USERDEFINED) AND EXISTS
(SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    (' IFC4. IFCELECTRICFLOWSTORAGEDEVICETYPE' IN

```

---

```

TYPEOF(SELf\IfcObject. IsTypedBy[1]. RelatingType));
END_ENTITY;

ENTITY IfcElectricFlowStorageDeviceType
  SUBTYPE OF (IfcFlowStorageDeviceType);
  PredefinedType : IfcElectricFlowStorageDeviceTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <>
IfcElectricFlowStorageDeviceTypeEnum. USERDEFINED) OR
((PredefinedType = IfcElectricFlowStorageDeviceTypeEnum. USERDEFINED) AND
EXISTS(SELf\IfcElementType. ElementType));
END_ENTITY;

ENTITY IfcElectricGenerator
  SUBTYPE OF (IfcEnergyConversionDevice);
  PredefinedType : OPTIONAL IfcElectricGeneratorTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
(PredefinedType <> IfcElectricGeneratorTypeEnum. USERDEFINED) OR
((PredefinedType = IfcElectricGeneratorTypeEnum. USERDEFINED) AND EXISTS
(SELf\IfcObject. ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
(' IFC4. IFCELECTRICGENERATORATYPE' IN
TYPEOF(SELf\IfcObject. IsTypedBy[1]. RelatingType));
END_ENTITY;

ENTITY IfcElectricGeneratorType
  SUBTYPE OF (IfcEnergyConversionDeviceType);
  PredefinedType : IfcElectricGeneratorTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <>
IfcElectricGeneratorTypeEnum. USERDEFINED) OR
((PredefinedType = IfcElectricGeneratorTypeEnum. USERDEFINED) AND
EXISTS(SELf\IfcElementType. ElementType));
END_ENTITY;

ENTITY IfcElectricMotor
  SUBTYPE OF (IfcEnergyConversionDevice);
  PredefinedType : OPTIONAL IfcElectricMotorTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
(PredefinedType <> IfcElectricMotorTypeEnum. USERDEFINED) OR
((PredefinedType = IfcElectricMotorTypeEnum. USERDEFINED) AND EXISTS
(SELf\IfcObject. ObjectType));

```

---

```

        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
(' IFC4. IFCELECTRICMOTORTYPE'                                     IN
TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;
```

```

ENTITY IfcElectricMotorType
  SUBTYPE OF (IfcEnergyConversionDeviceType);
    PredefinedType : IfcElectricMotorTypeEnum;
  WHERE
    CorrectPredefinedType      :      (PredefinedType      <>
IfcElectricMotorTypeEnum.USERDEFINED) OR
((PredefinedType      =      IfcElectricMotorTypeEnum.USERDEFINED)      AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;
```

```

ENTITY IfcElectricTimeControl
  SUBTYPE OF (IfcFlowController);
    PredefinedType : OPTIONAL IfcElectricTimeControlTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
(PredefinedType <> IfcElectricTimeControlTypeEnum.USERDEFINED) OR
((PredefinedType = IfcElectricTimeControlTypeEnum.USERDEFINED) AND EXISTS
(SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
(' IFC4. IFCELECTRICTIMECONTROLTYPE'                                     IN
TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;
```

```

ENTITY IfcElectricTimeControlType
  SUBTYPE OF (IfcFlowControllerType);
    PredefinedType : IfcElectricTimeControlTypeEnum;
  WHERE
    CorrectPredefinedType      :      (PredefinedType      <>
IfcElectricTimeControlTypeEnum.USERDEFINED) OR
((PredefinedType      =      IfcElectricTimeControlTypeEnum.USERDEFINED)      AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;
```

```

ENTITY IfcElement
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcBuildingElement
    , IfcCivilElement
    , IfcDistributionElement
    , IfcElementAssembly
```

---

```

    , IfcElementComponent
    , IfcFeatureElement
    , IfcFurnishingElement
    , IfcGeographicElement
    , IfcTransportElement
    , IfcVirtualElement))
SUBTYPE OF (IfcProduct);
    Tag : OPTIONAL IfcIdentifier;
INVERSE
    FillsVoids : SET [0:1] OF IfcRelFillsElement FOR RelatedBuildingElement;
    ConnectedTo : SET [0:?] OF IfcRelConnectsElements FOR RelatingElement;
    IsInterferedByElements : SET [0:?] OF IfcRelInterferesElements FOR
RelatedElement;
    InterferesElements : SET [0:?] OF IfcRelInterferesElements FOR
RelatingElement;
    HasProjections : SET [0:?] OF IfcRelProjectsElement FOR RelatingElement;
    ReferencedInStructures : SET [0:?] OF IfcRelReferencedInSpatialStructure FOR
RelatedElements;
    HasOpenings : SET [0:?] OF IfcRelVoidsElement FOR RelatingBuildingElement;
    IsConnectionRealization : SET [0:?] OF IfcRelConnectsWithRealizingElements FOR
RealizingElements;
    ProvidesBoundaries : SET [0:?] OF IfcRelSpaceBoundary FOR
RelatedBuildingElement;
    ConnectedFrom : SET [0:?] OF IfcRelConnectsElements FOR RelatedElement;
    ContainedInStructure : SET [0:1] OF IfcRelContainedInSpatialStructure FOR
RelatedElements;
    HasCoverings : SET [0:?] OF IfcRelCoversBldgElements FOR
RelatingBuildingElement;
END_ENTITY;

ENTITY IfcElementAssembly
SUBTYPE OF (IfcElement);
    AssemblyPlace : OPTIONAL IfcAssemblyPlaceEnum;
    PredefinedType : OPTIONAL IfcElementAssemblyTypeEnum;
WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcElementAssemblyTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcElementAssemblyTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4. IFCELEMENTASSEMBLYTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```



---

```

ENTITY IfcElementAssemblyType
  SUBTYPE OF (IfcElementType);
    PredefinedType : IfcElementAssemblyTypeEnum;
  WHERE
    CorrectPredefinedType          :          (PredefinedType          <>
IfcElementAssemblyTypeEnum.USERDEFINED) OR
((PredefinedType          =          IfcElementAssemblyTypeEnum.USERDEFINED)          AND
EXISTS(SELf\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcElementComponent
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcBuildingElementPart
    , IfcDiscreteAccessory
    , IfcFastener
    , IfcMechanicalFastener
    , IfcReinforcingElement
    , IfcVibrationIsolator))
  SUBTYPE OF (IfcElement);
END_ENTITY;

```

```

ENTITY IfcElementComponentType
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcBuildingElementPartType
    , IfcDiscreteAccessoryType
    , IfcFastenerType
    , IfcMechanicalFastenerType
    , IfcReinforcingElementType
    , IfcVibrationIsolatorType))
  SUBTYPE OF (IfcElementType);
END_ENTITY;

```

```

ENTITY IfcElementQuantity
  SUBTYPE OF (IfcQuantitySet);
    MethodOfMeasurement : OPTIONAL IfcLabel;
    Quantities : SET [1:?] OF IfcPhysicalQuantity;
  WHERE
    UniqueQuantityNames : IfcUniqueQuantityNames(Quantities);
END_ENTITY;

```

```

ENTITY IfcElementType
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcBuildingElementType
    , IfcCivilElementType

```

---

```

        , IfcDistributionElementType
        , IfcElementAssemblyType
        , IfcElementComponentType
        , IfcFurnishingElementType
        , IfcGeographicElementType
        , IfcTransportElementType))
SUBTYPE OF (IfcTypeProduct);
    ElementType : OPTIONAL IfcLabel;
END_ENTITY;

ENTITY IfcElementarySurface
ABSTRACT SUPERTYPE OF (ONEOF
    (IfcCylindricalSurface
    , IfcPlane
    , IfcSphericalSurface
    , IfcToroidalSurface))
SUBTYPE OF (IfcSurface);
    Position : IfcAxis2Placement3D;
END_ENTITY;

ENTITY IfcEllipse
SUBTYPE OF (IfcConic);
    SemiAxis1 : IfcPositiveLengthMeasure;
    SemiAxis2 : IfcPositiveLengthMeasure;
END_ENTITY;

ENTITY IfcEllipseProfileDef
SUBTYPE OF (IfcParameterizedProfileDef);
    SemiAxis1 : IfcPositiveLengthMeasure;
    SemiAxis2 : IfcPositiveLengthMeasure;
END_ENTITY;

ENTITY IfcEnergyConversionDevice
SUPERTYPE OF (ONEOF
    (IfcAirToAirHeatRecovery
    , IfcBoiler
    , IfcBurner
    , IfcChiller
    , IfcCoil
    , IfcCondenser
    , IfcCooledBeam
    , IfcCoolingTower
    , IfcElectricGenerator
    , IfcElectricMotor

```

---

```

    , IfcEngine
    , IfcEvaporativeCooler
    , IfcEvaporator
    , IfcHeatExchanger
    , IfcHumidifier
    , IfcMotorConnection
    , IfcSolarDevice
    , IfcTransformer
    , IfcTubeBundle
    , IfcUnitaryEquipment))
SUBTYPE OF (IfcDistributionFlowElement);
END_ENTITY;

```

```

ENTITY IfcEnergyConversionDeviceType
ABSTRACT SUPERTYPE OF (ONEOF
    (IfcAirToAirHeatRecoveryType
    , IfcBoilerType
    , IfcBurnerType
    , IfcChillerType
    , IfcCoilType
    , IfcCondenserType
    , IfcCooledBeamType
    , IfcCoolingTowerType
    , IfcElectricGeneratorType
    , IfcElectricMotorType
    , IfcEngineType
    , IfcEvaporativeCoolerType
    , IfcEvaporatorType
    , IfcHeatExchangerType
    , IfcHumidifierType
    , IfcMotorConnectionType
    , IfcSolarDeviceType
    , IfcTransformerType
    , IfcTubeBundleType
    , IfcUnitaryEquipmentType))
SUBTYPE OF (IfcDistributionFlowElementType);
END_ENTITY;

```

```

ENTITY IfcEngine
SUBTYPE OF (IfcEnergyConversionDevice);
    PredefinedType : OPTIONAL IfcEngineTypeEnum;
WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcEngineTypeEnum.USERDEFINED) OR

```

---

```

    ((PredefinedType = IfcEngineTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCENGINE TYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcEngineType
    SUBTYPE OF (IfcEnergyConversionDeviceType);
    PredefinedType : IfcEngineTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <> IfcEngineTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcEngineTypeEnum.USERDEFINED) AND
        EXISTS (SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcEvaporativeCooler
    SUBTYPE OF (IfcEnergyConversionDevice);
    PredefinedType : OPTIONAL IfcEvaporativeCoolerTypeEnum;
    WHERE
        CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
        (PredefinedType <> IfcEvaporativeCoolerTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcEvaporativeCoolerTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject.ObjectType));
        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
        ('IFC4.IFCEVAPORATIVECOOLERTYPE' IN
        TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcEvaporativeCoolerType
    SUBTYPE OF (IfcEnergyConversionDeviceType);
    PredefinedType : IfcEvaporativeCoolerTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <>
        IfcEvaporativeCoolerTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcEvaporativeCoolerTypeEnum.USERDEFINED) AND
        EXISTS (SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcEvaporator
    SUBTYPE OF (IfcEnergyConversionDevice);
    PredefinedType : OPTIONAL IfcEvaporatorTypeEnum;
    WHERE
        CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
        (PredefinedType <> IfcEvaporatorTypeEnum.USERDEFINED) OR

```

---

```

    ((PredefinedType      =      IfcEvaporatorTypeEnum. USERDEFINED)      AND      EXISTS
    (SELF\IfcObject. ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    (' IFC4. IFCEVAPORATORTYPE'                                     IN
    TYPEOF(SELF\IfcObject. IsTypedBy[1]. RelatingType));
END_ENTITY;

```

```

ENTITY IfcEvaporatorType
  SUBTYPE OF (IfcEnergyConversionDeviceType);
  PredefinedType : IfcEvaporatorTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcEvaporatorTypeEnum. USERDEFINED)
OR
    ((PredefinedType      =      IfcEvaporatorTypeEnum. USERDEFINED)      AND
    EXISTS(SELF\IfcElementType. ElementType));
END_ENTITY;

```

```

ENTITY IfcEvent
  SUBTYPE OF (IfcProcess);
  PredefinedType : OPTIONAL IfcEventTypeEnum;
  EventTriggerType : OPTIONAL IfcEventTriggerTypeEnum;
  UserDefinedEventTriggerType : OPTIONAL IfcLabel;
  EventOccurenceTime : OPTIONAL IfcEventTime;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR (PredefinedType <>
    IfcEventTypeEnum. USERDEFINED) OR ((PredefinedType = IfcEventTypeEnum. USERDEFINED)
    AND EXISTS(SELF\IfcObject. ObjectType));
    CorrectTypeAssigned : NOT(EXISTS(EventTriggerType)) OR (EventTriggerType <>
    IfcEventTriggerTypeEnum. USERDEFINED)      OR      ((EventTriggerType      =
    IfcEventTriggerTypeEnum. USERDEFINED) AND EXISTS(UserDefinedEventTriggerType));
END_ENTITY;

```

```

ENTITY IfcEventTime
  SUBTYPE OF (IfcSchedulingTime);
  ActualDate : OPTIONAL IfcDateTime;
  EarlyDate : OPTIONAL IfcDateTime;
  LateDate : OPTIONAL IfcDateTime;
  ScheduleDate : OPTIONAL IfcDateTime;
END_ENTITY;

```

```

ENTITY IfcEventType
  SUBTYPE OF (IfcTypeProcess);
  PredefinedType : IfcEventTypeEnum;
  EventTriggerType : IfcEventTriggerTypeEnum;

```

---

```

        UserDefinedEventTriggerType : OPTIONAL IfcLabel;
WHERE
    CorrectPredefinedType : (PredefinedType <> IfcEventTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcEventTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcTypeProcess.ProcessType));
    CorrectEventTriggerType : (EventTriggerType <>
    IfcEventTriggerTypeEnum.USERDEFINED) OR ((EventTriggerType =
    IfcEventTriggerTypeEnum.USERDEFINED) AND EXISTS(UserDefinedEventTriggerType));
END_ENTITY;

ENTITY IfcExtendedProperties
ABSTRACT SUPERTYPE OF (ONEOF
    (IfcMaterialProperties
    , IfcProfileProperties))
SUBTYPE OF (IfcPropertyAbstraction);
    Name : OPTIONAL IfcIdentifier;
    Description : OPTIONAL IfcText;
    Properties : SET [1:?] OF IfcProperty;
END_ENTITY;

ENTITY IfcExternalInformation
ABSTRACT SUPERTYPE OF (ONEOF
    (IfcClassification
    , IfcDocumentInformation
    , IfcLibraryInformation));
END_ENTITY;

ENTITY IfcExternalReference
ABSTRACT SUPERTYPE OF (ONEOF
    (IfcClassificationReference
    , IfcDocumentReference
    , IfcExternallyDefinedHatchStyle
    , IfcExternallyDefinedSurfaceStyle
    , IfcExternallyDefinedTextFont
    , IfcLibraryReference));
    Location : OPTIONAL IfcURIReference;
    Identification : OPTIONAL IfcIdentifier;
    Name : OPTIONAL IfcLabel;
INVERSE
    ExternalReferenceForResources : SET [0:?] OF IfcExternalReferenceRelationship
FOR RelatingReference;
WHERE
    WR1 : EXISTS(Identification) OR EXISTS(Location) OR EXISTS(Name);
END_ENTITY;

```

---

```

ENTITY IfcExternalReferenceRelationship
  SUBTYPE OF (IfcResourceLevelRelationship);
    RelatingReference : IfcExternalReference;
    RelatedResourceObjects : SET [1:?] OF IfcResourceObjectSelect;
END_ENTITY;

ENTITY IfcExternalSpatialElement
  SUBTYPE OF (IfcExternalSpatialStructureElement);
    PredefinedType : OPTIONAL IfcExternalSpatialElementTypeEnum;
  INVERSE
    BoundedBy : SET [0:?] OF IfcRelSpaceBoundary FOR RelatingSpace;
END_ENTITY;

ENTITY IfcExternalSpatialStructureElement
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcExternalSpatialElement))
  SUBTYPE OF (IfcSpatialElement);
END_ENTITY;

ENTITY IfcExternallyDefinedHatchStyle
  SUBTYPE OF (IfcExternalReference);
END_ENTITY;

ENTITY IfcExternallyDefinedSurfaceStyle
  SUBTYPE OF (IfcExternalReference);
END_ENTITY;

ENTITY IfcExternallyDefinedTextFont
  SUBTYPE OF (IfcExternalReference);
END_ENTITY;

ENTITY IfcExtrudedAreaSolid
  SUPERTYPE OF (ONEOF
    (IfcExtrudedAreaSolidTapered))
  SUBTYPE OF (IfcSweptAreaSolid);
    ExtrudedDirection : IfcDirection;
    Depth : IfcPositiveLengthMeasure;
  WHERE
    ValidExtrusionDirection : IfcDotProduct(IfcRepresentationItem() ||
IfcGeometricRepresentationItem() || IfcDirection([0.0, 0.0, 1.0]),
SELF.ExtrudedDirection) <> 0.0;
END_ENTITY;

```

---

```

ENTITY IfcExtrudedAreaSolidTapered
  SUBTYPE OF (IfcExtrudedAreaSolid);
    EndSweptArea : IfcProfileDef;
  WHERE
    CorrectProfileAssignment :
IfcTaperedSweptAreaProfiles(SELf\IfcSweptAreaSolid. SweptArea,
SELF. EndSweptArea);
END_ENTITY;

ENTITY IfcFace
  SUPERTYPE OF (ONEOF
    (IfcFaceSurface))
  SUBTYPE OF (IfcTopologicalRepresentationItem);
    Bounds : SET [1:?] OF IfcFaceBound;
  INVERSE
    HasTextureMaps : SET [0:?] OF IfcTextureMap FOR MappedTo;
  WHERE
    HasOuterBound : SIZEOF(QUERY(temp <* Bounds | ' IFC4. IFCFACEOUTERBOUND' IN
TYPEOF(temp))) <= 1;
END_ENTITY;

ENTITY IfcFaceBasedSurfaceModel
  SUBTYPE OF (IfcGeometricRepresentationItem);
    FbsmFaces : SET [1:?] OF IfcConnectedFaceSet;
  DERIVE
    Dim : IfcDimensionCount := 3;
END_ENTITY;

ENTITY IfcFaceBound
  SUPERTYPE OF (ONEOF
    (IfcFaceOuterBound))
  SUBTYPE OF (IfcTopologicalRepresentationItem);
    Bound : IfcLoop;
    Orientation : IfcBoolean;
END_ENTITY;

ENTITY IfcFaceOuterBound
  SUBTYPE OF (IfcFaceBound);
END_ENTITY;

ENTITY IfcFaceSurface
  SUPERTYPE OF (ONEOF
    (IfcAdvancedFace))
  SUBTYPE OF (IfcFace);

```



---

```

    FaceSurface : IfcSurface;
    SameSense : IfcBoolean;
END_ENTITY;

ENTITY IfcFacetedBrep
  SUPERTYPE OF (ONEOF
    (IfcFacetedBrepWithVoids))
  SUBTYPE OF (IfcManifoldSolidBrep);
END_ENTITY;

ENTITY IfcFacetedBrepWithVoids
  SUBTYPE OF (IfcFacetedBrep);
    Voids : SET [1:?] OF IfcClosedShell;
END_ENTITY;

ENTITY IfcFailureConnectionCondition
  SUBTYPE OF (IfcStructuralConnectionCondition);
    TensionFailureX : OPTIONAL IfcForceMeasure;
    TensionFailureY : OPTIONAL IfcForceMeasure;
    TensionFailureZ : OPTIONAL IfcForceMeasure;
    CompressionFailureX : OPTIONAL IfcForceMeasure;
    CompressionFailureY : OPTIONAL IfcForceMeasure;
    CompressionFailureZ : OPTIONAL IfcForceMeasure;
END_ENTITY;

ENTITY IfcFan
  SUBTYPE OF (IfcFlowMovingDevice);
    PredefinedType : OPTIONAL IfcFanTypeEnum;
  WHERE
    CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
    (PredefinedType <> IfcFanTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcFanTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCFANTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcFanType
  SUBTYPE OF (IfcFlowMovingDeviceType);
    PredefinedType : IfcFanTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcFanTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcFanTypeEnum.USERDEFINED) AND
    EXISTS (SELF\IfcElementType.ElementType));

```

---

```

END_ENTITY;

ENTITY IfcFastener
  SUBTYPE OF (IfcElementComponent);
    PredefinedType : OPTIONAL IfcFastenerTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcFastenerTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcFastenerTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCFASTENERTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcFastenerType
  SUBTYPE OF (IfcElementComponentType);
    PredefinedType : IfcFastenerTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcFastenerTypeEnum.USERDEFINED)
OR
    ((PredefinedType = IfcFastenerTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcFeatureElement
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcFeatureElementAddition
    , IfcFeatureElementSubtraction
    , IfcSurfaceFeature))
  SUBTYPE OF (IfcElement);
END_ENTITY;

ENTITY IfcFeatureElementAddition
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcProjectionElement))
  SUBTYPE OF (IfcFeatureElement);
  INVERSE
    ProjectsElements : IfcRelProjectsElement FOR RelatedFeatureElement;
END_ENTITY;

ENTITY IfcFeatureElementSubtraction
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcOpeningElement
    , IfcVoidingFeature))

```

---

```

SUBTYPE OF (IfcFeatureElement);
INVERSE
    VoidsElements : IfcRelVoidsElement FOR RelatedOpeningElement;
WHERE
    HasNoSubtraction : SIZEOF(SELF\IfcElement.HasOpenings) = 0;
    IsNotFilling : SIZEOF(SELF\IfcElement.FillsVoids) = 0;
END_ENTITY;

ENTITY IfcFillAreaStyle
SUBTYPE OF (IfcPresentationStyle);
    FillStyles : SET [1:?] OF IfcFillStyleSelect;
    ModelorDraughting : OPTIONAL IfcBoolean;
WHERE
    MaxOneColour : SIZEOF(QUERY(Style <* SELF.FillStyles |
' IFC4. IFCCOLOUR' IN
    TYPEOF(Style)
    )) <= 1;
    MaxOneExtHatchStyle : SIZEOF(QUERY(Style <* SELF.FillStyles |
' IFC4. IFCEXTERNALLYDEFINEDHATCHSTYLE' IN
    TYPEOF(Style)
    )) <= 1;
    ConsistentHatchStyleDef : IfcCorrectFillAreaStyle(SELF.FillStyles);
END_ENTITY;

ENTITY IfcFillAreaStyleHatching
SUBTYPE OF (IfcGeometricRepresentationItem);
    HatchLineAppearance : IfcCurveStyle;
    StartOfNextHatchLine : IfcHatchLineDistanceSelect;
    PointOfReferenceHatchLine : OPTIONAL IfcCartesianPoint;
    PatternStart : OPTIONAL IfcCartesianPoint;
    HatchLineAngle : IfcPlaneAngleMeasure;
WHERE
    PatternStart2D : NOT(EXISTS(PatternStart)) OR (PatternStart.Dim = 2);
    RefHatchLine2D : NOT(EXISTS(PointOfReferenceHatchLine)) OR
(PointOfReferenceHatchLine.Dim = 2);
END_ENTITY;

ENTITY IfcFillAreaStyleTiles
SUBTYPE OF (IfcGeometricRepresentationItem);
    TilingPattern : LIST [2:2] OF IfcVector;
    Tiles : SET [1:?] OF IfcStyledItem;
    TilingScale : IfcPositiveRatioMeasure;
END_ENTITY;

```

---

```

ENTITY IfcFilter
  SUBTYPE OF (IfcFlowTreatmentDevice);
    PredefinedType : OPTIONAL IfcFilterTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcFilterTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcFilterTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCFILTERTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcFilterType
  SUBTYPE OF (IfcFlowTreatmentDeviceType);
    PredefinedType : IfcFilterTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcFilterTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcFilterTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcFireSuppressionTerminal
  SUBTYPE OF (IfcFlowTerminal);
    PredefinedType : OPTIONAL IfcFireSuppressionTerminalTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcFireSuppressionTerminalTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcFireSuppressionTerminalTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCFIRESUPPRESSIONTERMINALTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcFireSuppressionTerminalType
  SUBTYPE OF (IfcFlowTerminalType);
    PredefinedType : IfcFireSuppressionTerminalTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <>
    IfcFireSuppressionTerminalTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcFireSuppressionTerminalTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

---

```

ENTITY IfcFixedReferenceSweptAreaSolid
  SUBTYPE OF (IfcSweptAreaSolid);
    Directrix : IfcCurve;
    StartParam : OPTIONAL IfcParameterValue;
    EndParam : OPTIONAL IfcParameterValue;
    FixedReference : IfcDirection;
  WHERE
    DirectrixBounded : (EXISTS(StartParam) AND EXISTS(EndParam)) OR
    (SIZEOF([' IFC4. IFCCONIC', ' IFC4. IFCBOUNDEDCURVE' ] * TYPEOF(Directrix)) = 1);
END_ENTITY;

```

```

ENTITY IfcFlowController
  SUPERTYPE OF (ONEOF
    (IfcAirTerminalBox
    , IfcDamper
    , IfcElectricDistributionBoard
    , IfcElectricTimeControl
    , IfcFlowMeter
    , IfcProtectiveDevice
    , IfcSwitchingDevice
    , IfcValve))
  SUBTYPE OF (IfcDistributionFlowElement);
END_ENTITY;

```

```

ENTITY IfcFlowControllerType
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcAirTerminalBoxType
    , IfcDamperType
    , IfcElectricDistributionBoardType
    , IfcElectricTimeControlType
    , IfcFlowMeterType
    , IfcProtectiveDeviceType
    , IfcSwitchingDeviceType
    , IfcValveType))
  SUBTYPE OF (IfcDistributionFlowElementType);
END_ENTITY;

```

```

ENTITY IfcFlowFitting
  SUPERTYPE OF (ONEOF
    (IfcCableCarrierFitting
    , IfcCableFitting
    , IfcDuctFitting
    , IfcJunctionBox
    , IfcPipeFitting))

```

---

```

    SUBTYPE OF (IfcDistributionFlowElement);
END_ENTITY;

ENTITY IfcFlowFittingType
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcCableCarrierFittingType
        , IfcCableFittingType
        , IfcDuctFittingType
        , IfcJunctionBoxType
        , IfcPipeFittingType))
    SUBTYPE OF (IfcDistributionFlowElementType);
END_ENTITY;

ENTITY IfcFlowInstrument
    SUBTYPE OF (IfcDistributionControlElement);
    PredefinedType : OPTIONAL IfcFlowInstrumentTypeEnum;
    WHERE
        CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
        (PredefinedType <> IfcFlowInstrumentTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcFlowInstrumentTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject.ObjectType));
        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
        ('IFC4.IFCFLOWINSTRUMENTTYPE' IN
        TYPEOF (SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcFlowInstrumentType
    SUBTYPE OF (IfcDistributionControlElementType);
    PredefinedType : IfcFlowInstrumentTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <>
        IfcFlowInstrumentTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcFlowInstrumentTypeEnum.USERDEFINED) AND
        EXISTS (SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcFlowMeter
    SUBTYPE OF (IfcFlowController);
    PredefinedType : OPTIONAL IfcFlowMeterTypeEnum;
    WHERE
        CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
        (PredefinedType <> IfcFlowMeterTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcFlowMeterTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject.ObjectType));

```

---

```
        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR  
        ('IFC4.IFCFLOWMETERTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));  
END_ENTITY;
```

```
ENTITY IfcFlowMeterType  
    SUBTYPE OF (IfcFlowControllerType);  
        PredefinedType : IfcFlowMeterTypeEnum;  
    WHERE  
        CorrectPredefinedType : (PredefinedType <> IfcFlowMeterTypeEnum.USERDEFINED)  
OR  
        ((PredefinedType = IfcFlowMeterTypeEnum.USERDEFINED) AND  
        EXISTS(SELF\IfcElementType.ElementType));  
END_ENTITY;
```

```
ENTITY IfcFlowMovingDevice  
    SUPERTYPE OF (ONEOF  
        (IfcCompressor  
        , IfcFan  
        , IfcPump))  
    SUBTYPE OF (IfcDistributionFlowElement);  
END_ENTITY;
```

```
ENTITY IfcFlowMovingDeviceType  
    ABSTRACT SUPERTYPE OF (ONEOF  
        (IfcCompressorType  
        , IfcFanType  
        , IfcPumpType))  
    SUBTYPE OF (IfcDistributionFlowElementType);  
END_ENTITY;
```

```
ENTITY IfcFlowSegment  
    SUPERTYPE OF (ONEOF  
        (IfcCableCarrierSegment  
        , IfcCableSegment  
        , IfcDuctSegment  
        , IfcPipeSegment))  
    SUBTYPE OF (IfcDistributionFlowElement);  
END_ENTITY;
```

```
ENTITY IfcFlowSegmentType  
    ABSTRACT SUPERTYPE OF (ONEOF  
        (IfcCableCarrierSegmentType  
        , IfcCableSegmentType  
        , IfcDuctSegmentType
```

---

```
        , IfcPipeSegmentType))
    SUBTYPE OF (IfcDistributionFlowElementType);
END_ENTITY;
```

```
ENTITY IfcFlowStorageDevice
    SUPERTYPE OF (ONEOF
        (IfcElectricFlowStorageDevice
        , IfcTank))
    SUBTYPE OF (IfcDistributionFlowElement);
END_ENTITY;
```

```
ENTITY IfcFlowStorageDeviceType
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcElectricFlowStorageDeviceType
        , IfcTankType))
    SUBTYPE OF (IfcDistributionFlowElementType);
END_ENTITY;
```

```
ENTITY IfcFlowTerminal
    SUPERTYPE OF (ONEOF
        (IfcAirTerminal
        , IfcAudioVisualAppliance
        , IfcCommunicationsAppliance
        , IfcElectricAppliance
        , IfcFireSuppressionTerminal
        , IfcLamp
        , IfcLightFixture
        , IfcMedicalDevice
        , IfcOutlet
        , IfcSanitaryTerminal
        , IfcSpaceHeater
        , IfcStackTerminal
        , IfcWasteTerminal))
    SUBTYPE OF (IfcDistributionFlowElement);
END_ENTITY;
```

```
ENTITY IfcFlowTerminalType
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcAirTerminalType
        , IfcAudioVisualApplianceType
        , IfcCommunicationsApplianceType
        , IfcElectricApplianceType
        , IfcFireSuppressionTerminalType
        , IfcLampType
```



---

```

    , IfcLightFixtureType
    , IfcMedicalDeviceType
    , IfcOutletType
    , IfcSanitaryTerminalType
    , IfcSpaceHeaterType
    , IfcStackTerminalType
    , IfcWasteTerminalType))
SUBTYPE OF (IfcDistributionFlowElementType);
END_ENTITY;

ENTITY IfcFlowTreatmentDevice
SUPERTYPE OF (ONEOF
    (IfcDuctSilencer
    , IfcFilter
    , IfcInterceptor))
SUBTYPE OF (IfcDistributionFlowElement);
END_ENTITY;

ENTITY IfcFlowTreatmentDeviceType
ABSTRACT SUPERTYPE OF (ONEOF
    (IfcDuctSilencerType
    , IfcFilterType
    , IfcInterceptorType))
SUBTYPE OF (IfcDistributionFlowElementType);
END_ENTITY;

ENTITY IfcFooting
SUBTYPE OF (IfcBuildingElement);
    PredefinedType : OPTIONAL IfcFootingTypeEnum;
WHERE
    CorrectPredefinedType : NOT EXISTS(PredefinedType) OR
    (PredefinedType <> IfcFootingTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcFootingTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCFOOTINGTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcFootingType
SUBTYPE OF (IfcBuildingElementType);
    PredefinedType : IfcFootingTypeEnum;
WHERE
    CorrectPredefinedType : (PredefinedType <> IfcFootingTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcFootingTypeEnum.USERDEFINED) AND

```

---

```
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;
```

```
ENTITY IfcFurnishingElement
  SUPERTYPE OF (ONEOF
    (IfcFurniture
      , IfcSystemFurnitureElement))
  SUBTYPE OF (IfcElement);
END_ENTITY;
```

```
ENTITY IfcFurnishingElementType
  SUPERTYPE OF (ONEOF
    (IfcFurnitureType
      , IfcSystemFurnitureElementType))
  SUBTYPE OF (IfcElementType);
END_ENTITY;
```

```
ENTITY IfcFurniture
  SUBTYPE OF (IfcFurnishingElement);
  PredefinedType : OPTIONAL IfcFurnitureTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcFurnitureTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcFurnitureTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    (' IFC4. IFCFURNITURETYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;
```

```
ENTITY IfcFurnitureType
  SUBTYPE OF (IfcFurnishingElementType);
  AssemblyPlace : IfcAssemblyPlaceEnum;
  PredefinedType : OPTIONAL IfcFurnitureTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcFurnitureTypeEnum.USERDEFINED)
OR
    ((PredefinedType = IfcFurnitureTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;
```

```
ENTITY IfcGeographicElement
  SUBTYPE OF (IfcElement);
  PredefinedType : OPTIONAL IfcGeographicElementTypeEnum;
```

---

```

WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcGeographicElementTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcGeographicElementTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    (' IFC4. IFCGEOGRAPHICELEMENTTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcGeographicElementType
    SUBTYPE OF (IfcElementType);
    PredefinedType : IfcGeographicElementTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <>
    IfcGeographicElementTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcGeographicElementTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcGeometricCurveSet
    SUBTYPE OF (IfcGeometricSet);
    WHERE
        NoSurfaces : SIZEOF(QUERY(Temp < * SELF\IfcGeometricSet.Elements |
    ' IFC4. IFCSURFACE' IN TYPEOF(Temp))) = 0;
END_ENTITY;

```

```

ENTITY IfcGeometricRepresentationContext
    SUPERTYPE OF (ONEOF
    (IfcGeometricRepresentationSubContext))
    SUBTYPE OF (IfcRepresentationContext);
    CoordinateSpaceDimension : IfcDimensionCount;
    Precision : OPTIONAL IfcReal;
    WorldCoordinateSystem : IfcAxis2Placement;
    TrueNorth : OPTIONAL IfcDirection;
    INVERSE
        HasSubContexts : SET [0:?] OF IfcGeometricRepresentationSubContext FOR
    ParentContext;
        HasCoordinateOperation : SET [0:1] OF IfcCoordinateOperation FOR SourceCRS;
    WHERE
        North2D : NOT(EXISTS(TrueNorth)) OR (HIINDEX(TrueNorth.DirectionRatios) = 2);
END_ENTITY;

```

```

ENTITY IfcGeometricRepresentationItem

```

---

```

ABSTRACT SUPERTYPE OF (ONEOF
  (IfcAnnotationFillArea
  , IfcBooleanResult
  , IfcBoundingBox
  , IfcCartesianPointList
  , IfcCartesianTransformationOperator
  , IfcCompositeCurveSegment
  , IfcCsgPrimitive3D
  , IfcCurve
  , IfcDirection
  , IfcFaceBasedSurfaceModel
  , IfcFillAreaStyleHatching
  , IfcFillAreaStyleTiles
  , IfcGeometricSet
  , IfcHalfSpaceSolid
  , IfcLightSource
  , IfcPlacement
  , IfcPlanarExtent
  , IfcPoint
  , IfcSectionedSpine
  , IfcShellBasedSurfaceModel
  , IfcSolidModel
  , IfcSurface
  , IfcTessellatedItem
  , IfcTextLiteral
  , IfcVector))
SUBTYPE OF (IfcRepresentationItem);
END_ENTITY;

ENTITY IfcGeometricRepresentationSubContext
  SUBTYPE OF (IfcGeometricRepresentationContext);
    ParentContext : IfcGeometricRepresentationContext;
    TargetScale : OPTIONAL IfcPositiveRatioMeasure;
    TargetView : IfcGeometricProjectionEnum;
    UserDefinedTargetView : OPTIONAL IfcLabel;
  DERIVE
    SELF\IfcGeometricRepresentationContext.WorldCoordinateSystem      :
IfcAxis2Placement := ParentContext.WorldCoordinateSystem;
    SELF\IfcGeometricRepresentationContext.CoordinateSpaceDimension    :
IfcDimensionCount := ParentContext.CoordinateSpaceDimension;
    SELF\IfcGeometricRepresentationContext.TrueNorth      : IfcDirection      :=
NVL (ParentContext.TrueNorth,
IfcConvertDirectionInto2D(SELF\IfcGeometricRepresentationContext.WorldCoordinat
eSystem.P[2]));

```

---

```

        SELF\IfcGeometricRepresentationContext.Precision      :      IfcReal      :=
NVL (ParentContext.Precision, 1. E-5);
    WHERE
        ParentNoSub      :      NOT(' IFC4. IFCGEOMETRICREPRESENTATIONSUBCONTEXT'      IN
TYPEOF(ParentContext));
        UserTargetProvided : (TargetView <> IfcGeometricProjectionEnum. USERDEFINED)
OR
((TargetView      =      IfcGeometricProjectionEnum. USERDEFINED)      AND
EXISTS(UserDefinedTargetView));
        NoCoordOperation      :
SIZEOF (SELF\IfcGeometricRepresentationContext.HasCoordinateOperation) = 0;
END_ENTITY;

```

```

ENTITY IfcGeometricSet
    SUPERTYPE OF (ONEOF
        (IfcGeometricCurveSet))
    SUBTYPE OF (IfcGeometricRepresentationItem);
    Elements : SET [1:?] OF IfcGeometricSetSelect;
    DERIVE
        Dim : IfcDimensionCount := Elements[1].Dim;
    WHERE
        ConsistentDim : SIZEOF(QUERY(Temp < * Elements |
        Temp.Dim <> Elements[1].Dim))
= 0;
END_ENTITY;

```

```

ENTITY IfcGrid
    SUBTYPE OF (IfcProduct);
    UAxes : LIST [1:?] OF UNIQUE IfcGridAxis;
    VAxes : LIST [1:?] OF UNIQUE IfcGridAxis;
    WAxes : OPTIONAL LIST [1:?] OF UNIQUE IfcGridAxis;
    PredefinedType : OPTIONAL IfcGridTypeEnum;
    INVERSE
        ContainedInStructure : SET [0:1] OF IfcRelContainedInSpatialStructure FOR
RelatedElements;
    WHERE
        HasPlacement : EXISTS (SELF\IfcProduct.ObjectPlacement);
END_ENTITY;

```

```

ENTITY IfcGridAxis;
    AxisTag : OPTIONAL IfcLabel;
    AxisCurve : IfcCurve;
    SameSense : IfcBoolean;
    INVERSE

```

---

```

    PartOfW : SET [0:1] OF IfcGrid FOR WAxes;
    PartOfV : SET [0:1] OF IfcGrid FOR VAxes;
    PartOfU : SET [0:1] OF IfcGrid FOR UAxes;
    HasIntersections : SET [0:?] OF IfcVirtualGridIntersection FOR
IntersectingAxes;
WHERE
    WR1 : AxisCurve.Dim = 2;
    WR2 : (SIZEOF(PartOfU) = 1) XOR (SIZEOF(PartOfV) = 1) XOR (SIZEOF(PartOfW) =
1);
END_ENTITY;

ENTITY IfcGridPlacement
    SUBTYPE OF (IfcObjectPlacement);
    PlacementLocation : IfcVirtualGridIntersection;
    PlacementRefDirection : OPTIONAL IfcGridPlacementDirectionSelect;
END_ENTITY;

ENTITY IfcGroup
    SUPERTYPE OF (ONEOF
        (IfcAsset
        , IfcInventory
        , IfcStructuralLoadGroup
        , IfcStructuralResultGroup
        , IfcSystem))
    SUBTYPE OF (IfcObject);
    INVERSE
        IsGroupedBy : SET [0:?] OF IfcRelAssignsToGroup FOR RelatingGroup;
END_ENTITY;

ENTITY IfcHalfSpaceSolid
    SUPERTYPE OF (ONEOF
        (IfcBoxedHalfSpace
        , IfcPolygonalBoundedHalfSpace))
    SUBTYPE OF (IfcGeometricRepresentationItem);
    BaseSurface : IfcSurface;
    AgreementFlag : IfcBoolean;
    DERIVE
        Dim : IfcDimensionCount := 3;
END_ENTITY;

ENTITY IfcHeatExchanger
    SUBTYPE OF (IfcEnergyConversionDevice);
    PredefinedType : OPTIONAL IfcHeatExchangerTypeEnum;
    WHERE

```

---

```

        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
        (PredefinedType <> IfcHeatExchangerTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcHeatExchangerTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject.ObjectType));
        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
        (' IFC4. IFCHEATEXCHANGERTYPE' IN
        TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcHeatExchangerType
    SUBTYPE OF (IfcEnergyConversionDeviceType);
    PredefinedType : IfcHeatExchangerTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <>
        IfcHeatExchangerTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcHeatExchangerTypeEnum.USERDEFINED) AND
        EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcHumidifier
    SUBTYPE OF (IfcEnergyConversionDevice);
    PredefinedType : OPTIONAL IfcHumidifierTypeEnum;
    WHERE
        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
        (PredefinedType <> IfcHumidifierTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcHumidifierTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject.ObjectType));
        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
        (' IFC4. IFCHUMIDIFIERTYPE' IN
        TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcHumidifierType
    SUBTYPE OF (IfcEnergyConversionDeviceType);
    PredefinedType : IfcHumidifierTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <> IfcHumidifierTypeEnum.USERDEFINED)
    OR
        ((PredefinedType = IfcHumidifierTypeEnum.USERDEFINED) AND
        EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcIShapeProfileDef
    SUBTYPE OF (IfcParameterizedProfileDef);

```

---

```

    OverallWidth : IfcPositiveLengthMeasure;
    OverallDepth : IfcPositiveLengthMeasure;
    WebThickness : IfcPositiveLengthMeasure;
    FlangeThickness : IfcPositiveLengthMeasure;
    FilletRadius : OPTIONAL IfcNonNegativeLengthMeasure;
    FlangeEdgeRadius : OPTIONAL IfcNonNegativeLengthMeasure;
    FlangeSlope : OPTIONAL IfcPlaneAngleMeasure;
WHERE
    ValidFlangeThickness : (2. * FlangeThickness) < OverallDepth;
    ValidWebThickness : WebThickness < OverallWidth;
    ValidFilletRadius : NOT(EXISTS(FilletRadius)) OR
((FilletRadius <= (OverallWidth - WebThickness)/2.) AND
(FilletRadius <= (OverallDepth - (2. * FlangeThickness))/2.));
END_ENTITY;

ENTITY IfcImageTexture
    SUBTYPE OF (IfcSurfaceTexture);
    URLReference : IfcURIReference;
END_ENTITY;

ENTITY IfcIndexedColourMap
    SUBTYPE OF (IfcPresentationItem);
    MappedTo : IfcTessellatedFaceSet;
    Opacity : OPTIONAL IfcNormalisedRatioMeasure;
    Colours : IfcColourRgbList;
    ColourIndex : LIST [1:?] OF IfcPositiveInteger;
END_ENTITY;

ENTITY IfcIndexedPolyCurve
    SUBTYPE OF (IfcBoundedCurve);
    Points : IfcCartesianPointList;
    Segments : OPTIONAL LIST [1:?] OF IfcSegmentIndexSelect;
    SelfIntersect : OPTIONAL IfcBoolean;
WHERE
    Consecutive : (SIZEOF(Segments) = 0) OR IfcConsecutiveSegments(Segments);
END_ENTITY;

ENTITY IfcIndexedPolygonalFace
    SUPERTYPE OF (ONEOF
        (IfcIndexedPolygonalFaceWithVoids))
    SUBTYPE OF (IfcTessellatedItem);
    CoordIndex : LIST [3:?] OF IfcPositiveInteger;
INVERSE
    ToFaceSet : SET [1:?] OF IfcPolygonalFaceSet FOR Faces;

```



---

```

END_ENTITY;

ENTITY IfcIndexedPolygonalFaceWithVoids
  SUBTYPE OF (IfcIndexedPolygonalFace);
    InnerCoordIndices : LIST [1:?] OF LIST [3:?] OF UNIQUE IfcPositiveInteger;
END_ENTITY;

ENTITY IfcIndexedTextureMap
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcIndexedTriangleTextureMap))
  SUBTYPE OF (IfcTextureCoordinate);
    MappedTo : IfcTessellatedFaceSet;
    TexCoords : IfcTextureVertexList;
END_ENTITY;

ENTITY IfcIndexedTriangleTextureMap
  SUBTYPE OF (IfcIndexedTextureMap);
    TexCoordIndex : OPTIONAL LIST [1:?] OF LIST [3:3] OF IfcPositiveInteger;
END_ENTITY;

ENTITY IfcInterceptor
  SUBTYPE OF (IfcFlowTreatmentDevice);
    PredefinedType : OPTIONAL IfcInterceptorTypeEnum;
  WHERE
    CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
    (PredefinedType <> IfcInterceptorTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcInterceptorTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCINTERCEPTORTYPE' IN
    TYPEOF (SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcInterceptorType
  SUBTYPE OF (IfcFlowTreatmentDeviceType);
    PredefinedType : IfcInterceptorTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcInterceptorTypeEnum.USERDEFINED)
  OR
    ((PredefinedType = IfcInterceptorTypeEnum.USERDEFINED) AND
    EXISTS (SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcIntersectionCurve

```

---

```

SUBTYPE OF (IfcSurfaceCurve);
WHERE
    TwoPCurves : SIZEOF(SELF\IfcSurfaceCurve.AssociatedGeometry) = 2;
    DistinctSurfaces :
IfcAssociatedSurface(SELF\IfcSurfaceCurve.AssociatedGeometry[1]) <>
IfcAssociatedSurface(SELF\IfcSurfaceCurve.AssociatedGeometry[2]);
END_ENTITY;

ENTITY IfcInventory
SUBTYPE OF (IfcGroup);
    PredefinedType : OPTIONAL IfcInventoryTypeEnum;
    Jurisdiction : OPTIONAL IfcActorSelect;
    ResponsiblePersons : OPTIONAL SET [1:?] OF IfcPerson;
    LastUpdateDate : OPTIONAL IfcDate;
    CurrentValue : OPTIONAL IfcCostValue;
    OriginalValue : OPTIONAL IfcCostValue;
END_ENTITY;

ENTITY IfcIrregularTimeSeries
SUBTYPE OF (IfcTimeSeries);
    Values : LIST [1:?] OF IfcIrregularTimeSeriesValue;
END_ENTITY;

ENTITY IfcIrregularTimeSeriesValue;
    TimeStamp : IfcDateTime;
    ListValues : LIST [1:?] OF IfcValue;
END_ENTITY;

ENTITY IfcJunctionBox
SUBTYPE OF (IfcFlowFitting);
    PredefinedType : OPTIONAL IfcJunctionBoxTypeEnum;
WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcJunctionBoxTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcJunctionBoxTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCJUNCTIONBOXTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcJunctionBoxType
SUBTYPE OF (IfcFlowFittingType);
    PredefinedType : IfcJunctionBoxTypeEnum;

```

---

```

WHERE
    CorrectPredefinedType : (PredefinedType <> IfcJunctionBoxTypeEnum.USERDEFINED)
OR
    ((PredefinedType      =      IfcJunctionBoxTypeEnum.USERDEFINED)      AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcLShapeProfileDef
    SUBTYPE OF (IfcParameterizedProfileDef);
    Depth : IfcPositiveLengthMeasure;
    Width : OPTIONAL IfcPositiveLengthMeasure;
    Thickness : IfcPositiveLengthMeasure;
    FilletRadius : OPTIONAL IfcNonNegativeLengthMeasure;
    EdgeRadius : OPTIONAL IfcNonNegativeLengthMeasure;
    LegSlope : OPTIONAL IfcPlaneAngleMeasure;
    WHERE
        ValidThickness : (Thickness < Depth) AND (NOT(EXISTS(Width)) OR (Thickness <
Width));
    END_ENTITY;

ENTITY IfcLaborResource
    SUBTYPE OF (IfcConstructionResource);
    PredefinedType : OPTIONAL IfcLaborResourceTypeEnum;
    WHERE
        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
        (PredefinedType <> IfcLaborResourceTypeEnum.USERDEFINED) OR
        ((PredefinedType      =      IfcLaborResourceTypeEnum.USERDEFINED)      AND      EXISTS
        (SELF\IfcObject.ObjectType));
    END_ENTITY;

ENTITY IfcLaborResourceType
    SUBTYPE OF (IfcConstructionResourceType);
    PredefinedType : IfcLaborResourceTypeEnum;
    WHERE
        CorrectPredefinedType      :      (PredefinedType      <>
IfcLaborResourceTypeEnum.USERDEFINED) OR
        ((PredefinedType      =      IfcLaborResourceTypeEnum.USERDEFINED)      AND
        EXISTS(SELF\IfcTypeResource.ResourceType));
    END_ENTITY;

ENTITY IfcLagTime
    SUBTYPE OF (IfcSchedulingTime);
    LagValue : IfcTimeOrRatioSelect;
    DurationType : IfcTaskDurationEnum;

```

---

```

END_ENTITY;

ENTITY IfcLamp
  SUBTYPE OF (IfcFlowTerminal);
    PredefinedType : OPTIONAL IfcLampTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcLampTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcLampTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCLAMPTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcLampType
  SUBTYPE OF (IfcFlowTerminalType);
    PredefinedType : IfcLampTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcLampTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcLampTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcLibraryInformation
  SUBTYPE OF (IfcExternalInformation);
    Name : IfcLabel;
    Version : OPTIONAL IfcLabel;
    Publisher : OPTIONAL IfcActorSelect;
    VersionDate : OPTIONAL IfcDateTime;
    Location : OPTIONAL IfcURIReference;
    Description : OPTIONAL IfcText;
  INVERSE
    LibraryInfoForObjects : SET [0:?] OF IfcRelAssociatesLibrary FOR
    RelatingLibrary;
    HasLibraryReferences : SET [0:?] OF IfcLibraryReference FOR ReferencedLibrary;
END_ENTITY;

ENTITY IfcLibraryReference
  SUBTYPE OF (IfcExternalReference);
    Description : OPTIONAL IfcText;
    Language : OPTIONAL IfcLanguageId;
    ReferencedLibrary : OPTIONAL IfcLibraryInformation;
  INVERSE
    LibraryRefForObjects : SET [0:?] OF IfcRelAssociatesLibrary FOR

```

---

```

RelatingLibrary;
END_ENTITY;

ENTITY IfcLightDistributionData;
    MainPlaneAngle : IfcPlaneAngleMeasure;
    SecondaryPlaneAngle : LIST [1:?] OF IfcPlaneAngleMeasure;
    LuminousIntensity : LIST [1:?] OF IfcLuminousIntensityDistributionMeasure;
END_ENTITY;

ENTITY IfcLightFixture
    SUBTYPE OF (IfcFlowTerminal);
    PredefinedType : OPTIONAL IfcLightFixtureTypeEnum;
    WHERE
        CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
        (PredefinedType <> IfcLightFixtureTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcLightFixtureTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCLIGHTFIXTURETYPE' IN
    TYPEOF (SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcLightFixtureType
    SUBTYPE OF (IfcFlowTerminalType);
    PredefinedType : IfcLightFixtureTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <>
        IfcLightFixtureTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcLightFixtureTypeEnum.USERDEFINED) AND
        EXISTS (SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcLightIntensityDistribution;
    LightDistributionCurve : IfcLightDistributionCurveEnum;
    DistributionData : LIST [1:?] OF IfcLightDistributionData;
END_ENTITY;

ENTITY IfcLightSource
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcLightSourceAmbient
        , IfcLightSourceDirectional
        , IfcLightSourceGoniometric
        , IfcLightSourcePositional))
    SUBTYPE OF (IfcGeometricRepresentationItem);

```

---

```

    Name : OPTIONAL IfcLabel;
    LightColour : IfcColourRgb;
    AmbientIntensity : OPTIONAL IfcNormalisedRatioMeasure;
    Intensity : OPTIONAL IfcNormalisedRatioMeasure;
END_ENTITY;

ENTITY IfcLightSourceAmbient
    SUBTYPE OF (IfcLightSource);
END_ENTITY;

ENTITY IfcLightSourceDirectional
    SUBTYPE OF (IfcLightSource);
    Orientation : IfcDirection;
END_ENTITY;

ENTITY IfcLightSourceGoniometric
    SUBTYPE OF (IfcLightSource);
    Position : IfcAxis2Placement3D;
    ColourAppearance : OPTIONAL IfcColourRgb;
    ColourTemperature : IfcThermodynamicTemperatureMeasure;
    LuminousFlux : IfcLuminousFluxMeasure;
    LightEmissionSource : IfcLightEmissionSourceEnum;
    LightDistributionDataSource : IfcLightDistributionDataSourceSelect;
END_ENTITY;

ENTITY IfcLightSourcePositional
    SUPERTYPE OF (ONEOF
        (IfcLightSourceSpot))
    SUBTYPE OF (IfcLightSource);
    Position : IfcCartesianPoint;
    Radius : IfcPositiveLengthMeasure;
    ConstantAttenuation : IfcReal;
    DistanceAttenuation : IfcReal;
    QuadricAttenuation : IfcReal;
END_ENTITY;

ENTITY IfcLightSourceSpot
    SUBTYPE OF (IfcLightSourcePositional);
    Orientation : IfcDirection;
    ConcentrationExponent : OPTIONAL IfcReal;
    SpreadAngle : IfcPositivePlaneAngleMeasure;
    BeamWidthAngle : IfcPositivePlaneAngleMeasure;
END_ENTITY;

```

---

```

ENTITY IfcLine
  SUBTYPE OF (IfcCurve);
    Pnt : IfcCartesianPoint;
    Dir : IfcVector;
  WHERE
    SameDim : Dir.Dim = Pnt.Dim;
END_ENTITY;

ENTITY IfcLocalPlacement
  SUBTYPE OF (IfcObjectPlacement);
    PlacementRelTo : OPTIONAL IfcObjectPlacement;
    RelativePlacement : IfcAxis2Placement;
  WHERE
    WR21 : IfcCorrectLocalPlacement(RelativePlacement, PlacementRelTo);
END_ENTITY;

ENTITY IfcLoop
  SUPERTYPE OF (ONEOF
    (IfcEdgeLoop
      , IfcPolyLoop
      , IfcVertexLoop))
  SUBTYPE OF (IfcTopologicalRepresentationItem);
END_ENTITY;

ENTITY IfcManifoldSolidBrep
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcAdvancedBrep
      , IfcFacetedBrep))
  SUBTYPE OF (IfcSolidModel);
    Outer : IfcClosedShell;
END_ENTITY;

ENTITY IfcMapConversion
  SUBTYPE OF (IfcCoordinateOperation);
    Eastings : IfcLengthMeasure;
    Northings : IfcLengthMeasure;
    OrthogonalHeight : IfcLengthMeasure;
    XAxisAbscissa : OPTIONAL IfcReal;
    XAxisOrdinate : OPTIONAL IfcReal;
    Scale : OPTIONAL IfcReal;
END_ENTITY;

ENTITY IfcMappedItem
  SUBTYPE OF (IfcRepresentationItem);

```

---

```

        MappingSource : IfcRepresentationMap;
        MappingTarget : IfcCartesianTransformationOperator;
END_ENTITY;

ENTITY IfcMaterial
    SUBTYPE OF (IfcMaterialDefinition);
        Name : IfcLabel;
        Description : OPTIONAL IfcText;
        Category : OPTIONAL IfcLabel;
    INVERSE
        HasRepresentation : SET [0:1] OF IfcMaterialDefinitionRepresentation FOR
RepresentedMaterial;
        IsRelatedWith : SET [0:?] OF IfcMaterialRelationship FOR RelatedMaterials;
        RelatesTo : SET [0:1] OF IfcMaterialRelationship FOR RelatingMaterial;
END_ENTITY;

ENTITY IfcMaterialClassificationRelationship;
    MaterialClassifications : SET [1:?] OF IfcClassificationSelect;
    ClassifiedMaterial : IfcMaterial;
END_ENTITY;

ENTITY IfcMaterialConstituent
    SUBTYPE OF (IfcMaterialDefinition);
        Name : OPTIONAL IfcLabel;
        Description : OPTIONAL IfcText;
        Material : IfcMaterial;
        Fraction : OPTIONAL IfcNormalisedRatioMeasure;
        Category : OPTIONAL IfcLabel;
    INVERSE
        ToMaterialConstituentSet      :      IfcMaterialConstituentSet      FOR
MaterialConstituents;
END_ENTITY;

ENTITY IfcMaterialConstituentSet
    SUBTYPE OF (IfcMaterialDefinition);
        Name : OPTIONAL IfcLabel;
        Description : OPTIONAL IfcText;
        MaterialConstituents : OPTIONAL SET [1:?] OF IfcMaterialConstituent;
END_ENTITY;

ENTITY IfcMaterialDefinition
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcMaterial
        , IfcMaterialConstituent

```



---

```

    , IfcMaterialConstituentSet
    , IfcMaterialLayer
    , IfcMaterialLayerSet
    , IfcMaterialProfile
    , IfcMaterialProfileSet));
INVERSE
    AssociatedTo : SET [0:?] OF IfcRelAssociatesMaterial FOR RelatingMaterial;
    HasExternalReferences : SET [0:?] OF IfcExternalReferenceRelationship FOR
RelatedResourceObjects;
    HasProperties : SET [0:?] OF IfcMaterialProperties FOR Material;
END_ENTITY;

ENTITY IfcMaterialDefinitionRepresentation
    SUBTYPE OF (IfcProductRepresentation);
    RepresentedMaterial : IfcMaterial;
    WHERE
        OnlyStyledRepresentations : SIZEOF(QUERY(temp <* Representations |
        (NOT('IFC4.IFCSTYLEDREPRESENTATION' IN TYPEOF(temp))))
    )) = 0;
END_ENTITY;

ENTITY IfcMaterialLayer
    SUPERTYPE OF (ONEOF
        (IfcMaterialLayerWithOffsets))
    SUBTYPE OF (IfcMaterialDefinition);
    Material : OPTIONAL IfcMaterial;
    LayerThickness : IfcNonNegativeLengthMeasure;
    IsVentilated : OPTIONAL IfcLogical;
    Name : OPTIONAL IfcLabel;
    Description : OPTIONAL IfcText;
    Category : OPTIONAL IfcLabel;
    Priority : OPTIONAL IfcInteger;
    INVERSE
        ToMaterialLayerSet : IfcMaterialLayerSet FOR MaterialLayers;
    WHERE
        NormalizedPriority : NOT(EXISTS(Priority)) OR {0 <= Priority <= 100};
END_ENTITY;

ENTITY IfcMaterialLayerSet
    SUBTYPE OF (IfcMaterialDefinition);
    MaterialLayers : LIST [1:?] OF IfcMaterialLayer;
    LayerSetName : OPTIONAL IfcLabel;
    Description : OPTIONAL IfcText;
    DERIVE

```

---

```

        TotalThickness : IfcLengthMeasure := IfcMlsTotalThickness(SELF);
END_ENTITY;

ENTITY IfcMaterialLayerSetUsage
  SUBTYPE OF (IfcMaterialUsageDefinition);
    ForLayerSet : IfcMaterialLayerSet;
    LayerSetDirection : IfcLayerSetDirectionEnum;
    DirectionSense : IfcDirectionSenseEnum;
    OffsetFromReferenceLine : IfcLengthMeasure;
    ReferenceExtent : OPTIONAL IfcPositiveLengthMeasure;
END_ENTITY;

ENTITY IfcMaterialLayerWithOffsets
  SUBTYPE OF (IfcMaterialLayer);
    OffsetDirection : IfcLayerSetDirectionEnum;
    OffsetValues : ARRAY [1:2] OF IfcLengthMeasure;
END_ENTITY;

ENTITY IfcMaterialList;
  Materials : LIST [1:?] OF IfcMaterial;
END_ENTITY;

ENTITY IfcMaterialProfile
  SUPERTYPE OF (ONEOF
    (IfcMaterialProfileWithOffsets))
  SUBTYPE OF (IfcMaterialDefinition);
    Name : OPTIONAL IfcLabel;
    Description : OPTIONAL IfcText;
    Material : OPTIONAL IfcMaterial;
    Profile : IfcProfileDef;
    Priority : OPTIONAL IfcInteger;
    Category : OPTIONAL IfcLabel;
  INVERSE
    ToMaterialProfileSet : IfcMaterialProfileSet FOR MaterialProfiles;
  WHERE
    NormalizedPriority : NOT(EXISTS(Priority)) OR {0 <= Priority <= 100};
END_ENTITY;

ENTITY IfcMaterialProfileSet
  SUBTYPE OF (IfcMaterialDefinition);
    Name : OPTIONAL IfcLabel;
    Description : OPTIONAL IfcText;
    MaterialProfiles : LIST [1:?] OF IfcMaterialProfile;
    CompositeProfile : OPTIONAL IfcCompositeProfileDef;

```

---

```

END_ENTITY;

ENTITY IfcMaterialProfileSetUsage
  SUPERTYPE OF (ONEOF
    (IfcMaterialProfileSetUsageTapering))
  SUBTYPE OF (IfcMaterialUsageDefinition);
  ForProfileSet : IfcMaterialProfileSet;
  CardinalPoint : OPTIONAL IfcCardinalPointReference;
  ReferenceExtent : OPTIONAL IfcPositiveLengthMeasure;
END_ENTITY;

ENTITY IfcMaterialProfileSetUsageTapering
  SUBTYPE OF (IfcMaterialProfileSetUsage);
  ForProfileEndSet : IfcMaterialProfileSet;
  CardinalEndPoint : OPTIONAL IfcCardinalPointReference;
END_ENTITY;

ENTITY IfcMaterialProfileWithOffsets
  SUBTYPE OF (IfcMaterialProfile);
  OffsetValues : ARRAY [1:2] OF IfcLengthMeasure;
END_ENTITY;

ENTITY IfcMaterialProperties
  SUBTYPE OF (IfcExtendedProperties);
  Material : IfcMaterialDefinition;
END_ENTITY;

ENTITY IfcMaterialRelationship
  SUBTYPE OF (IfcResourceLevelRelationship);
  RelatingMaterial : IfcMaterial;
  RelatedMaterials : SET [1:?] OF IfcMaterial;
  Expression : OPTIONAL IfcLabel;
END_ENTITY;

ENTITY IfcMaterialUsageDefinition
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcMaterialLayerSetUsage
    , IfcMaterialProfileSetUsage));
  INVERSE
    AssociatedTo : SET [1:?] OF IfcRelAssociatesMaterial FOR RelatingMaterial;
END_ENTITY;

ENTITY IfcMeasureWithUnit;
  ValueComponent : IfcValue;

```

---

```

    UnitComponent : IfcUnit;
END_ENTITY;

ENTITY IfcMechanicalFastener
  SUBTYPE OF (IfcElementComponent);
    NominalDiameter : OPTIONAL IfcPositiveLengthMeasure;
    NominalLength : OPTIONAL IfcPositiveLengthMeasure;
    PredefinedType : OPTIONAL IfcMechanicalFastenerTypeEnum;
  WHERE
    CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
    (PredefinedType <> IfcMechanicalFastenerTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcMechanicalFastenerTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCMECHANICALFASTENERTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcMechanicalFastenerType
  SUBTYPE OF (IfcElementComponentType);
    PredefinedType : IfcMechanicalFastenerTypeEnum;
    NominalDiameter : OPTIONAL IfcPositiveLengthMeasure;
    NominalLength : OPTIONAL IfcPositiveLengthMeasure;
  WHERE
    CorrectPredefinedType : (PredefinedType <>
    IfcMechanicalFastenerTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcMechanicalFastenerTypeEnum.USERDEFINED) AND
    EXISTS (SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcMedicalDevice
  SUBTYPE OF (IfcFlowTerminal);
    PredefinedType : OPTIONAL IfcMedicalDeviceTypeEnum;
  WHERE
    CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
    (PredefinedType <> IfcMedicalDeviceTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcMedicalDeviceTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCMEDICALDEVICETYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcMedicalDeviceType

```

---

```

SUBTYPE OF (IfcFlowTerminalType);
    PredefinedType : IfcMedicalDeviceTypeEnum;
WHERE
    CorrectPredefinedType : (PredefinedType <>
IfcMedicalDeviceTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcMedicalDeviceTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcMember
    SUPERTYPE OF (ONEOF
        (IfcMemberStandardCase))
    SUBTYPE OF (IfcBuildingElementType);
    PredefinedType : OPTIONAL IfcMemberTypeEnum;
WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcMemberTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcMemberTypeEnum.USERDEFINED) AND EXISTS
(SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCMEMBERTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcMemberStandardCase
    SUBTYPE OF (IfcMember);
WHERE
    HasMaterialProfileSetUsage : SIZEOF (QUERY(temp < * USEDIN(SELF,
'IFC4.IFCRELASSOCIATES.RELATEDOBJECTS') |
        ('IFC4.IFCRELASSOCIATESMATERIAL' IN TYPEOF(temp)) AND
        ('IFC4.IFCMATERIALPROFILESETUSAGE' IN
TYPEOF(temp.RelatingMaterial))
    )) = 1;
END_ENTITY;

```

```

ENTITY IfcMemberType
    SUBTYPE OF (IfcBuildingElementType);
    PredefinedType : IfcMemberTypeEnum;
WHERE
    CorrectPredefinedType : (PredefinedType <> IfcMemberTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcMemberTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcMetric

```

---

```

SUBTYPE OF (IfcConstraint);
    Benchmark : IfcBenchmarkEnum;
    ValueSource : OPTIONAL IfcLabel;
    DataValue : OPTIONAL IfcMetricValueSelect;
    ReferencePath : OPTIONAL IfcReference;
END_ENTITY;

ENTITY IfcMirroredProfileDef
    SUBTYPE OF (IfcDerivedProfileDef);
    DERIVE
        SELF\IfcDerivedProfileDef.Operator : IfcCartesianTransformationOperator2D :=
IfcRepresentationItem() || IfcGeometricRepresentationItem() ||
IfcCartesianTransformationOperator(
    -- Axis1
    IfcRepresentationItem() || IfcGeometricRepresentationItem() ||
    IfcDirection([-1., 0.]),
    -- Axis2
    IfcRepresentationItem() || IfcGeometricRepresentationItem() ||
    IfcDirection([ 0., 1.]),
    -- LocalOrigin
    IfcRepresentationItem() || IfcGeometricRepresentationItem() ||
    IfcPoint() || IfcCartesianPoint([0., 0.]),
    -- Scale
    1.) ||
IfcCartesianTransformationOperator2D();
END_ENTITY;

ENTITY IfcMonetaryUnit;
    Currency : IfcLabel;
END_ENTITY;

ENTITY IfcMotorConnection
    SUBTYPE OF (IfcEnergyConversionDevice);
    PredefinedType : OPTIONAL IfcMotorConnectionTypeEnum;
    WHERE
        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
        (PredefinedType <> IfcMotorConnectionTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcMotorConnectionTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject.ObjectType));
        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
        (' IFC4. IFCMOTORCONNECTIONTYPE' IN
        TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

---

```

ENTITY IfcMotorConnectionType
  SUBTYPE OF (IfcEnergyConversionDeviceType);
    PredefinedType : IfcMotorConnectionTypeEnum;
  WHERE
    CorrectPredefinedType          :          (PredefinedType          <>
IfcMotorConnectionTypeEnum.USERDEFINED) OR
((PredefinedType          =          IfcMotorConnectionTypeEnum.USERDEFINED)          AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcNamedUnit
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcContextDependentUnit
    , IfcConversionBasedUnit
    , IfcSIUnit));
    Dimensions : IfcDimensionalExponents;
    UnitType : IfcUnitEnum;
  WHERE
    WR1 : IfcCorrectDimensions (SELF.UnitType, SELF.Dimensions);
END_ENTITY;

```

```

ENTITY IfcObject
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcActor
    , IfcControl
    , IfcGroup
    , IfcProcess
    , IfcProduct
    , IfcResource))
  SUBTYPE OF (IfcObjectDefinition);
    ObjectType : OPTIONAL IfcLabel;
  INVERSE
    IsDeclaredBy : SET [0:1] OF IfcRelDefinesByObject FOR RelatedObjects;
    Declares : SET [0:?] OF IfcRelDefinesByObject FOR RelatingObject;
    IsTypedBy : SET [0:1] OF IfcRelDefinesByType FOR RelatedObjects;
    IsDefinedBy : SET [0:?] OF IfcRelDefinesByProperties FOR RelatedObjects;
  WHERE
    UniquePropertySetNames          :          ((SIZEOF(IsDefinedBy)          =          0)          OR
IfcUniqueDefinitionNames(IsDefinedBy));
END_ENTITY;

```

```

ENTITY IfcObjectDefinition
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcContext

```

---

```

        , IfcObject
        , IfcTypeObject))
SUBTYPE OF (IfcRoot);
INVERSE
    HasAssignments : SET [0:?] OF IfcRelAssigns FOR RelatedObjects;
    Nests : SET [0:1] OF IfcRelNests FOR RelatedObjects;
    IsNestedBy : SET [0:?] OF IfcRelNests FOR RelatingObject;
    HasContext : SET [0:1] OF IfcRelDeclares FOR RelatedDefinitions;
    IsDecomposedBy : SET [0:?] OF IfcRelAggregates FOR RelatingObject;
    Decomposes : SET [0:1] OF IfcRelAggregates FOR RelatedObjects;
    HasAssociations : SET [0:?] OF IfcRelAssociates FOR RelatedObjects;
END_ENTITY;

ENTITY IfcObjectPlacement
ABSTRACT SUPERTYPE OF (ONEOF
    (IfcGridPlacement
    , IfcLocalPlacement));
INVERSE
    PlacesObject : SET [0:?] OF IfcProduct FOR ObjectPlacement;
    ReferencedByPlacements : SET [0:?] OF IfcLocalPlacement FOR PlacementRelTo;
END_ENTITY;

ENTITY IfcObjective
SUBTYPE OF (IfcConstraint);
    BenchmarkValues : OPTIONAL LIST [1:?] OF IfcConstraint;
    LogicalAggregator : OPTIONAL IfcLogicalOperatorEnum;
    ObjectiveQualifier : IfcObjectiveEnum;
    UserDefinedQualifier : OPTIONAL IfcLabel;
WHERE
    WR21 : (ObjectiveQualifier <> IfcObjectiveEnum.USERDEFINED) OR
    ((ObjectiveQualifier = IfcObjectiveEnum.USERDEFINED) AND
    EXISTS(SELf\IfcObjective.UserDefinedQualifier));
END_ENTITY;

ENTITY IfcOccupant
SUBTYPE OF (IfcActor);
    PredefinedType : OPTIONAL IfcOccupantTypeEnum;
WHERE
    WR31 : NOT(PredefinedType = IfcOccupantTypeEnum.USERDEFINED)
OR EXISTS(SELf\IfcObject.ObjectType);
END_ENTITY;

ENTITY IfcOffsetCurve2D
SUBTYPE OF (IfcCurve);

```



---

```

        BasisCurve : IfcCurve;
        Distance : IfcLengthMeasure;
        SelfIntersect : IfcLogical;
    WHERE
        DimIs2D : BasisCurve.Dim = 2;
END_ENTITY;

ENTITY IfcOffsetCurve3D
    SUBTYPE OF (IfcCurve);
        BasisCurve : IfcCurve;
        Distance : IfcLengthMeasure;
        SelfIntersect : IfcLogical;
        RefDirection : IfcDirection;
    WHERE
        DimIs2D : BasisCurve.Dim = 3;
END_ENTITY;

ENTITY IfcOpenShell
    SUBTYPE OF (IfcConnectedFaceSet);
END_ENTITY;

ENTITY IfcOpeningElement
    SUPERTYPE OF (ONEOF
        (IfcOpeningStandardCase))
    SUBTYPE OF (IfcFeatureElementSubtraction);
        PredefinedType : OPTIONAL IfcOpeningElementTypeEnum;
    INVERSE
        HasFillings : SET [0:?] OF IfcRelFillsElement FOR RelatingOpeningElement;
END_ENTITY;

ENTITY IfcOpeningStandardCase
    SUBTYPE OF (IfcOpeningElement);
END_ENTITY;

ENTITY IfcOrganization;
        Identification : OPTIONAL IfcIdentifier;
        Name : IfcLabel;
        Description : OPTIONAL IfcText;
        Roles : OPTIONAL LIST [1:?] OF IfcActorRole;
        Addresses : OPTIONAL LIST [1:?] OF IfcAddress;
    INVERSE
        IsRelatedBy : SET [0:?] OF IfcOrganizationRelationship FOR
RelatedOrganizations;
        Relates : SET [0:?] OF IfcOrganizationRelationship FOR RelatingOrganization;

```

---

```

    Engages : SET [0:?] OF IfcPersonAndOrganization FOR TheOrganization;
END_ENTITY;

ENTITY IfcOrganizationRelationship
  SUBTYPE OF (IfcResourceLevelRelationship);
    RelatingOrganization : IfcOrganization;
    RelatedOrganizations : SET [1:?] OF IfcOrganization;
END_ENTITY;

ENTITY IfcOrientedEdge
  SUBTYPE OF (IfcEdge);
    EdgeElement : IfcEdge;
    Orientation : IfcBoolean;
  DERIVE
    SELF\IfcEdge.EdgeStart : IfcVertex := IfcBooleanChoose
(Orientation, EdgeElement.EdgeStart, EdgeElement.EdgeEnd);
    SELF\IfcEdge.EdgeEnd : IfcVertex := IfcBooleanChoose
(Orientation, EdgeElement.EdgeEnd, EdgeElement.EdgeStart);
  WHERE
    EdgeElementNotOriented : NOT(' IFC4. IFCORIENTEDEDGE' IN TYPEOF (EdgeElement));
END_ENTITY;

ENTITY IfcOuterBoundaryCurve
  SUBTYPE OF (IfcBoundaryCurve);
END_ENTITY;

ENTITY IfcOutlet
  SUBTYPE OF (IfcFlowTerminal);
    PredefinedType : OPTIONAL IfcOutletTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
(PredefinedType <> IfcOutletTypeEnum.USERDEFINED) OR
((PredefinedType = IfcOutletTypeEnum.USERDEFINED) AND EXISTS
(SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
(' IFC4. IFCOUTLETTYPE' IN TYPEOF (SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcOutletType
  SUBTYPE OF (IfcFlowTerminalType);
    PredefinedType : IfcOutletTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcOutletTypeEnum.USERDEFINED) OR
((PredefinedType = IfcOutletTypeEnum.USERDEFINED) AND

```

---

```

EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcOwnerHistory;
    OwningUser : IfcPersonAndOrganization;
    OwningApplication : IfcApplication;
    State : OPTIONAL IfcStateEnum;
    ChangeAction : OPTIONAL IfcChangeActionEnum;
    LastModifiedDate : OPTIONAL IfcTimeStamp;
    LastModifyingUser : OPTIONAL IfcPersonAndOrganization;
    LastModifyingApplication : OPTIONAL IfcApplication;
    CreationDate : IfcTimeStamp;
WHERE
    CorrectChangeAction : (EXISTS(LastModifiedDate)) OR
(NOT (EXISTS(LastModifiedDate)) AND NOT (EXISTS(ChangeAction))) OR
(NOT (EXISTS(LastModifiedDate)) AND EXISTS(ChangeAction) AND ((ChangeAction =
IfcChangeActionEnum.NOTDEFINED) OR (ChangeAction =
IfcChangeActionEnum.NOCHANGE)));
END_ENTITY;

ENTITY IfcParameterizedProfileDef
ABSTRACT SUPERTYPE OF (ONEOF
    (IfcAsymmetricIShapeProfileDef
    , IfcCShapeProfileDef
    , IfcCircleProfileDef
    , IfcEllipseProfileDef
    , IfcIShapeProfileDef
    , IfcLShapeProfileDef
    , IfcRectangleProfileDef
    , IfcTShapeProfileDef
    , IfcTrapeziumProfileDef
    , IfcUShapeProfileDef
    , IfcZShapeProfileDef))
SUBTYPE OF (IfcProfileDef);
    Position : OPTIONAL IfcAxis2Placement2D;
END_ENTITY;

ENTITY IfcPath
SUBTYPE OF (IfcTopologicalRepresentationItem);
    EdgeList : LIST [1:?] OF UNIQUE IfcOrientedEdge;
WHERE
    IsContinuous : IfcPathHeadToTail(SELF);
END_ENTITY;

```

---

```

ENTITY IfcPcurve
  SUBTYPE OF (IfcCurve);
    BasisSurface : IfcSurface;
    ReferenceCurve : IfcCurve;
  WHERE
    DimIs2D : ReferenceCurve.Dim = 2;
END_ENTITY;

ENTITY IfcPerformanceHistory
  SUBTYPE OF (IfcControl);
    LifeCyclePhase : IfcLabel;
    PredefinedType : OPTIONAL IfcPerformanceHistoryTypeEnum;
END_ENTITY;

ENTITY IfcPermeableCoveringProperties
  SUBTYPE OF (IfcPreDefinedPropertySet);
    OperationType : IfcPermeableCoveringOperationEnum;
    PanelPosition : IfcWindowPanelPositionEnum;
    FrameDepth : OPTIONAL IfcPositiveLengthMeasure;
    FrameThickness : OPTIONAL IfcPositiveLengthMeasure;
    ShapeAspectStyle : OPTIONAL IfcShapeAspect;
END_ENTITY;

ENTITY IfcPermit
  SUBTYPE OF (IfcControl);
    PredefinedType : OPTIONAL IfcPermitTypeEnum;
    Status : OPTIONAL IfcLabel;
    LongDescription : OPTIONAL IfcText;
END_ENTITY;

ENTITY IfcPerson;
  Identification : OPTIONAL IfcIdentifier;
  FamilyName : OPTIONAL IfcLabel;
  GivenName : OPTIONAL IfcLabel;
  MiddleNames : OPTIONAL LIST [1:?] OF IfcLabel;
  PrefixTitles : OPTIONAL LIST [1:?] OF IfcLabel;
  SuffixTitles : OPTIONAL LIST [1:?] OF IfcLabel;
  Roles : OPTIONAL LIST [1:?] OF IfcActorRole;
  Addresses : OPTIONAL LIST [1:?] OF IfcAddress;
  INVERSE
    EngagedIn : SET [0:?] OF IfcPersonAndOrganization FOR ThePerson;
  WHERE
    IdentifiablePerson : EXISTS(Identification) OR EXISTS(FamilyName) OR
    EXISTS(GivenName);

```

---

```
ValidSetOfNames : NOT EXISTS(MiddleNames) OR EXISTS(FamilyName) OR  
EXISTS(GivenName);  
END_ENTITY;
```

```
ENTITY IfcPersonAndOrganization;  
  ThePerson : IfcPerson;  
  TheOrganization : IfcOrganization;  
  Roles : OPTIONAL LIST [1:?] OF IfcActorRole;  
END_ENTITY;
```

```
ENTITY IfcPhysicalComplexQuantity  
  SUBTYPE OF (IfcPhysicalQuantity);  
  HasQuantities : SET [1:?] OF IfcPhysicalQuantity;  
  Discrimination : IfcLabel;  
  Quality : OPTIONAL IfcLabel;  
  Usage : OPTIONAL IfcLabel;  
  WHERE  
    NoSelfReference : SIZEOF(QUERY(temp <* HasQuantities | SELF :=: temp)) = 0;  
    UniqueQuantityNames : IfcUniqueQuantityNames(HasQuantities);  
END_ENTITY;
```

```
ENTITY IfcPhysicalQuantity  
  ABSTRACT SUPERTYPE OF (ONEOF  
    (IfcPhysicalComplexQuantity  
    , IfcPhysicalSimpleQuantity));  
  Name : IfcLabel;  
  Description : OPTIONAL IfcText;  
  INVERSE  
    HasExternalReferences : SET [0:?] OF IfcExternalReferenceRelationship FOR  
    RelatedResourceObjects;  
    PartOfComplex : SET [0:1] OF IfcPhysicalComplexQuantity FOR HasQuantities;  
END_ENTITY;
```

```
ENTITY IfcPhysicalSimpleQuantity  
  ABSTRACT SUPERTYPE OF (ONEOF  
    (IfcQuantityArea  
    , IfcQuantityCount  
    , IfcQuantityLength  
    , IfcQuantityTime  
    , IfcQuantityVolume  
    , IfcQuantityWeight))  
  SUBTYPE OF (IfcPhysicalQuantity);  
  Unit : OPTIONAL IfcNamedUnit;  
END_ENTITY;
```

---

```

ENTITY IfcPile
  SUBTYPE OF (IfcBuildingElement);
    PredefinedType : OPTIONAL IfcPileTypeEnum;
    ConstructionType : OPTIONAL IfcPileConstructionEnum;
  WHERE
    CorrectPredefinedType : NOT EXISTS(PredefinedType) OR
    (PredefinedType <> IfcPileTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcPileTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCPILETYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcPileType
  SUBTYPE OF (IfcBuildingElementType);
    PredefinedType : IfcPileTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcPileTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcPileTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcPipeFitting
  SUBTYPE OF (IfcFlowFitting);
    PredefinedType : OPTIONAL IfcPipeFittingTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcPipeFittingTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcPipeFittingTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCPIPEFITTINGTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcPipeFittingType
  SUBTYPE OF (IfcFlowFittingType);
    PredefinedType : IfcPipeFittingTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcPipeFittingTypeEnum.USERDEFINED)
    OR
    ((PredefinedType = IfcPipeFittingTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));

```

---

```

END_ENTITY;

ENTITY IfcPipeSegment
  SUBTYPE OF (IfcFlowSegment);
    PredefinedType : OPTIONAL IfcPipeSegmentTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcPipeSegmentTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcPipeSegmentTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCPIPESEGMENTTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcPipeSegmentType
  SUBTYPE OF (IfcFlowSegmentType);
    PredefinedType : IfcPipeSegmentTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcPipeSegmentTypeEnum.USERDEFINED)
OR
    ((PredefinedType = IfcPipeSegmentTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcPixelTexture
  SUBTYPE OF (IfcSurfaceTexture);
    Width : IfcInteger;
    Height : IfcInteger;
    ColourComponents : IfcInteger;
    Pixel : LIST [1:?] OF IfcBinary;
  WHERE
    MinPixelInS : Width >= 1;
    MinPixelInT : Height >= 1;
    NumberOfColours : {1 <= ColourComponents <= 4};
    SizeOfPixelList : SIZEOF(Pixel) = (Width * Height);
    PixelAsByteAndSameLength : SIZEOF(QUERY(temp<* Pixel |
    (BLENGTH(temp) MOD 8 = 0) AND
    (BLENGTH(temp) = BLENGTH(Pixel[1]))
    )) = SIZEOF(Pixel);
END_ENTITY;

ENTITY IfcPlacement
  ABSTRACT SUPERTYPE OF (ONEOF

```

---

```

        (IfcAxis1Placement
        , IfcAxis2Placement2D
        , IfcAxis2Placement3D))
SUBTYPE OF (IfcGeometricRepresentationItem);
    Location : IfcCartesianPoint;
DERIVE
    Dim : IfcDimensionCount := Location.Dim;
END_ENTITY;

ENTITY IfcPlanarBox
    SUBTYPE OF (IfcPlanarExtent);
        Placement : IfcAxis2Placement;
END_ENTITY;

ENTITY IfcPlanarExtent
    SUPERTYPE OF (ONEOF
        (IfcPlanarBox))
    SUBTYPE OF (IfcGeometricRepresentationItem);
        SizeInX : IfcLengthMeasure;
        SizeInY : IfcLengthMeasure;
END_ENTITY;

ENTITY IfcPlane
    SUBTYPE OF (IfcElementarySurface);
END_ENTITY;

ENTITY IfcPlate
    SUPERTYPE OF (ONEOF
        (IfcPlateStandardCase))
    SUBTYPE OF (IfcBuildingElement);
        PredefinedType : OPTIONAL IfcPlateTypeEnum;
WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcPlateTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcPlateTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCPLATETYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcPlateStandardCase
    SUBTYPE OF (IfcPlate);
WHERE
    HasMaterialLayerSetUsage : SIZEOF (QUERY(temp <* USEDIN(SELF,
```



---

```

' IFC4. IFCRELASSOCIATES.RELATEDOBJECTS' ) |
    (' IFC4. IFCRELASSOCIATESMATERIAL' IN TYPEOF(temp)) AND
    (' IFC4. IFCMATERIALALLAYERSETUSAGE' IN TYPEOF(temp.RelatingMaterial))
)) = 1;
END_ENTITY;

```

```

ENTITY IfcPlateType
  SUBTYPE OF (IfcBuildingElementType);
  PredefinedType : IfcPlateTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcPlateTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcPlateTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcPoint
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcCartesianPoint
    , IfcPointOnCurve
    , IfcPointOnSurface))
  SUBTYPE OF (IfcGeometricRepresentationItem);
END_ENTITY;

```

```

ENTITY IfcPointOnCurve
  SUBTYPE OF (IfcPoint);
  BasisCurve : IfcCurve;
  PointParameter : IfcParameterValue;
  DERIVE
    Dim : IfcDimensionCount := BasisCurve.Dim;
END_ENTITY;

```

```

ENTITY IfcPointOnSurface
  SUBTYPE OF (IfcPoint);
  BasisSurface : IfcSurface;
  PointParameterU : IfcParameterValue;
  PointParameterV : IfcParameterValue;
  DERIVE
    Dim : IfcDimensionCount := BasisSurface.Dim;
END_ENTITY;

```

```

ENTITY IfcPolyLoop
  SUBTYPE OF (IfcLoop);
  Polygon : LIST [3:?] OF UNIQUE IfcCartesianPoint;
  WHERE

```

---

```
    AllPointsSameDim : SIZEOF(QUERY(Temp <* Polygon | Temp.Dim <> Polygon[1].Dim))
= 0;
END_ENTITY;
```

```
ENTITY IfcPolygonalBoundedHalfSpace
SUBTYPE OF (IfcHalfSpaceSolid);
    Position : IfcAxis2Placement3D;
    PolygonalBoundary : IfcBoundedCurve;
WHERE
    BoundaryDim : PolygonalBoundary.Dim = 2;
    BoundaryType : SIZEOF(TYPEOF(PolygonalBoundary) * [
' IFC4. IFCPOLYLINE',
' IFC4. IFCCOMPOSITECURVE' ]
) = 1;
END_ENTITY;
```

```
ENTITY IfcPolygonalFaceSet
SUBTYPE OF (IfcTessellatedFaceSet);
    Closed : OPTIONAL IfcBoolean;
    Faces : LIST [1:?] OF IfcIndexedPolygonalFace;
    PnIndex : OPTIONAL LIST [1:?] OF IfcPositiveInteger;
END_ENTITY;
```

```
ENTITY IfcPolyline
SUBTYPE OF (IfcBoundedCurve);
    Points : LIST [2:?] OF IfcCartesianPoint;
WHERE
    SameDim : SIZEOF(QUERY(Temp <* Points | Temp.Dim <> Points[1].Dim)) = 0;
END_ENTITY;
```

```
ENTITY IfcPort
ABSTRACT SUPERTYPE OF (ONEOF
    (IfcDistributionPort))
SUBTYPE OF (IfcProduct);
INVERSE
    ContainedIn : SET [0:1] OF IfcRelConnectsPortToElement FOR RelatingPort;
    ConnectedFrom : SET [0:1] OF IfcRelConnectsPorts FOR RelatedPort;
    ConnectedTo : SET [0:1] OF IfcRelConnectsPorts FOR RelatingPort;
END_ENTITY;
```

```
ENTITY IfcPostalAddress
SUBTYPE OF (IfcAddress);
    InternalLocation : OPTIONAL IfcLabel;
    AddressLines : OPTIONAL LIST [1:?] OF IfcLabel;
```

---

```

    PostalBox : OPTIONAL IfcLabel;
    Town : OPTIONAL IfcLabel;
    Region : OPTIONAL IfcLabel;
    PostalCode : OPTIONAL IfcLabel;
    Country : OPTIONAL IfcLabel;
WHERE
    WR1 : EXISTS (InternalLocation) OR
EXISTS (AddressLines) OR
EXISTS (PostalBox) OR
EXISTS (PostalCode) OR
EXISTS (Town) OR
EXISTS (Region) OR
EXISTS (Country);
END_ENTITY;

ENTITY IfcPreDefinedColour
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcDraughtingPreDefinedColour))
    SUBTYPE OF (IfcPreDefinedItem);
END_ENTITY;

ENTITY IfcPreDefinedCurveFont
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcDraughtingPreDefinedCurveFont))
    SUBTYPE OF (IfcPreDefinedItem);
END_ENTITY;

ENTITY IfcPreDefinedItem
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcPreDefinedColour
        , IfcPreDefinedCurveFont
        , IfcPreDefinedTextFont))
    SUBTYPE OF (IfcPresentationItem);
    Name : IfcLabel;
END_ENTITY;

ENTITY IfcPreDefinedProperties
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcReinforcementBarProperties
        , IfcSectionProperties
        , IfcSectionReinforcementProperties))
    SUBTYPE OF (IfcPropertyAbstraction);
END_ENTITY;

```

---

```

ENTITY IfcPreDefinedPropertySet
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcDoorLiningProperties
    , IfcDoorPanelProperties
    , IfcPermeableCoveringProperties
    , IfcReinforcementDefinitionProperties
    , IfcWindowLiningProperties
    , IfcWindowPanelProperties))
  SUBTYPE OF (IfcPropertySetDefinition);
END_ENTITY;

ENTITY IfcPreDefinedTextFont
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcTextStyleFontModel))
  SUBTYPE OF (IfcPreDefinedItem);
END_ENTITY;

ENTITY IfcPresentationItem
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcColourRgbList
    , IfcColourSpecification
    , IfcCurveStyleFont
    , IfcCurveStyleFontAndScaling
    , IfcCurveStyleFontPattern
    , IfcIndexedColourMap
    , IfcPreDefinedItem
    , IfcSurfaceStyleLighting
    , IfcSurfaceStyleRefraction
    , IfcSurfaceStyleShading
    , IfcSurfaceStyleWithTextures
    , IfcSurfaceTexture
    , IfcTextStyleForDefinedFont
    , IfcTextStyleTextModel
    , IfcTextureCoordinate
    , IfcTextureVertex
    , IfcTextureVertexList));
END_ENTITY;

ENTITY IfcPresentationLayerAssignment
  SUPERTYPE OF (ONEOF
    (IfcPresentationLayerWithStyle));
  Name : IfcLabel;
  Description : OPTIONAL IfcText;
  AssignedItems : SET [1:?] OF IfcLayeredItem;

```

---

```

        Identifier : OPTIONAL IfcIdentifier;
WHERE
    ApplicableItems : SIZEOF(QUERY(temp <* AssignedItems | (
        SIZEOF(TYPEOF(temp) * [
            'IFC4.IFCSHAPEREPRESENTATION',
            'IFC4.IFCGEOMETRICREPRESENTATIONITEM',
            'IFC4.IFCMAPPEDITEM' ]) = 1)
    )) = SIZEOF(AssignedItems);
END_ENTITY;

ENTITY IfcPresentationLayerWithStyle
    SUBTYPE OF (IfcPresentationLayerAssignment);
        LayerOn : IfcLogical;
        LayerFrozen : IfcLogical;
        LayerBlocked : IfcLogical;
        LayerStyles : SET [0:?] OF IfcPresentationStyle;
WHERE
    ApplicableOnlyToItems : SIZEOF(QUERY(temp <* AssignedItems | (
        SIZEOF(TYPEOF(temp) * [
            'IFC4.IFCGEOMETRICREPRESENTATIONITEM',
            'IFC4.IFCMAPPEDITEM' ]) = 1)
    )) = SIZEOF(AssignedItems);
END_ENTITY;

ENTITY IfcPresentationStyle
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcCurveStyle
        , IfcFillAreaStyle
        , IfcSurfaceStyle
        , IfcTextStyle));
        Name : OPTIONAL IfcLabel;
END_ENTITY;

ENTITY IfcPresentationStyleAssignment;
        Styles : SET [1:?] OF IfcPresentationStyleSelect;
END_ENTITY;

ENTITY IfcProcedure
    SUBTYPE OF (IfcProcess);
        PredefinedType : OPTIONAL IfcProcedureTypeEnum;
WHERE
        HasName : EXISTS(SELF\IfcRoot.Name);
        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR (PredefinedType <>
IfcProcedureTypeEnum.USERDEFINED) OR

```

---

```

((PredefinedType = IfcProcedureTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcObject.ObjectType));
END_ENTITY;

```

```

ENTITY IfcProcedureType
  SUBTYPE OF (IfcTypeProcess);
    PredefinedType : IfcProcedureTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcProcedureTypeEnum.USERDEFINED)
OR    ((PredefinedType = IfcProcedureTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcTypeProcess.ProcessType));
END_ENTITY;

```

```

ENTITY IfcProcess
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcEvent
    , IfcProcedure
    , IfcTask))
  SUBTYPE OF (IfcObject);
    Identification : OPTIONAL IfcIdentifier;
    LongDescription : OPTIONAL IfcText;
  INVERSE
    IsPredecessorTo : SET [0:?] OF IfcRelSequence FOR RelatingProcess;
    IsSuccessorFrom : SET [0:?] OF IfcRelSequence FOR RelatedProcess;
    OperatesOn : SET [0:?] OF IfcRelAssignsToProcess FOR RelatingProcess;
END_ENTITY;

```

```

ENTITY IfcProduct
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcAnnotation
    , IfcElement
    , IfcGrid
    , IfcPort
    , IfcProxy
    , IfcSpatialElement
    , IfcStructuralActivity
    , IfcStructuralItem))
  SUBTYPE OF (IfcObject);
    ObjectPlacement : OPTIONAL IfcObjectPlacement;
    Representation : OPTIONAL IfcProductRepresentation;
  INVERSE
    ReferencedBy : SET [0:?] OF IfcRelAssignsToProduct FOR RelatingProduct;
  WHERE
    PlacementForShapeRepresentation : (EXISTS(Representation) AND

```

---

```

EXISTS(ObjectPlacement))
    OR (EXISTS(Representation) AND
        (SIZEOF(QUERY(temp <* Representation.Representations |
' IFC4. IFCSHAPEREPRESENTATION' IN TYPEOF(temp))) = 0))
    OR (NOT(EXISTS(Representation)));
END_ENTITY;

ENTITY IfcProductDefinitionShape
    SUBTYPE OF (IfcProductRepresentation);
    INVERSE
        ShapeOfProduct : SET [1:?] OF IfcProduct FOR Representation;
        HasShapeAspects : SET [0:?] OF IfcShapeAspect FOR
PartOfProductDefinitionShape;
    WHERE
        OnlyShapeModel : SIZEOF(QUERY(temp <* Representations |
        (NOT(' IFC4. IFCSHAPEMODEL' IN TYPEOF(temp)))
        )) = 0;
END_ENTITY;

ENTITY IfcProductRepresentation
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcMaterialDefinitionRepresentation
        , IfcProductDefinitionShape));
    Name : OPTIONAL IfcLabel;
    Description : OPTIONAL IfcText;
    Representations : LIST [1:?] OF IfcRepresentation;
END_ENTITY;

ENTITY IfcProfileDef
    SUPERTYPE OF (ONEOF
        (IfcArbitraryClosedProfileDef
        , IfcArbitraryOpenProfileDef
        , IfcCompositeProfileDef
        , IfcDerivedProfileDef
        , IfcParameterizedProfileDef));
    ProfileType : IfcProfileTypeEnum;
    ProfileName : OPTIONAL IfcLabel;
    INVERSE
        HasExternalReference : SET [0:?] OF IfcExternalReferenceRelationship FOR
RelatedResourceObjects;
        HasProperties : SET [0:?] OF IfcProfileProperties FOR ProfileDefinition;
END_ENTITY;

ENTITY IfcProfileProperties

```

---

```

    SUBTYPE OF (IfcExtendedProperties);
        ProfileDefinition : IfcProfileDef;
END_ENTITY;

ENTITY IfcProject
    SUBTYPE OF (IfcContext);
    WHERE
        HasName : EXISTS(SELF\IfcRoot.Name);
        CorrectContext : NOT(EXISTS(SELF\IfcContext.RepresentationContexts)) OR
        (SIZEOF(QUERY(Temp <* SELF\IfcContext.RepresentationContexts |
        'IFC4. IFCGEOMETRICREPRESENTATIONSUBCONTEXT' IN TYPEOF(Temp)
        )) = 0);
        NoDecomposition : SIZEOF(SELF\IfcObjectDefinition.Decomposes) = 0;
END_ENTITY;

ENTITY IfcProjectLibrary
    SUBTYPE OF (IfcContext);
END_ENTITY;

ENTITY IfcProjectOrder
    SUBTYPE OF (IfcControl);
        PredefinedType : OPTIONAL IfcProjectOrderTypeEnum;
        Status : OPTIONAL IfcLabel;
        LongDescription : OPTIONAL IfcText;
END_ENTITY;

ENTITY IfcProjectedCRS
    SUBTYPE OF (IfcCoordinateReferenceSystem);
        MapProjection : OPTIONAL IfcIdentifier;
        MapZone : OPTIONAL IfcIdentifier;
        MapUnit : OPTIONAL IfcNamedUnit;
    WHERE
        IsLengthUnit : NOT(EXISTS(MapUnit)) OR (MapUnit.UnitType =
        IfcUnitEnum.LENGTHUNIT);
END_ENTITY;

ENTITY IfcProjectionElement
    SUBTYPE OF (IfcFeatureElementAddition);
        PredefinedType : OPTIONAL IfcProjectionElementTypeEnum;
END_ENTITY;

ENTITY IfcProperty
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcComplexProperty

```



---

```

    , IfcSimpleProperty))
SUBTYPE OF (IfcPropertyAbstraction);
    Name : IfcIdentifier;
    Description : OPTIONAL IfcText;
INVERSE
    PartOfPset : SET [0:?] OF IfcPropertySet FOR HasProperties;
    PropertyForDependence : SET [0:?] OF IfcPropertyDependencyRelationship FOR
DependingProperty;
    PropertyDependsOn : SET [0:?] OF IfcPropertyDependencyRelationship FOR
DependantProperty;
    PartOfComplex : SET [0:?] OF IfcComplexProperty FOR HasProperties;
    HasConstraints : SET [0:?] OF IfcResourceConstraintRelationship FOR
RelatedResourceObjects;
    HasApprovals : SET [0:?] OF IfcResourceApprovalRelationship FOR
RelatedResourceObjects;
END_ENTITY;

ENTITY IfcPropertyAbstraction
ABSTRACT SUPERTYPE OF (ONEOF
    (IfcExtendedProperties
    , IfcPreDefinedProperties
    , IfcProperty
    , IfcPropertyEnumeration));
INVERSE
    HasExternalReferences : SET [0:?] OF IfcExternalReferenceRelationship FOR
RelatedResourceObjects;
END_ENTITY;

ENTITY IfcPropertyBoundedValue
SUBTYPE OF (IfcSimpleProperty);
    UpperBoundValue : OPTIONAL IfcValue;
    LowerBoundValue : OPTIONAL IfcValue;
    Unit : OPTIONAL IfcUnit;
    SetPointValue : OPTIONAL IfcValue;
WHERE
    SameUnitUpperLower : NOT (EXISTS (UpperBoundValue)) OR
NOT (EXISTS (LowerBoundValue)) OR
(TYPEOF (UpperBoundValue) = TYPEOF (LowerBoundValue));
    SameUnitUpperSet : NOT (EXISTS (UpperBoundValue)) OR NOT (EXISTS (SetPointValue))
OR
(TYPEOF (UpperBoundValue) = TYPEOF (SetPointValue));
    SameUnitLowerSet : NOT (EXISTS (LowerBoundValue)) OR NOT (EXISTS (SetPointValue))
OR
(TYPEOF (LowerBoundValue) = TYPEOF (SetPointValue));

```

---

```

END_ENTITY;

ENTITY IfcPropertyDefinition
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcPropertySetDefinition
      , IfcPropertyTemplateDefinition))
  SUBTYPE OF (IfcRoot);
  INVERSE
    HasContext : SET [0:1] OF IfcRelDeclares FOR RelatedDefinitions;
    HasAssociations : SET [0:?] OF IfcRelAssociates FOR RelatedObjects;
END_ENTITY;

ENTITY IfcPropertyDependencyRelationship
  SUBTYPE OF (IfcResourceLevelRelationship);
  DependingProperty : IfcProperty;
  DependantProperty : IfcProperty;
  Expression : OPTIONAL IfcText;
  WHERE
    NoSelfReference : DependingProperty :<>: DependantProperty;
END_ENTITY;

ENTITY IfcPropertyEnumeratedValue
  SUBTYPE OF (IfcSimpleProperty);
  EnumerationValues : OPTIONAL LIST [1:?] OF IfcValue;
  EnumerationReference : OPTIONAL IfcPropertyEnumeration;
  WHERE
    WR21 : NOT(EXISTS(EnumerationReference))
OR NOT(EXISTS(EnumerationValues))
OR (SIZEOF(QUERY(temp <* EnumerationValues |
  temp IN EnumerationReference.EnumerationValues))
  = SIZEOF(EnumerationValues));
END_ENTITY;

ENTITY IfcPropertyEnumeration
  SUBTYPE OF (IfcPropertyAbstraction);
  Name : IfcLabel;
  EnumerationValues : LIST [1:?] OF UNIQUE IfcValue;
  Unit : OPTIONAL IfcUnit;
  UNIQUE
    UR1 : Name;
  WHERE
    WR01 : SIZEOF(QUERY(temp <* SELF.EnumerationValues |
  NOT(TYPEOF(SELF.EnumerationValues[1]) = TYPEOF(temp))
  )) = 0;

```

---

```

END_ENTITY;

ENTITY IfcPropertyListValue
  SUBTYPE OF (IfcSimpleProperty);
    ListValues : OPTIONAL LIST [1:?] OF IfcValue;
    Unit : OPTIONAL IfcUnit;
  WHERE
    WR31 : SIZEOF(QUERY(temp <* SELF.ListValues |
      NOT(TYPEOF(SELF.ListValues[1]) = TYPEOF(temp))
    )) = 0;
END_ENTITY;

ENTITY IfcPropertyReferenceValue
  SUBTYPE OF (IfcSimpleProperty);
    UsageName : OPTIONAL IfcText;
    PropertyReference : OPTIONAL IfcObjectReferenceSelect;
END_ENTITY;

ENTITY IfcPropertySet
  SUBTYPE OF (IfcPropertySetDefinition);
    HasProperties : SET [1:?] OF IfcProperty;
  WHERE
    ExistsName : EXISTS(SELF\IfcRoot.Name);
    UniquePropertyNames : IfcUniquePropertyName(HasProperties);
END_ENTITY;

ENTITY IfcPropertySetDefinition
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcPreDefinedPropertySet
    , IfcPropertySet
    , IfcQuantitySet))
  SUBTYPE OF (IfcPropertyDefinition);
  INVERSE
    DefinesType : SET [0:?] OF IfcTypeObject FOR HasPropertySets;
    IsDefinedBy : SET [0:?] OF IfcRelDefinesByTemplate FOR RelatedPropertySets;
    DefinesOccurrence : SET [0:?] OF IfcRelDefinesByProperties FOR
    RelatingPropertyDefinition;
END_ENTITY;

ENTITY IfcPropertySetTemplate
  SUBTYPE OF (IfcPropertyTemplateDefinition);
    TemplateType : OPTIONAL IfcPropertySetTemplateTypeEnum;
    ApplicableEntity : OPTIONAL IfcIdentifier;
    HasPropertyTemplates : SET [1:?] OF IfcPropertyTemplate;

```

---

```

INVERSE
    Defines : SET [0:?] OF IfcRelDefinesByTemplate FOR RelatingTemplate;
WHERE
    ExistsName : EXISTS(SELF\IfcRoot.Name);
    UniquePropertyNames : IfcUniquePropertyTemplateNameNames(HasPropertyTemplates);
END_ENTITY;

ENTITY IfcPropertySingleValue
    SUBTYPE OF (IfcSimpleProperty);
    NominalValue : OPTIONAL IfcValue;
    Unit : OPTIONAL IfcUnit;
END_ENTITY;

ENTITY IfcPropertyTableValue
    SUBTYPE OF (IfcSimpleProperty);
    DefiningValues : OPTIONAL LIST [1:?] OF UNIQUE IfcValue;
    DefinedValues : OPTIONAL LIST [1:?] OF IfcValue;
    Expression : OPTIONAL IfcText;
    DefiningUnit : OPTIONAL IfcUnit;
    DefinedUnit : OPTIONAL IfcUnit;
    CurveInterpolation : OPTIONAL IfcCurveInterpolationEnum;
WHERE
    WR21 : (NOT(EXISTS(DefiningValues)) AND NOT(EXISTS(DefinedValues)))
OR (SIZEOF(DefiningValues) = SIZEOF(DefinedValues));
    WR22 : NOT(EXISTS(DefiningValues)) OR
(SIZEOF(QUERY(temp < * SELF.DefiningValues | TYPEOF(temp) <>
TYPEOF(SELF.DefiningValues[1])
)) = 0);
    WR23 : NOT(EXISTS(DefinedValues)) OR
(SIZEOF(QUERY(temp < * SELF.DefinedValues | TYPEOF(temp) <>
TYPEOF(SELF.DefinedValues[1])
)) = 0);
END_ENTITY;

ENTITY IfcPropertyTemplate
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcComplexPropertyTemplate
        , IfcSimplePropertyTemplate))
    SUBTYPE OF (IfcPropertyTemplateDefinition);
    INVERSE
        PartOfComplexTemplate : SET [0:?] OF IfcComplexPropertyTemplate FOR
HasPropertyTemplates;
        PartOfPsetTemplate : SET [0:?] OF IfcPropertySetTemplate FOR
HasPropertyTemplates;

```

---

```

END_ENTITY;

ENTITY IfcPropertyTemplateDefinition
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcPropertySetTemplate
      , IfcPropertyTemplate))
  SUBTYPE OF (IfcPropertyDefinition);
END_ENTITY;

ENTITY IfcProtectiveDevice
  SUBTYPE OF (IfcFlowController);
  PredefinedType : OPTIONAL IfcProtectiveDeviceTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcProtectiveDeviceTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcProtectiveDeviceTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    (' IFC4. IFCPROTECTIVEDEVICETYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcProtectiveDeviceTrippingUnit
  SUBTYPE OF (IfcDistributionControlElement);
  PredefinedType : OPTIONAL IfcProtectiveDeviceTrippingUnitTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcProtectiveDeviceTrippingUnitTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcProtectiveDeviceTrippingUnitTypeEnum.USERDEFINED) AND
    EXISTS (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    (' IFC4. IFCPROTECTIVEDEVICETRIPPINGUNITTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcProtectiveDeviceTrippingUnitType
  SUBTYPE OF (IfcDistributionControlElementType);
  PredefinedType : IfcProtectiveDeviceTrippingUnitTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <>
    IfcProtectiveDeviceTrippingUnitTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcProtectiveDeviceTrippingUnitTypeEnum.USERDEFINED) AND
    EXISTS (SELF\IfcElementType.ElementType));
END_ENTITY;

```

---

```

ENTITY IfcProtectiveDeviceType
  SUBTYPE OF (IfcFlowControllerType);
    PredefinedType : IfcProtectiveDeviceTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <>
IfcProtectiveDeviceTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcProtectiveDeviceTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcProxy
  SUBTYPE OF (IfcProduct);
    ProxyType : IfcObjectTypeEnum;
    Tag : OPTIONAL IfcLabel;
  WHERE
    WR1 : EXISTS(SELF\IfcRoot.Name);
END_ENTITY;

```

```

ENTITY IfcPump
  SUBTYPE OF (IfcFlowMovingDevice);
    PredefinedType : OPTIONAL IfcPumpTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcPumpTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcPumpTypeEnum.USERDEFINED) AND EXISTS
(SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCPUMPTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcPumpType
  SUBTYPE OF (IfcFlowMovingDeviceType);
    PredefinedType : IfcPumpTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcPumpTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcPumpTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcQuantityArea
  SUBTYPE OF (IfcPhysicalSimpleQuantity);
    AreaValue : IfcAreaMeasure;
    Formula : OPTIONAL IfcLabel;

```

---

```

WHERE
    WR21 : NOT(EXISTS(SELF\IfcPhysicalSimpleQuantity.Unit)) OR
    (SELF\IfcPhysicalSimpleQuantity.Unit.UnitType = IfcUnitEnum.AREAUNIT);
    WR22 : AreaValue >= 0.;
END_ENTITY;

ENTITY IfcQuantityCount
    SUBTYPE OF (IfcPhysicalSimpleQuantity);
        CountValue : IfcCountMeasure;
        Formula : OPTIONAL IfcLabel;
    WHERE
        WR21 : CountValue >= 0.;
    END_ENTITY;

ENTITY IfcQuantityLength
    SUBTYPE OF (IfcPhysicalSimpleQuantity);
        LengthValue : IfcLengthMeasure;
        Formula : OPTIONAL IfcLabel;
    WHERE
        WR21 : NOT(EXISTS(SELF\IfcPhysicalSimpleQuantity.Unit)) OR
        (SELF\IfcPhysicalSimpleQuantity.Unit.UnitType = IfcUnitEnum.LENGTHUNIT);
        WR22 : LengthValue >= 0.;
    END_ENTITY;

ENTITY IfcQuantitySet
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcElementQuantity))
    SUBTYPE OF (IfcPropertySetDefinition);
END_ENTITY;

ENTITY IfcQuantityTime
    SUBTYPE OF (IfcPhysicalSimpleQuantity);
        TimeValue : IfcTimeMeasure;
        Formula : OPTIONAL IfcLabel;
    WHERE
        WR21 : NOT(EXISTS(SELF\IfcPhysicalSimpleQuantity.Unit)) OR
        (SELF\IfcPhysicalSimpleQuantity.Unit.UnitType = IfcUnitEnum.TIMEUNIT);
        WR22 : TimeValue >= 0.;
    END_ENTITY;

ENTITY IfcQuantityVolume
    SUBTYPE OF (IfcPhysicalSimpleQuantity);
        VolumeValue : IfcVolumeMeasure;
        Formula : OPTIONAL IfcLabel;

```

---

```

WHERE
    WR21 : NOT(EXISTS(SELF\IfcPhysicalSimpleQuantity.Unit)) OR
    (SELF\IfcPhysicalSimpleQuantity.Unit.UnitType = IfcUnitEnum.VOLUMEUNIT);
    WR22 : VolumeValue >= 0.;
END_ENTITY;

ENTITY IfcQuantityWeight
    SUBTYPE OF (IfcPhysicalSimpleQuantity);
    WeightValue : IfcMassMeasure;
    Formula : OPTIONAL IfcLabel;
    WHERE
        WR21 : NOT(EXISTS(SELF\IfcPhysicalSimpleQuantity.Unit)) OR
        (SELF\IfcPhysicalSimpleQuantity.Unit.UnitType = IfcUnitEnum.MASSUNIT);
        WR22 : WeightValue >= 0.;
    END_ENTITY;

ENTITY IfcRailing
    SUBTYPE OF (IfcBuildingElement);
    PredefinedType : OPTIONAL IfcRailingTypeEnum;
    WHERE
        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
        (PredefinedType <> IfcRailingTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcRailingTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject.ObjectType));
        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
        ('IFC4.IFCRAILINGTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
    END_ENTITY;

ENTITY IfcRailingType
    SUBTYPE OF (IfcBuildingElementType);
    PredefinedType : IfcRailingTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <> IfcRailingTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcRailingTypeEnum.USERDEFINED) AND
        EXISTS(SELF\IfcElementType.ElementType));
    END_ENTITY;

ENTITY IfcRamp
    SUBTYPE OF (IfcBuildingElement);
    PredefinedType : OPTIONAL IfcRampTypeEnum;
    WHERE
        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
        (PredefinedType <> IfcRampTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcRampTypeEnum.USERDEFINED) AND EXISTS

```



---

```

(SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCRAMPTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcRampFlight
    SUBTYPE OF (IfcBuildingElement);
    PredefinedType : OPTIONAL IfcRampFlightTypeEnum;
    WHERE
        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
        (PredefinedType <> IfcRampFlightTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcRampFlightTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject.ObjectType));
        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
        ('IFC4.IFCRAMPFLIGHTTYPE' IN
        TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcRampFlightType
    SUBTYPE OF (IfcBuildingElementType);
    PredefinedType : IfcRampFlightTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <> IfcRampFlightTypeEnum.USERDEFINED)
    OR
    ((PredefinedType = IfcRampFlightTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcRampType
    SUBTYPE OF (IfcBuildingElementType);
    PredefinedType : IfcRampTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <> IfcRampTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcRampTypeEnum.USERDEFINED) AND
        EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcRationalBSplineCurveWithKnots
    SUBTYPE OF (IfcBSplineCurveWithKnots);
    WeightsData : LIST [2:?] OF IfcReal;
    DERIVE
        Weights : ARRAY [0:UpperIndexOnControlPoints] OF IfcReal :=
        IfcListToArray(WeightsData, 0, SELF\IfcBSplineCurve.UpperIndexOnControlPoints);
    WHERE

```

---

```

        SameNumOfWeightsAndPoints          :          SIZEOF(WeightsData)          =
SIZEOF(SELF\IfcBSplineCurve.ControlPointsList);
        WeightsGreaterZero : IfcCurveWeightsPositive(SELF);
END_ENTITY;

ENTITY IfcRationalBSplineSurfaceWithKnots
    SUBTYPE OF (IfcBSplineSurfaceWithKnots);
        WeightsData : LIST [2:?] OF LIST [2:?] OF IfcReal;
    DERIVE
        Weights : ARRAY [0:UUpper] OF ARRAY [0:VUpper] OF IfcReal :=
IfcMakeArrayOfArray(WeightsData, 0, UUpper, 0, VUpper);
    WHERE
        CorrespondingWeightsDataLists          :          (SIZEOF(WeightsData)          =
SIZEOF(SELF\IfcBSplineSurface.ControlPointsList))
    AND
        (SIZEOF(WeightsData[1]) = SIZEOF(SELF\IfcBSplineSurface.ControlPointsList[1]));
        WeightValuesGreaterZero : IfcSurfaceWeightsPositive(SELF);
END_ENTITY;

ENTITY IfcRectangleHollowProfileDef
    SUBTYPE OF (IfcRectangleProfileDef);
        WallThickness : IfcPositiveLengthMeasure;
        InnerFilletRadius : OPTIONAL IfcNonNegativeLengthMeasure;
        OuterFilletRadius : OPTIONAL IfcNonNegativeLengthMeasure;
    WHERE
        ValidWallThickness : (WallThickness < (SELF\IfcRectangleProfileDef.XDim/2.))
    AND
        (WallThickness < (SELF\IfcRectangleProfileDef.YDim/2.));
        ValidInnerRadius : NOT(EXISTS(InnerFilletRadius)) OR
        ((InnerFilletRadius <= (SELF\IfcRectangleProfileDef.XDim/2. - WallThickness)) AND
        (InnerFilletRadius <= (SELF\IfcRectangleProfileDef.YDim/2. - WallThickness)));
        ValidOuterRadius : NOT(EXISTS(OuterFilletRadius)) OR
        ((OuterFilletRadius <= (SELF\IfcRectangleProfileDef.XDim/2.)) AND
        (OuterFilletRadius <= (SELF\IfcRectangleProfileDef.YDim/2.)));
END_ENTITY;

ENTITY IfcRectangleProfileDef
    SUPERTYPE OF (ONEOF
        (IfcRectangleHollowProfileDef
        , IfcRoundedRectangleProfileDef))
    SUBTYPE OF (IfcParameterizedProfileDef);
        XDim : IfcPositiveLengthMeasure;
        YDim : IfcPositiveLengthMeasure;
END_ENTITY;

```

---

```

ENTITY IfcRectangularPyramid
  SUBTYPE OF (IfcCsgPrimitive3D);
    XLength : IfcPositiveLengthMeasure;
    YLength : IfcPositiveLengthMeasure;
    Height : IfcPositiveLengthMeasure;
END_ENTITY;

ENTITY IfcRectangularTrimmedSurface
  SUBTYPE OF (IfcBoundedSurface);
    BasisSurface : IfcSurface;
    U1 : IfcParameterValue;
    V1 : IfcParameterValue;
    U2 : IfcParameterValue;
    V2 : IfcParameterValue;
    Usense : IfcBoolean;
    Vsense : IfcBoolean;
  WHERE
    U1AndU2Different : U1 <> U2;
    V1AndV2Different : V1 <> V2;
    UseCompatible : ((' IFC4. IFCELEMENTARYSURFACE' IN TYPEOF(BasisSurface)) AND
      (NOT (' IFC4. IFCPLANE' IN TYPEOF(BasisSurface)))) OR
      (' IFC4. IFCSURFACEOFREVOLUTION' IN TYPEOF(BasisSurface)) OR
      (Usense = (U2 > U1));
    VsenseCompatible : Vsense = (V2 > V1);
END_ENTITY;

ENTITY IfcRecurrencePattern;
  RecurrenceType : IfcRecurrenceTypeEnum;
  DayComponent : OPTIONAL SET [1:?] OF IfcDayInMonthNumber;
  WeekdayComponent : OPTIONAL SET [1:?] OF IfcDayInWeekNumber;
  MonthComponent : OPTIONAL SET [1:?] OF IfcMonthInYearNumber;
  Position : OPTIONAL IfcInteger;
  Interval : OPTIONAL IfcInteger;
  Occurrences : OPTIONAL IfcInteger;
  TimePeriods : OPTIONAL LIST [1:?] OF IfcTimePeriod;
END_ENTITY;

ENTITY IfcReference;
  TypeIdentifier : OPTIONAL IfcIdentifier;
  AttributeIdentifier : OPTIONAL IfcIdentifier;
  InstanceName : OPTIONAL IfcLabel;
  ListPositions : OPTIONAL LIST [1:?] OF IfcInteger;
  InnerReference : OPTIONAL IfcReference;

```

---

```

END_ENTITY;

ENTITY IfcRegularTimeSeries
  SUBTYPE OF (IfcTimeSeries);
    TimeStep : IfcTimeMeasure;
    Values : LIST [1:?] OF IfcTimeSeriesValue;
END_ENTITY;

ENTITY IfcReinforcementBarProperties
  SUBTYPE OF (IfcPreDefinedProperties);
    TotalCrossSectionArea : IfcAreaMeasure;
    SteelGrade : IfcLabel;
    BarSurface : OPTIONAL IfcReinforcingBarSurfaceEnum;
    EffectiveDepth : OPTIONAL IfcLengthMeasure;
    NominalBarDiameter : OPTIONAL IfcPositiveLengthMeasure;
    BarCount : OPTIONAL IfcCountMeasure;
END_ENTITY;

ENTITY IfcReinforcementDefinitionProperties
  SUBTYPE OF (IfcPreDefinedPropertySet);
    DefinitionType : OPTIONAL IfcLabel;
    ReinforcementSectionDefinitions : LIST [1:?] OF
IfcSectionReinforcementProperties;
END_ENTITY;

ENTITY IfcReinforcingBar
  SUBTYPE OF (IfcReinforcingElement);
    NominalDiameter : OPTIONAL IfcPositiveLengthMeasure;
    CrossSectionArea : OPTIONAL IfcAreaMeasure;
    BarLength : OPTIONAL IfcPositiveLengthMeasure;
    PredefinedType : OPTIONAL IfcReinforcingBarTypeEnum;
    BarSurface : OPTIONAL IfcReinforcingBarSurfaceEnum;
  WHERE
    CorrectPredefinedType : NOT EXISTS(PredefinedType) OR
    (PredefinedType <> IfcReinforcingBarTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcReinforcingBarTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCREINFORCINGBARTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcReinforcingBarType
  SUBTYPE OF (IfcReinforcingElementType);

```

---

```

    PredefinedType : IfcReinforcingBarTypeEnum;
    NominalDiameter : OPTIONAL IfcPositiveLengthMeasure;
    CrossSectionArea : OPTIONAL IfcAreaMeasure;
    BarLength : OPTIONAL IfcPositiveLengthMeasure;
    BarSurface : OPTIONAL IfcReinforcingBarSurfaceEnum;
    BendingShapeCode : OPTIONAL IfcLabel;
    BendingParameters : OPTIONAL LIST [1:?] OF IfcBendingParameterSelect;
WHERE
    CorrectPredefinedType          :          (PredefinedType          <>
IfcReinforcingBarTypeEnum.USERDEFINED) OR
((PredefinedType          =          IfcReinforcingBarTypeEnum.USERDEFINED)          AND
EXISTS(SELF\IfcElementType.ElementType));
    BendingShapeCodeProvided      :      NOT      EXISTS(BendingParameters)      OR
EXISTS(BendingShapeCode);
END_ENTITY;

```

```

ENTITY IfcReinforcingElement
ABSTRACT SUPERTYPE OF (ONEOF
    (IfcReinforcingBar
    , IfcReinforcingMesh
    , IfcTendon
    , IfcTendonAnchor))
SUBTYPE OF (IfcElementComponent);
    SteelGrade : OPTIONAL IfcLabel;
END_ENTITY;

```

```

ENTITY IfcReinforcingElementType
ABSTRACT SUPERTYPE OF (ONEOF
    (IfcReinforcingBarType
    , IfcReinforcingMeshType
    , IfcTendonAnchorType
    , IfcTendonType))
SUBTYPE OF (IfcElementComponentType);
END_ENTITY;

```

```

ENTITY IfcReinforcingMesh
SUBTYPE OF (IfcReinforcingElement);
    MeshLength : OPTIONAL IfcPositiveLengthMeasure;
    MeshWidth : OPTIONAL IfcPositiveLengthMeasure;
    LongitudinalBarNominalDiameter : OPTIONAL IfcPositiveLengthMeasure;
    TransverseBarNominalDiameter : OPTIONAL IfcPositiveLengthMeasure;
    LongitudinalBarCrossSectionArea : OPTIONAL IfcAreaMeasure;
    TransverseBarCrossSectionArea : OPTIONAL IfcAreaMeasure;
    LongitudinalBarSpacing : OPTIONAL IfcPositiveLengthMeasure;

```

---

```

    TransverseBarSpacing : OPTIONAL IfcPositiveLengthMeasure;
    PredefinedType : OPTIONAL IfcReinforcingMeshTypeEnum;
    WHERE
        CorrectPredefinedType : NOT EXISTS(PredefinedType) OR
        (PredefinedType <> IfcReinforcingMeshTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcReinforcingMeshTypeEnum.USERDEFINED) AND
        EXISTS(SELF\IfcObject.ObjectType));
        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
        ('IFC4.IFCREINFORCINGMESHTYPE' IN
        TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
    END_ENTITY;

```

```

ENTITY IfcReinforcingMeshType

```

```

    SUBTYPE OF (IfcReinforcingElementType);
    PredefinedType : IfcReinforcingMeshTypeEnum;
    MeshLength : OPTIONAL IfcPositiveLengthMeasure;
    MeshWidth : OPTIONAL IfcPositiveLengthMeasure;
    LongitudinalBarNominalDiameter : OPTIONAL IfcPositiveLengthMeasure;
    TransverseBarNominalDiameter : OPTIONAL IfcPositiveLengthMeasure;
    LongitudinalBarCrossSectionArea : OPTIONAL IfcAreaMeasure;
    TransverseBarCrossSectionArea : OPTIONAL IfcAreaMeasure;
    LongitudinalBarSpacing : OPTIONAL IfcPositiveLengthMeasure;
    TransverseBarSpacing : OPTIONAL IfcPositiveLengthMeasure;
    BendingShapeCode : OPTIONAL IfcLabel;
    BendingParameters : OPTIONAL LIST [1:?] OF IfcBendingParameterSelect;
    WHERE
        CorrectPredefinedType : (PredefinedType <>
        IfcReinforcingMeshTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcReinforcingMeshTypeEnum.USERDEFINED) AND
        EXISTS(SELF\IfcElementType.ElementType));
        BendingShapeCodeProvided : NOT EXISTS(BendingParameters) OR
        EXISTS(BendingShapeCode);
    END_ENTITY;

```

```

ENTITY IfcRelAggregates

```

```

    SUBTYPE OF (IfcRelDecomposes);
    RelatingObject : IfcObjectDefinition;
    RelatedObjects : SET [1:?] OF IfcObjectDefinition;
    WHERE
        NoSelfReference : SIZEOF(QUERY(Temp < * RelatedObjects | RelatingObject :=:
        Temp)) = 0;
    END_ENTITY;

```

```

ENTITY IfcRelAssigns

```

---

```

ABSTRACT SUPERTYPE OF (ONEOF
    (IfcRelAssignsToActor
    , IfcRelAssignsToControl
    , IfcRelAssignsToGroup
    , IfcRelAssignsToProcess
    , IfcRelAssignsToProduct
    , IfcRelAssignsToResource))
SUBTYPE OF (IfcRelationship);
    RelatedObjects : SET [1:?] OF IfcObjectDefinition;
    RelatedObjectsType : OPTIONAL IfcObjectTypeEnum;
WHERE
    WR1 : IfcCorrectObjectAssignment(RelatedObjectsType, RelatedObjects);
END_ENTITY;

ENTITY IfcRelAssignsToActor
    SUBTYPE OF (IfcRelAssigns);
        RelatingActor : IfcActor;
        ActingRole : OPTIONAL IfcActorRole;
WHERE
    NoSelfReference : SIZEOF(QUERY(Temp <* SELF\IfcRelAssigns.RelatedObjects |
RelatingActor :=: Temp)) = 0;
END_ENTITY;

ENTITY IfcRelAssignsToControl
    SUBTYPE OF (IfcRelAssigns);
        RelatingControl : IfcControl;
WHERE
    NoSelfReference : SIZEOF(QUERY(Temp <* SELF\IfcRelAssigns.RelatedObjects |
RelatingControl :=: Temp)) = 0;
END_ENTITY;

ENTITY IfcRelAssignsToGroup
    SUPERTYPE OF (ONEOF
        (IfcRelAssignsToGroupByFactor))
    SUBTYPE OF (IfcRelAssigns);
        RelatingGroup : IfcGroup;
WHERE
    NoSelfReference : SIZEOF(QUERY(Temp <* SELF\IfcRelAssigns.RelatedObjects |
RelatingGroup :=: Temp)) = 0;
END_ENTITY;

ENTITY IfcRelAssignsToGroupByFactor
    SUBTYPE OF (IfcRelAssignsToGroup);
        Factor : IfcRatioMeasure;

```

---

```

END_ENTITY;

ENTITY IfcRelAssignsToProcess
  SUBTYPE OF (IfcRelAssigns);
    RelatingProcess : IfcProcessSelect;
    QuantityInProcess : OPTIONAL IfcMeasureWithUnit;
  WHERE
    NoSelfReference : SIZEOF(QUERY(Temp <* SELF\IfcRelAssigns.RelatedObjects |
RelatingProcess :=: Temp)) = 0;
END_ENTITY;

ENTITY IfcRelAssignsToProduct
  SUBTYPE OF (IfcRelAssigns);
    RelatingProduct : IfcProductSelect;
  WHERE
    NoSelfReference : SIZEOF(QUERY(Temp <* SELF\IfcRelAssigns.RelatedObjects |
RelatingProduct :=: Temp)) = 0;
END_ENTITY;

ENTITY IfcRelAssignsToResource
  SUBTYPE OF (IfcRelAssigns);
    RelatingResource : IfcResourceSelect;
  WHERE
    NoSelfReference : SIZEOF(QUERY(Temp <* SELF\IfcRelAssigns.RelatedObjects |
RelatingResource :=: Temp)) = 0;
END_ENTITY;

ENTITY IfcRelAssociates
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcRelAssociatesApproval
    , IfcRelAssociatesClassification
    , IfcRelAssociatesConstraint
    , IfcRelAssociatesDocument
    , IfcRelAssociatesLibrary
    , IfcRelAssociatesMaterial))
  SUBTYPE OF (IfcRelationship);
    RelatedObjects : SET [1:?] OF IfcDefinitionSelect;
END_ENTITY;

ENTITY IfcRelAssociatesApproval
  SUBTYPE OF (IfcRelAssociates);
    RelatingApproval : IfcApproval;
END_ENTITY;

```



---

```

ENTITY IfcRelAssociatesClassification
  SUBTYPE OF (IfcRelAssociates);
    RelatingClassification : IfcClassificationSelect;
END_ENTITY;

ENTITY IfcRelAssociatesConstraint
  SUBTYPE OF (IfcRelAssociates);
    Intent : OPTIONAL IfcLabel;
    RelatingConstraint : IfcConstraint;
END_ENTITY;

ENTITY IfcRelAssociatesDocument
  SUBTYPE OF (IfcRelAssociates);
    RelatingDocument : IfcDocumentSelect;
END_ENTITY;

ENTITY IfcRelAssociatesLibrary
  SUBTYPE OF (IfcRelAssociates);
    RelatingLibrary : IfcLibrarySelect;
END_ENTITY;

ENTITY IfcRelAssociatesMaterial
  SUBTYPE OF (IfcRelAssociates);
    RelatingMaterial : IfcMaterialSelect;
  WHERE
    NoVoidElement : SIZEOF(QUERY(temp <* SELF\IfcRelAssociates.RelatedObjects |
      (' IFC4. IFCFEATUREELEMENTSUBTRACTION' IN TYPEOF(temp)) OR
      (' IFC4. IFCVIRTUALELEMENT' IN TYPEOF(temp))
    )) = 0;
    AllowedElements : SIZEOF(QUERY(temp <* SELF\IfcRelAssociates.RelatedObjects |
      (
        SIZEOF(TYPEOF(temp) * [
          ' IFC4. IFCELEMENT',
          ' IFC4. IFCELEMENTTYPE',
          ' IFC4. IFCWINDOWSTYLE',
          ' IFC4. IFCDOORSTYLE',
          ' IFC4. IFCSTRUCTURALMEMBER',
          ' IFC4. IFCPORT' ]) = 0)
    )) = 0;
END_ENTITY;

ENTITY IfcRelConnects
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcRelConnectsElements

```

---

```

    , IfcRelConnectsPortToElement
    , IfcRelConnectsPorts
    , IfcRelConnectsStructuralActivity
    , IfcRelConnectsStructuralMember
    , IfcRelContainedInSpatialStructure
    , IfcRelCoversBldgElements
    , IfcRelCoversSpaces
    , IfcRelFillsElement
    , IfcRelFlowControlElements
    , IfcRelInterferesElements
    , IfcRelReferencedInSpatialStructure
    , IfcRelSequence
    , IfcRelServicesBuildings
    , IfcRelSpaceBoundary))
SUBTYPE OF (IfcRelationship);
END_ENTITY;

ENTITY IfcRelConnectsElements
SUPERTYPE OF (ONEOF
    (IfcRelConnectsPathElements
    , IfcRelConnectsWithRealizingElements))
SUBTYPE OF (IfcRelConnects);
    ConnectionGeometry : OPTIONAL IfcConnectionGeometry;
    RelatingElement : IfcElement;
    RelatedElement : IfcElement;
WHERE
    NoSelfReference : RelatingElement :<>: RelatedElement;
END_ENTITY;

ENTITY IfcRelConnectsPathElements
SUBTYPE OF (IfcRelConnectsElements);
    RelatingPriorities : LIST [0:?] OF IfcInteger;
    RelatedPriorities : LIST [0:?] OF IfcInteger;
    RelatedConnectionType : IfcConnectionTypeEnum;
    RelatingConnectionType : IfcConnectionTypeEnum;
WHERE
    NormalizedRelatingPriorities : (SIZEOF(RelatingPriorities) = 0)
OR
    (SIZEOF (QUERY (temp <* RelatingPriorities
        | {0 <= temp <= 100}
        )) = SIZEOF(RelatingPriorities));
    NormalizedRelatedPriorities : (SIZEOF(RelatedPriorities) = 0)
OR
    (SIZEOF (QUERY (temp <* RelatedPriorities

```

---

```

    | {0 <= temp <= 100}
  )) = SIZEOF(RelatedPriorities));
END_ENTITY;

ENTITY IfcRelConnectsPortToElement
  SUBTYPE OF (IfcRelConnects);
    RelatingPort : IfcPort;
    RelatedElement : IfcDistributionElement;
END_ENTITY;

ENTITY IfcRelConnectsPorts
  SUBTYPE OF (IfcRelConnects);
    RelatingPort : IfcPort;
    RelatedPort : IfcPort;
    RealizingElement : OPTIONAL IfcElement;
  WHERE
    NoSelfReference : RelatingPort :<>: RelatedPort;
END_ENTITY;

ENTITY IfcRelConnectsStructuralActivity
  SUBTYPE OF (IfcRelConnects);
    RelatingElement : IfcStructuralActivityAssignmentSelect;
    RelatedStructuralActivity : IfcStructuralActivity;
END_ENTITY;

ENTITY IfcRelConnectsStructuralMember
  SUPERTYPE OF (ONEOF
    (IfcRelConnectsWithEccentricity))
  SUBTYPE OF (IfcRelConnects);
    RelatingStructuralMember : IfcStructuralMember;
    RelatedStructuralConnection : IfcStructuralConnection;
    AppliedCondition : OPTIONAL IfcBoundaryCondition;
    AdditionalConditions : OPTIONAL IfcStructuralConnectionCondition;
    SupportedLength : OPTIONAL IfcLengthMeasure;
    ConditionCoordinateSystem : OPTIONAL IfcAxis2Placement3D;
END_ENTITY;

ENTITY IfcRelConnectsWithEccentricity
  SUBTYPE OF (IfcRelConnectsStructuralMember);
    ConnectionConstraint : IfcConnectionGeometry;
END_ENTITY;

ENTITY IfcRelConnectsWithRealizingElements
  SUBTYPE OF (IfcRelConnectsElements);

```

---

```

    RealizingElements : SET [1:?] OF IfcElement;
    ConnectionType : OPTIONAL IfcLabel;
END_ENTITY;

ENTITY IfcRelContainedInSpatialStructure
  SUBTYPE OF (IfcRelConnects);
    RelatedElements : SET [1:?] OF IfcProduct;
    RelatingStructure : IfcSpatialElement;
  WHERE
    WR31      :      SIZEOF(QUERY(temp      <*      RelatedElements
' IFC4. IFCSPATIALSTRUCTUREELEMENT' IN TYPEOF(temp))) = 0;
END_ENTITY;

ENTITY IfcRelCoversBldgElements
  SUBTYPE OF (IfcRelConnects);
    RelatingBuildingElement : IfcElement;
    RelatedCoverings : SET [1:?] OF IfcCovering;
END_ENTITY;

ENTITY IfcRelCoversSpaces
  SUBTYPE OF (IfcRelConnects);
    RelatingSpace : IfcSpace;
    RelatedCoverings : SET [1:?] OF IfcCovering;
END_ENTITY;

ENTITY IfcRelDeclares
  SUBTYPE OF (IfcRelationship);
    RelatingContext : IfcContext;
    RelatedDefinitions : SET [1:?] OF IfcDefinitionSelect;
  WHERE
    NoSelfReference : SIZEOF(QUERY(Temp <* RelatedDefinitions | RelatingContext :=:
Temp)) = 0;
END_ENTITY;

ENTITY IfcRelDecomposes
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcRelAggregates
    , IfcRelNests
    , IfcRelProjectsElement
    , IfcRelVoidsElement))
  SUBTYPE OF (IfcRelationship);
END_ENTITY;

ENTITY IfcRelDefines

```

---

```

ABSTRACT SUPERTYPE OF (ONEOF
    (IfcRelDefinesByObject
    , IfcRelDefinesByProperties
    , IfcRelDefinesByTemplate
    , IfcRelDefinesByType))
SUBTYPE OF (IfcRelationship);
END_ENTITY;

ENTITY IfcRelDefinesByObject
SUBTYPE OF (IfcRelDefines);
    RelatedObjects : SET [1:?] OF IfcObject;
    RelatingObject : IfcObject;
END_ENTITY;

ENTITY IfcRelDefinesByProperties
SUBTYPE OF (IfcRelDefines);
    RelatedObjects : SET [1:?] OF IfcObjectDefinition;
    RelatingPropertyDefinition : IfcPropertySetDefinitionSelect;
WHERE
    NoRelatedTypeObject : SIZEOF(QUERY(Types < *
SELF\IfcRelDefinesByProperties.RelatedObjects | ' IFC4. IFCTYPEOBJECT' IN
TYPEOF(Types))) = 0;
END_ENTITY;

ENTITY IfcRelDefinesByTemplate
SUBTYPE OF (IfcRelDefines);
    RelatedPropertySets : SET [1:?] OF IfcPropertySetDefinition;
    RelatingTemplate : IfcPropertySetTemplate;
END_ENTITY;

ENTITY IfcRelDefinesByType
SUBTYPE OF (IfcRelDefines);
    RelatedObjects : SET [1:?] OF IfcObject;
    RelatingType : IfcTypeObject;
END_ENTITY;

ENTITY IfcRelFillsElement
SUBTYPE OF (IfcRelConnects);
    RelatingOpeningElement : IfcOpeningElement;
    RelatedBuildingElement : IfcElement;
END_ENTITY;

ENTITY IfcRelFlowControlElements
SUBTYPE OF (IfcRelConnects);

```

---

```

        RelatedControlElements : SET [1:?] OF IfcDistributionControlElement;
        RelatingFlowElement : IfcDistributionFlowElement;
END_ENTITY;

ENTITY IfcRelInterferesElements
    SUBTYPE OF (IfcRelConnects);
        RelatingElement : IfcElement;
        RelatedElement : IfcElement;
        InterferenceGeometry : OPTIONAL IfcConnectionGeometry;
        InterferenceType : OPTIONAL IfcIdentifier;
        ImpliedOrder : LOGICAL;
    WHERE
        NotSelfReference : RelatingElement :<>: RelatedElement;
END_ENTITY;

ENTITY IfcRelNests
    SUBTYPE OF (IfcRelDecomposes);
        RelatingObject : IfcObjectDefinition;
        RelatedObjects : LIST [1:?] OF IfcObjectDefinition;
    WHERE
        NoSelfReference : SIZEOF(QUERY(Temp < * RelatedObjects | RelatingObject :=:
Temp)) = 0;
END_ENTITY;

ENTITY IfcRelProjectsElement
    SUBTYPE OF (IfcRelDecomposes);
        RelatingElement : IfcElement;
        RelatedFeatureElement : IfcFeatureElementAddition;
END_ENTITY;

ENTITY IfcRelReferencedInSpatialStructure
    SUBTYPE OF (IfcRelConnects);
        RelatedElements : SET [1:?] OF IfcProduct;
        RelatingStructure : IfcSpatialElement;
    WHERE
        AllowedRelatedElements : SIZEOF(QUERY(temp < * RelatedElements |
(' IFC4. IFCSPATIALSTRUCTUREELEMENT' IN TYPEOF(temp)) AND (NOT (' IFC4. IFCSPACE' IN
TYPEOF(temp))))
)) = 0;
END_ENTITY;

ENTITY IfcRelSequence
    SUBTYPE OF (IfcRelConnects);
        RelatingProcess : IfcProcess;

```

---

```

    RelatedProcess : IfcProcess;
    TimeLag : OPTIONAL IfcLagTime;
    SequenceType : OPTIONAL IfcSequenceEnum;
    UserDefinedSequenceType : OPTIONAL IfcLabel;
WHERE
    AvoidInconsistentSequence : RelatingProcess :<>: RelatedProcess;
    CorrectSequenceType : (SequenceType <> IfcSequenceEnum.USERDEFINED) OR
((SequenceType = IfcSequenceEnum.USERDEFINED) AND
EXISTS(UserDefinedSequenceType));
END_ENTITY;

ENTITY IfcRelServicesBuildings
    SUBTYPE OF (IfcRelConnects);
    RelatingSystem : IfcSystem;
    RelatedBuildings : SET [1:?] OF IfcSpatialElement;
END_ENTITY;

ENTITY IfcRelSpaceBoundary
    SUPERTYPE OF (ONEOF
        (IfcRelSpaceBoundary1stLevel))
    SUBTYPE OF (IfcRelConnects);
    RelatingSpace : IfcSpaceBoundarySelect;
    RelatedBuildingElement : IfcElement;
    ConnectionGeometry : OPTIONAL IfcConnectionGeometry;
    PhysicalOrVirtualBoundary : IfcPhysicalOrVirtualEnum;
    InternalOrExternalBoundary : IfcInternalOrExternalEnum;
WHERE
    CorrectPhysOrVirt : ((PhysicalOrVirtualBoundary =
IfcPhysicalOrVirtualEnum.Physical)
    AND (NOT('IFC4.IFCVIRTUALELEMENT' IN TYPEOF(RelatedBuildingElement))))
OR
    ((PhysicalOrVirtualBoundary = IfcPhysicalOrVirtualEnum.Virtual)
    AND (('IFC4.IFCVIRTUALELEMENT' IN TYPEOF(RelatedBuildingElement))
    OR ('IFC4.IFCOPENINGELEMENT' IN TYPEOF(RelatedBuildingElement))))
OR
    (PhysicalOrVirtualBoundary = IfcPhysicalOrVirtualEnum.NotDefined);
END_ENTITY;

ENTITY IfcRelSpaceBoundary1stLevel
    SUPERTYPE OF (ONEOF
        (IfcRelSpaceBoundary2ndLevel))
    SUBTYPE OF (IfcRelSpaceBoundary);
    ParentBoundary : OPTIONAL IfcRelSpaceBoundary1stLevel;
    INVERSE

```

---

InnerBoundaries : SET [0:?] OF IfcRelSpaceBoundary1stLevel FOR ParentBoundary;  
END\_ENTITY;

ENTITY IfcRelSpaceBoundary2ndLevel  
SUBTYPE OF (IfcRelSpaceBoundary1stLevel);  
CorrespondingBoundary : OPTIONAL IfcRelSpaceBoundary2ndLevel;  
INVERSE  
Corresponds : SET [0:1] OF IfcRelSpaceBoundary2ndLevel FOR  
CorrespondingBoundary;  
END\_ENTITY;

ENTITY IfcRelVoidsElement  
SUBTYPE OF (IfcRelDecomposes);  
RelatingBuildingElement : IfcElement;  
RelatedOpeningElement : IfcFeatureElementSubtraction;  
END\_ENTITY;

ENTITY IfcRelationship  
ABSTRACT SUPERTYPE OF (ONEOF  
(IfcRelAssigns  
, IfcRelAssociates  
, IfcRelConnects  
, IfcRelDeclares  
, IfcRelDecomposes  
, IfcRelDefines))  
SUBTYPE OF (IfcRoot);  
END\_ENTITY;

ENTITY IfcReparametrisedCompositeCurveSegment  
SUBTYPE OF (IfcCompositeCurveSegment);  
ParamLength : IfcParameterValue;  
WHERE  
PositiveLengthParameter : ParamLength > 0.0;  
END\_ENTITY;

ENTITY IfcRepresentation  
ABSTRACT SUPERTYPE OF (ONEOF  
(IfcShapeModel  
, IfcStyleModel));  
ContextOfItems : IfcRepresentationContext;  
RepresentationIdentifier : OPTIONAL IfcLabel;  
RepresentationType : OPTIONAL IfcLabel;  
Items : SET [1:?] OF IfcRepresentationItem;  
INVERSE



---

```

RepresentationMap : SET [0:1] OF IfcRepresentationMap FOR
MappedRepresentation;
LayerAssignments : SET [0:?] OF IfcPresentationLayerAssignment FOR
AssignedItems;
OfProductRepresentation : SET [0:?] OF IfcProductRepresentation FOR
Representations;
END_ENTITY;
```

```

ENTITY IfcRepresentationContext
ABSTRACT SUPERTYPE OF (ONEOF
    (IfcGeometricRepresentationContext));
ContextIdentifier : OPTIONAL IfcLabel;
ContextType : OPTIONAL IfcLabel;
INVERSE
    RepresentationsInContext : SET [0:?] OF IfcRepresentation FOR ContextOfItems;
END_ENTITY;
```

```

ENTITY IfcRepresentationItem
ABSTRACT SUPERTYPE OF (ONEOF
    (IfcGeometricRepresentationItem
    , IfcMappedItem
    , IfcStyledItem
    , IfcTopologicalRepresentationItem));
INVERSE
    LayerAssignment : SET [0:1] OF IfcPresentationLayerAssignment FOR
AssignedItems;
    StyledByItem : SET [0:1] OF IfcStyledItem FOR Item;
END_ENTITY;
```

```

ENTITY IfcRepresentationMap;
MappingOrigin : IfcAxis2Placement;
MappedRepresentation : IfcRepresentation;
INVERSE
    HasShapeAspects : SET [0:?] OF IfcShapeAspect FOR
PartOfProductDefinitionShape;
    MapUsage : SET [0:?] OF IfcMappedItem FOR MappingSource;
WHERE
    ApplicableMappedRepr : ' IFC4. IFCSHAPEMODEL' IN TYPEOF (MappedRepresentation);
END_ENTITY;
```

```

ENTITY IfcResource
ABSTRACT SUPERTYPE OF (ONEOF
    (IfcConstructionResource))
SUBTYPE OF (IfcObject);
```

---

```

        Identification : OPTIONAL IfcIdentifier;
        LongDescription : OPTIONAL IfcText;
    INVERSE
        ResourceOf : SET [0:?] OF IfcRelAssignsToResource FOR RelatingResource;
END_ENTITY;

ENTITY IfcResourceApprovalRelationship
    SUBTYPE OF (IfcResourceLevelRelationship);
        RelatedResourceObjects : SET [1:?] OF IfcResourceObjectSelect;
        RelatingApproval : IfcApproval;
END_ENTITY;

ENTITY IfcResourceConstraintRelationship
    SUBTYPE OF (IfcResourceLevelRelationship);
        RelatingConstraint : IfcConstraint;
        RelatedResourceObjects : SET [1:?] OF IfcResourceObjectSelect;
END_ENTITY;

ENTITY IfcResourceLevelRelationship
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcApprovalRelationship
        , IfcCurrencyRelationship
        , IfcDocumentInformationRelationship
        , IfcExternalReferenceRelationship
        , IfcMaterialRelationship
        , IfcOrganizationRelationship
        , IfcPropertyDependencyRelationship
        , IfcResourceApprovalRelationship
        , IfcResourceConstraintRelationship));
        Name : OPTIONAL IfcLabel;
        Description : OPTIONAL IfcText;
END_ENTITY;

ENTITY IfcResourceTime
    SUBTYPE OF (IfcSchedulingTime);
        ScheduleWork : OPTIONAL IfcDuration;
        ScheduleUsage : OPTIONAL IfcPositiveRatioMeasure;
        ScheduleStart : OPTIONAL IfcDateTime;
        ScheduleFinish : OPTIONAL IfcDateTime;
        ScheduleContour : OPTIONAL IfcLabel;
        LevelingDelay : OPTIONAL IfcDuration;
        IsOverAllocated : OPTIONAL IfcBoolean;
        StatusTime : OPTIONAL IfcDateTime;
        ActualWork : OPTIONAL IfcDuration;

```

---

```

    ActualUsage : OPTIONAL IfcPositiveRatioMeasure;
    ActualStart : OPTIONAL IfcDateTime;
    ActualFinish : OPTIONAL IfcDateTime;
    RemainingWork : OPTIONAL IfcDuration;
    RemainingUsage : OPTIONAL IfcPositiveRatioMeasure;
    Completion : OPTIONAL IfcPositiveRatioMeasure;
END_ENTITY;

ENTITY IfcRevolvedAreaSolid
  SUPERTYPE OF (ONEOF
    (IfcRevolvedAreaSolidTapered))
  SUBTYPE OF (IfcSweptAreaSolid);
  Axis : IfcAxis1Placement;
  Angle : IfcPlaneAngleMeasure;
  DERIVE
    AxisLine      :      IfcLine      :=      IfcRepresentationItem()      ||
IfcGeometricRepresentationItem () || IfcCurve() || IfcLine(Axis.Location,
  IfcRepresentationItem()      ||      IfcGeometricRepresentationItem      ()      ||
IfcVector(Axis.Z, 1.0));
  WHERE
    AxisStartInXY : Axis.Location.Coordinates[3] = 0.0;
    AxisDirectionInXY : Axis.Z.DirectionRatios[3] = 0.0;
END_ENTITY;

ENTITY IfcRevolvedAreaSolidTapered
  SUBTYPE OF (IfcRevolvedAreaSolid);
  EndSweptArea : IfcProfileDef;
  WHERE
    CorrectProfileAssignment :
IfcTaperedSweptAreaProfiles(SELf\IfcSweptAreaSolid.SweptArea,
SELf.EndSweptArea);
END_ENTITY;

ENTITY IfcRightCircularCone
  SUBTYPE OF (IfcCsgPrimitive3D);
  Height : IfcPositiveLengthMeasure;
  BottomRadius : IfcPositiveLengthMeasure;
END_ENTITY;

ENTITY IfcRightCircularCylinder
  SUBTYPE OF (IfcCsgPrimitive3D);
  Height : IfcPositiveLengthMeasure;
  Radius : IfcPositiveLengthMeasure;
END_ENTITY;

```

---

```

ENTITY IfcRoof
  SUBTYPE OF (IfcBuildingElement);
    PredefinedType : OPTIONAL IfcRoofTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcRoofTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcRoofTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCROOFTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcRoofType
  SUBTYPE OF (IfcBuildingElementType);
    PredefinedType : IfcRoofTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcRoofTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcRoofTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcRoot
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcObjectDefinition
    , IfcPropertyDefinition
    , IfcRelationship));
    GlobalId : IfcGloballyUniqueId;
    OwnerHistory : OPTIONAL IfcOwnerHistory;
    Name : OPTIONAL IfcLabel;
    Description : OPTIONAL IfcText;
  UNIQUE
    URI : GlobalId;
END_ENTITY;

```

```

ENTITY IfcRoundedRectangleProfileDef
  SUBTYPE OF (IfcRectangleProfileDef);
    RoundingRadius : IfcPositiveLengthMeasure;
  WHERE
    ValidRadius : ((RoundingRadius <= (SELF\IfcRectangleProfileDef.XDim/2.)) AND
    (RoundingRadius <= (SELF\IfcRectangleProfileDef.YDim/2.)));
END_ENTITY;

```

```

ENTITY IfcSIUnit

```

---

```

SUBTYPE OF (IfcNamedUnit);
  Prefix : OPTIONAL IfcSIPrefix;
  Name : IfcSIUnitName;
DERIVE
  SELF\IfcNamedUnit.Dimensions      :      IfcDimensionalExponents      :=
IfcDimensionsForSiUnit (SELF.Name);
END_ENTITY;

```

```

ENTITY IfcSanitaryTerminal
  SUBTYPE OF (IfcFlowTerminal);
    PredefinedType : OPTIONAL IfcSanitaryTerminalTypeEnum;
  WHERE
    CorrectPredefinedType : NOT (EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcSanitaryTerminalTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcSanitaryTerminalTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCSANITARYTERMINALTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
  END_ENTITY;

```

```

ENTITY IfcSanitaryTerminalType
  SUBTYPE OF (IfcFlowTerminalType);
    PredefinedType : IfcSanitaryTerminalTypeEnum;
  WHERE
    CorrectPredefinedType      :      (PredefinedType      <>
IfcSanitaryTerminalTypeEnum.USERDEFINED) OR
    ((PredefinedType      =      IfcSanitaryTerminalTypeEnum.USERDEFINED)      AND
EXISTS(SELF\IfcElementType.ElementType));
  END_ENTITY;

```

```

ENTITY IfcSchedulingTime
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcEventTime
    , IfcLagTime
    , IfcResourceTime
    , IfcTaskTime
    , IfcWorkTime));
  Name : OPTIONAL IfcLabel;
  DataOrigin : OPTIONAL IfcDataOriginEnum;
  UserDefinedDataOrigin : OPTIONAL IfcLabel;
END_ENTITY;

```

```

ENTITY IfcSeamCurve

```

---

```

SUBTYPE OF (IfcSurfaceCurve);
WHERE
    TwoPCurves : SIZEOF(SELF\IfcSurfaceCurve.AssociatedGeometry) = 2;
    SameSurface :
IfcAssociatedSurface(SELF\IfcSurfaceCurve.AssociatedGeometry[1]) =
IfcAssociatedSurface(SELF\IfcSurfaceCurve.AssociatedGeometry[2]);
END_ENTITY;

ENTITY IfcSectionProperties
SUBTYPE OF (IfcPreDefinedProperties);
    SectionType : IfcSectionTypeEnum;
    StartProfile : IfcProfileDef;
    EndProfile : OPTIONAL IfcProfileDef;
END_ENTITY;

ENTITY IfcSectionReinforcementProperties
SUBTYPE OF (IfcPreDefinedProperties);
    LongitudinalStartPosition : IfcLengthMeasure;
    LongitudinalEndPosition : IfcLengthMeasure;
    TransversePosition : OPTIONAL IfcLengthMeasure;
    ReinforcementRole : IfcReinforcingBarRoleEnum;
    SectionDefinition : IfcSectionProperties;
    CrossSectionReinforcementDefinitions : SET [1:?] OF
IfcReinforcementBarProperties;
END_ENTITY;

ENTITY IfcSectionedSpine
SUBTYPE OF (IfcGeometricRepresentationItem);
    SpineCurve : IfcCompositeCurve;
    CrossSections : LIST [2:?] OF IfcProfileDef;
    CrossSectionPositions : LIST [2:?] OF IfcAxis2Placement3D;
DERIVE
    Dim : IfcDimensionCount := 3;
WHERE
    CorrespondingSectionPositions : SIZEOF(CrossSections) =
SIZEOF(CrossSectionPositions);
    ConsistentProfileTypes : SIZEOF(QUERY(temp <* CrossSections |
CrossSections[1].ProfileType <> temp.ProfileType)) = 0;
    SpineCurveDim : SpineCurve.Dim = 3;
END_ENTITY;

ENTITY IfcSensor
SUBTYPE OF (IfcDistributionControlElement);
    PredefinedType : OPTIONAL IfcSensorTypeEnum;

```

---

```

WHERE
    CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
    (PredefinedType <> IfcSensorTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcSensorTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCSENSORTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcSensorType
    SUBTYPE OF (IfcDistributionControlElementType);
    PredefinedType : IfcSensorTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <> IfcSensorTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcSensorTypeEnum.USERDEFINED) AND
        EXISTS (SELF\IfcElementType.ElementType));
    END_ENTITY;

ENTITY IfcShadingDevice
    SUBTYPE OF (IfcBuildingElement);
    PredefinedType : OPTIONAL IfcShadingDeviceTypeEnum;
    WHERE
        CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
        (PredefinedType <> IfcShadingDeviceTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcShadingDeviceTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject.ObjectType));
        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
        ('IFC4.IFCSHADINGDEVICETYPE' IN
        TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
    END_ENTITY;

ENTITY IfcShadingDeviceType
    SUBTYPE OF (IfcBuildingElementType);
    PredefinedType : IfcShadingDeviceTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <>
        IfcShadingDeviceTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcShadingDeviceTypeEnum.USERDEFINED) AND
        EXISTS (SELF\IfcElementType.ElementType));
    END_ENTITY;

ENTITY IfcShapeAspect;
    ShapeRepresentations : LIST [1:?] OF IfcShapeModel;
    Name : OPTIONAL IfcLabel;

```

---

```

    Description : OPTIONAL IfcText;
    ProductDefinitional : IfcLogical;
    PartOfProductDefinitionShape : OPTIONAL IfcProductRepresentationSelect;
END_ENTITY;

ENTITY IfcShapeModel
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcShapeRepresentation
      , IfcTopologyRepresentation))
  SUBTYPE OF (IfcRepresentation);
  INVERSE
    OfShapeAspect : SET [0:1] OF IfcShapeAspect FOR ShapeRepresentations;
  WHERE
    WR11 : (SIZEOF(SELF\IfcRepresentation.OfProductRepresentation) = 1) XOR
    (SIZEOF(SELF\IfcRepresentation.RepresentationMap) = 1) XOR
    (SIZEOF(OfShapeAspect) = 1);
END_ENTITY;

ENTITY IfcShapeRepresentation
  SUBTYPE OF (IfcShapeModel);
  WHERE
    CorrectContext : ' IFC4. IFCGEOMETRICREPRESENTATIONCONTEXT'
  IN TYPEOF(SELF\IfcRepresentation.ContextOfItems);
    NoTopologicalItem : SIZEOF(QUERY(temp <* Items |
    (' IFC4. IFCTOPOLOGICALREPRESENTATIONITEM' IN TYPEOF(temp))
    AND (NOT (SIZEOF(
      [' IFC4. IFCVERTEXPOINT',
        ' IFC4. IFCEDGECURVE',
        ' IFC4. IFCFACESURFACE' ] * TYPEOF(temp)) = 1))
    )) = 0;
    HasRepresentationType : EXISTS(SELF\IfcRepresentation.RepresentationType);
    HasRepresentationIdentifier :
  EXISTS(SELF\IfcRepresentation.RepresentationIdentifier);
    CorrectItemsForType :
  IfcShapeRepresentationTypes(SELF\IfcRepresentation.RepresentationType,
  SELF\IfcRepresentation.Items);
END_ENTITY;

ENTITY IfcShellBasedSurfaceModel
  SUBTYPE OF (IfcGeometricRepresentationItem);
    SbsmBoundary : SET [1:?] OF IfcShell;
  DERIVE
    Dim : IfcDimensionCount := 3;
END_ENTITY;

```



---

```

ENTITY IfcSimpleProperty
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcPropertyBoundedValue
    , IfcPropertyEnumeratedValue
    , IfcPropertyListValue
    , IfcPropertyReferenceValue
    , IfcPropertySingleValue
    , IfcPropertyTableValue))
  SUBTYPE OF (IfcProperty);
END_ENTITY;

ENTITY IfcSimplePropertyTemplate
  SUBTYPE OF (IfcPropertyTemplate);
    TemplateType : OPTIONAL IfcSimplePropertyTemplateTypeEnum;
    PrimaryMeasureType : OPTIONAL IfcLabel;
    SecondaryMeasureType : OPTIONAL IfcLabel;
    Enumerators : OPTIONAL IfcPropertyEnumeration;
    PrimaryUnit : OPTIONAL IfcUnit;
    SecondaryUnit : OPTIONAL IfcUnit;
    Expression : OPTIONAL IfcLabel;
    AccessState : OPTIONAL IfcStateEnum;
END_ENTITY;

ENTITY IfcSite
  SUBTYPE OF (IfcSpatialStructureElement);
    RefLatitude : OPTIONAL IfcCompoundPlaneAngleMeasure;
    RefLongitude : OPTIONAL IfcCompoundPlaneAngleMeasure;
    RefElevation : OPTIONAL IfcLengthMeasure;
    LandTitleNumber : OPTIONAL IfcLabel;
    SiteAddress : OPTIONAL IfcPostalAddress;
END_ENTITY;

ENTITY IfcSlab
  SUPERTYPE OF (ONEOF
    (IfcSlabElementedCase
    , IfcSlabStandardCase))
  SUBTYPE OF (IfcBuildingElement);
    PredefinedType : OPTIONAL IfcSlabTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcSlabTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcSlabTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));

```

---

```

        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
        (' IFC4. IFCSLABTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcSlabElementedCase
    SUBTYPE OF (IfcSlab);
    WHERE
        HasDecomposition : HIINDEX(SELF\IfcObjectDefinition.IsDecomposedBy) > 0;
END_ENTITY;

ENTITY IfcSlabStandardCase
    SUBTYPE OF (IfcSlab);
    WHERE
        HasMaterialLayerSetusage : SIZEOF (QUERY(temp <* USEDIN(SELF,
' IFC4. IFCRELASSOCIATES.RELATEDOBJECTS' ) |
                (' IFC4. IFCRELASSOCIATESMATERIAL' IN TYPEOF(temp)) AND
                (' IFC4. IFCMATERIALLAYERSETUSAGE' IN TYPEOF(temp.RelatingMaterial))
                )) = 1;
END_ENTITY;

ENTITY IfcSlabType
    SUBTYPE OF (IfcBuildingElementType);
    PredefinedType : IfcSlabTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <> IfcSlabTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcSlabTypeEnum.USERDEFINED) AND
        EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcSlippageConnectionCondition
    SUBTYPE OF (IfcStructuralConnectionCondition);
    SlippageX : OPTIONAL IfcLengthMeasure;
    SlippageY : OPTIONAL IfcLengthMeasure;
    SlippageZ : OPTIONAL IfcLengthMeasure;
END_ENTITY;

ENTITY IfcSolarDevice
    SUBTYPE OF (IfcEnergyConversionDevice);
    PredefinedType : OPTIONAL IfcSolarDeviceTypeEnum;
    WHERE
        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
        (PredefinedType <> IfcSolarDeviceTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcSolarDeviceTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject.ObjectType));

```

---

```

        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
(' IFC4. IFCSolarDeviceType'
                                                                    IN
        TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcSolarDeviceType
    SUBTYPE OF (IfcEnergyConversionDeviceType);
        PredefinedType : IfcSolarDeviceTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <> IfcSolarDeviceTypeEnum.USERDEFINED)
OR
    ((PredefinedType = IfcSolarDeviceTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcSolidModel
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcCsgSolid
        , IfcManifoldSolidBrep
        , IfcSweptAreaSolid
        , IfcSweptDiskSolid))
    SUBTYPE OF (IfcGeometricRepresentationItem);
    DERIVE
        Dim : IfcDimensionCount := 3;
END_ENTITY;

```

```

ENTITY IfcSpace
    SUBTYPE OF (IfcSpatialStructureElement);
        PredefinedType : OPTIONAL IfcSpaceTypeEnum;
        ElevationWithFlooring : OPTIONAL IfcLengthMeasure;
    INVERSE
        HasCoverings : SET [0:?] OF IfcRelCoversSpaces FOR RelatingSpace;
        BoundedBy : SET [0:?] OF IfcRelSpaceBoundary FOR RelatingSpace;
    WHERE
        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
        (PredefinedType <> IfcSpaceTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcSpaceTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject.ObjectType));
        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
        (' IFC4. IFCSpaceType' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcSpaceHeater
    SUBTYPE OF (IfcFlowTerminal);

```

---

```

    PredefinedType : OPTIONAL IfcSpaceHeaterTypeEnum;
WHERE
    CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
    (PredefinedType <> IfcSpaceHeaterTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcSpaceHeaterTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCSPACEHEATERTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcSpaceHeaterType
    SUBTYPE OF (IfcFlowTerminalType);
    PredefinedType : IfcSpaceHeaterTypeEnum;
WHERE
    CorrectPredefinedType : (PredefinedType <> IfcSpaceHeaterTypeEnum.USERDEFINED)
OR
    ((PredefinedType = IfcSpaceHeaterTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcSpaceType
    SUBTYPE OF (IfcSpatialStructureElementType);
    PredefinedType : IfcSpaceTypeEnum;
    LongName : OPTIONAL IfcLabel;
WHERE
    CorrectPredefinedType : (PredefinedType <> IfcSpaceTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcSpaceTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcSpatialElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcSpatialElement
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcExternalSpatialStructureElement
        , IfcSpatialStructureElement
        , IfcSpatialZone))
    SUBTYPE OF (IfcProduct);
    LongName : OPTIONAL IfcLabel;
    INVERSE
        ContainsElements : SET [0:?] OF IfcRelContainedInSpatialStructure FOR
        RelatingStructure;
        ServicedBySystems : SET [0:?] OF IfcRelServicesBuildings FOR RelatedBuildings;
        ReferencesElements : SET [0:?] OF IfcRelReferencedInSpatialStructure FOR
        RelatingStructure;

```

---

```

END_ENTITY;

ENTITY IfcSpatialElementType
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcSpatialStructureElementType
    , IfcSpatialZoneType))
  SUBTYPE OF (IfcTypeProduct);
    ElementType : OPTIONAL IfcLabel;
END_ENTITY;

ENTITY IfcSpatialStructureElement
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcBuilding
    , IfcBuildingStorey
    , IfcSite
    , IfcSpace))
  SUBTYPE OF (IfcSpatialElement);
    CompositionType : OPTIONAL IfcElementCompositionEnum;
  WHERE
    WR41 : (HIINDEX(SELf\IfcObjectDefinition.Decomposes) = 1)
  AND
    (' IFC4. IFCRELAGGREGATES' IN TYPEOF(SELf\IfcObjectDefinition.Decomposes[1]))
  AND
    ((' IFC4. IFCPROJECT' IN TYPEOF
    (SELf\IfcObjectDefinition.Decomposes[1].RelatingObject)) OR
    (' IFC4. IFCSPATIALSTRUCTUREELEMENT' IN TYPEOF
    (SELf\IfcObjectDefinition.Decomposes[1].RelatingObject))
  );
END_ENTITY;

ENTITY IfcSpatialStructureElementType
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcSpaceType))
  SUBTYPE OF (IfcSpatialElementType);
END_ENTITY;

ENTITY IfcSpatialZone
  SUBTYPE OF (IfcSpatialElement);
    PredefinedType : OPTIONAL IfcSpatialZoneTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcSpatialZoneTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcSpatialZoneTypeEnum.USERDEFINED) AND EXISTS
    (SELf\IfcObject.ObjectType));

```

---

```

        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
        (' IFC4. IFCSPATIALZONETYPE'
                                                    IN
        TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;
```

```

ENTITY IfcSpatialZoneType
    SUBTYPE OF (IfcSpatialElementType);
        PredefinedType : IfcSpatialZoneTypeEnum;
        LongName : OPTIONAL IfcLabel;
    WHERE
        CorrectPredefinedType : (PredefinedType <> IfcSpatialZoneTypeEnum.USERDEFINED)
OR
        ((PredefinedType = IfcSpatialZoneTypeEnum.USERDEFINED) AND
        EXISTS(SELF\IfcSpatialElementType.ElementType));
END_ENTITY;
```

```

ENTITY IfcSphere
    SUBTYPE OF (IfcCsgPrimitive3D);
        Radius : IfcPositiveLengthMeasure;
END_ENTITY;
```

```

ENTITY IfcSphericalSurface
    SUBTYPE OF (IfcElementarySurface);
        Radius : IfcPositiveLengthMeasure;
END_ENTITY;
```

```

ENTITY IfcStackTerminal
    SUBTYPE OF (IfcFlowTerminal);
        PredefinedType : OPTIONAL IfcStackTerminalTypeEnum;
    WHERE
        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
        (PredefinedType <> IfcStackTerminalTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcStackTerminalTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject.ObjectType));
        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
        (' IFC4. IFCSTACKTERMINALTYPE'
                                                    IN
        TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;
```

```

ENTITY IfcStackTerminalType
    SUBTYPE OF (IfcFlowTerminalType);
        PredefinedType : IfcStackTerminalTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <>
```

---

```

IfcStackTerminalTypeEnum.USERDEFINED) OR
((PredefinedType = IfcStackTerminalTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcStair
  SUBTYPE OF (IfcBuildingElement);
    PredefinedType : OPTIONAL IfcStairTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcStairTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcStairTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCSTAIRTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcStairFlight
  SUBTYPE OF (IfcBuildingElement);
    NumberOfRisers : OPTIONAL IfcInteger;
    NumberOfTreads : OPTIONAL IfcInteger;
    RiserHeight : OPTIONAL IfcPositiveLengthMeasure;
    TreadLength : OPTIONAL IfcPositiveLengthMeasure;
    PredefinedType : OPTIONAL IfcStairFlightTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcStairFlightTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcStairFlightTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCSTAIRFLIGHTTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcStairFlightType
  SUBTYPE OF (IfcBuildingElementType);
    PredefinedType : IfcStairFlightTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcStairFlightTypeEnum.USERDEFINED)
OR
    ((PredefinedType = IfcStairFlightTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

---

```

ENTITY IfcStairType
  SUBTYPE OF (IfcBuildingElementType);
    PredefinedType : IfcStairTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcStairTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcStairTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcStructuralAction
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcStructuralCurveAction
    , IfcStructuralPointAction
    , IfcStructuralSurfaceAction))
  SUBTYPE OF (IfcStructuralActivity);
    DestabilizingLoad : OPTIONAL IfcBoolean;
END_ENTITY;

```

```

ENTITY IfcStructuralActivity
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcStructuralAction
    , IfcStructuralReaction))
  SUBTYPE OF (IfcProduct);
    AppliedLoad : IfcStructuralLoad;
    GlobalOrLocal : IfcGlobalOrLocalEnum;
  INVERSE
    AssignedToStructuralItem : SET [0:1] OF IfcRelConnectsStructuralActivity FOR
    RelatedStructuralActivity;
END_ENTITY;

```

```

ENTITY IfcStructuralAnalysisModel
  SUBTYPE OF (IfcSystem);
    PredefinedType : IfcAnalysisModelTypeEnum;
    OrientationOf2DPlane : OPTIONAL IfcAxis2Placement3D;
    LoadedBy : OPTIONAL SET [1:?] OF IfcStructuralLoadGroup;
    HasResults : OPTIONAL SET [1:?] OF IfcStructuralResultGroup;
    SharedPlacement : OPTIONAL IfcObjectPlacement;
  WHERE
    HasObjectType : (PredefinedType <> IfcAnalysisModelTypeEnum.USERDEFINED) OR
    EXISTS(SELF\IfcObject.ObjectType);
END_ENTITY;

```

```

ENTITY IfcStructuralConnection
  ABSTRACT SUPERTYPE OF (ONEOF

```



---

```

        (IfcStructuralCurveConnection
        , IfcStructuralPointConnection
        , IfcStructuralSurfaceConnection))
SUBTYPE OF (IfcStructuralItem);
    AppliedCondition : OPTIONAL IfcBoundaryCondition;
INVERSE
    ConnectsStructuralMembers : SET [1:?] OF IfcRelConnectsStructuralMember FOR
RelatedStructuralConnection;
END_ENTITY;

```

```

ENTITY IfcStructuralConnectionCondition
ABSTRACT SUPERTYPE OF (ONEOF
    (IfcFailureConnectionCondition
    , IfcSlippageConnectionCondition));
    Name : OPTIONAL IfcLabel;
END_ENTITY;

```

```

ENTITY IfcStructuralCurveAction
SUPERTYPE OF (ONEOF
    (IfcStructuralLinearAction))
SUBTYPE OF (IfcStructuralAction);
    ProjectedOrTrue : OPTIONAL IfcProjectedOrTrueLengthEnum;
    PredefinedType : IfcStructuralCurveActivityTypeEnum;
WHERE
    ProjectedIsGlobal : (NOT EXISTS(ProjectedOrTrue)) OR
((ProjectedOrTrue <> PROJECTED_LENGTH) OR
    (SELF\IfcStructuralActivity.GlobalOrLocal = GLOBAL_COORDS));
    HasObjectType : (PredefinedType <>
IfcStructuralCurveActivityTypeEnum.USERDEFINED) OR
EXISTS(SELF\IfcObject.ObjectType);
    SuitablePredefinedType : PredefinedType <>
IfcStructuralCurveActivityTypeEnum.EQUIDISTANT;
END_ENTITY;

```

```

ENTITY IfcStructuralCurveConnection
SUBTYPE OF (IfcStructuralConnection);
    Axis : IfcDirection;
END_ENTITY;

```

```

ENTITY IfcStructuralCurveMember
SUPERTYPE OF (ONEOF
    (IfcStructuralCurveMemberVarying))
SUBTYPE OF (IfcStructuralMember);
    PredefinedType : IfcStructuralCurveMemberTypeEnum;

```

---

```

    Axis : IfcDirection;
WHERE
    HasObjectType          :          (PredefinedType          <>
IfcStructuralCurveMemberTypeEnum. USERDEFINED)          OR
EXISTS(SELF\IfcObject. ObjectType);
END_ENTITY;

ENTITY IfcStructuralCurveMemberVarying
    SUBTYPE OF (IfcStructuralCurveMember);
END_ENTITY;

ENTITY IfcStructuralCurveReaction
    SUBTYPE OF (IfcStructuralReaction);
    PredefinedType : IfcStructuralCurveActivityTypeEnum;
WHERE
    HasObjectType          :          (PredefinedType          <>
IfcStructuralCurveActivityTypeEnum. USERDEFINED)          OR
EXISTS(SELF\IfcObject. ObjectType);
    SuitablePredefinedType          :          (PredefinedType          <>
IfcStructuralCurveActivityTypeEnum. SINUS)          AND          (PredefinedType          <>
IfcStructuralCurveActivityTypeEnum. PARABOLA);
END_ENTITY;

ENTITY IfcStructuralItem
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcStructuralConnection
        , IfcStructuralMember))
    SUBTYPE OF (IfcProduct);
    INVERSE
        AssignedStructuralActivity : SET [0:?] OF IfcRelConnectsStructuralActivity FOR
        RelatingElement;
END_ENTITY;

ENTITY IfcStructuralLinearAction
    SUBTYPE OF (IfcStructuralCurveAction);
WHERE
    SuitableLoadType          :          SIZEOF([' IFC4. IFCSTRUCTURALLOADLINEARFORCE',
' IFC4. IFCSTRUCTURALLOADTEMPERATURE' ]          *
TYPEOF(SELF\IfcStructuralActivity. AppliedLoad)) = 1;
    ConstPredefinedType          :          SELF\IfcStructuralCurveAction. PredefinedType          =
IfcStructuralCurveActivityTypeEnum. CONST;
END_ENTITY;

ENTITY IfcStructuralLoad

```

---

```

ABSTRACT SUPERTYPE OF (ONEOF
    (IfcStructuralLoadConfiguration
    , IfcStructuralLoadOrResult));
    Name : OPTIONAL IfcLabel;
END_ENTITY;

ENTITY IfcStructuralLoadCase
    SUBTYPE OF (IfcStructuralLoadGroup);
        SelfWeightCoefficients : OPTIONAL LIST [3:3] OF IfcRatioMeasure;
    WHERE
        IsLoadCasePredefinedType : SELF\IfcStructuralLoadGroup.PredefinedType =
IfcLoadGroupTypeEnum.LOAD_CASE;
END_ENTITY;

ENTITY IfcStructuralLoadConfiguration
    SUBTYPE OF (IfcStructuralLoad);
        Values : LIST [1:?] OF IfcStructuralLoadOrResult;
        Locations : OPTIONAL LIST [1:?] OF UNIQUE LIST [1:2] OF IfcLengthMeasure;
    WHERE
        ValidListSize : NOT EXISTS(Locations) OR (SIZEOF(Locations) = SIZEOF(Values));
END_ENTITY;

ENTITY IfcStructuralLoadGroup
    SUPERTYPE OF (ONEOF
        (IfcStructuralLoadCase))
    SUBTYPE OF (IfcGroup);
        PredefinedType : IfcLoadGroupTypeEnum;
        ActionType : IfcActionTypeEnum;
        ActionSource : IfcActionSourceTypeEnum;
        Coefficient : OPTIONAL IfcRatioMeasure;
        Purpose : OPTIONAL IfcLabel;
    INVERSE
        SourceOfResultGroup : SET [0:1] OF IfcStructuralResultGroup FOR
ResultForLoadGroup;
        LoadGroupFor : SET [0:?] OF IfcStructuralAnalysisModel FOR LoadedBy;
    WHERE
        HasObjectType : (
            (PredefinedType <> IfcLoadGroupTypeEnum.USERDEFINED) AND
            (ActionType <> IfcActionTypeEnum.USERDEFINED) AND
            (ActionSource <> IfcActionSourceTypeEnum.USERDEFINED)
        ) OR EXISTS(SELF\IfcObject.ObjectType);
END_ENTITY;

ENTITY IfcStructuralLoadLinearForce

```

---

```

SUBTYPE OF (IfcStructuralLoadStatic);
    LinearForceX : OPTIONAL IfcLinearForceMeasure;
    LinearForceY : OPTIONAL IfcLinearForceMeasure;
    LinearForceZ : OPTIONAL IfcLinearForceMeasure;
    LinearMomentX : OPTIONAL IfcLinearMomentMeasure;
    LinearMomentY : OPTIONAL IfcLinearMomentMeasure;
    LinearMomentZ : OPTIONAL IfcLinearMomentMeasure;
END_ENTITY;

ENTITY IfcStructuralLoadOrResult
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcStructuralLoadStatic
        , IfcSurfaceReinforcementArea))
    SUBTYPE OF (IfcStructuralLoad);
END_ENTITY;

ENTITY IfcStructuralLoadPlanarForce
    SUBTYPE OF (IfcStructuralLoadStatic);
        PlanarForceX : OPTIONAL IfcPlanarForceMeasure;
        PlanarForceY : OPTIONAL IfcPlanarForceMeasure;
        PlanarForceZ : OPTIONAL IfcPlanarForceMeasure;
END_ENTITY;

ENTITY IfcStructuralLoadSingleDisplacement
    SUPERTYPE OF (ONEOF
        (IfcStructuralLoadSingleDisplacementDistortion))
    SUBTYPE OF (IfcStructuralLoadStatic);
        DisplacementX : OPTIONAL IfcLengthMeasure;
        DisplacementY : OPTIONAL IfcLengthMeasure;
        DisplacementZ : OPTIONAL IfcLengthMeasure;
        RotationalDisplacementRX : OPTIONAL IfcPlaneAngleMeasure;
        RotationalDisplacementRY : OPTIONAL IfcPlaneAngleMeasure;
        RotationalDisplacementRZ : OPTIONAL IfcPlaneAngleMeasure;
END_ENTITY;

ENTITY IfcStructuralLoadSingleDisplacementDistortion
    SUBTYPE OF (IfcStructuralLoadSingleDisplacement);
        Distortion : OPTIONAL IfcCurvatureMeasure;
END_ENTITY;

ENTITY IfcStructuralLoadSingleForce
    SUPERTYPE OF (ONEOF
        (IfcStructuralLoadSingleForceWarping))
    SUBTYPE OF (IfcStructuralLoadStatic);

```

---

```

    ForceX : OPTIONAL IfcForceMeasure;
    ForceY : OPTIONAL IfcForceMeasure;
    ForceZ : OPTIONAL IfcForceMeasure;
    MomentX : OPTIONAL IfcTorqueMeasure;
    MomentY : OPTIONAL IfcTorqueMeasure;
    MomentZ : OPTIONAL IfcTorqueMeasure;
END_ENTITY;

ENTITY IfcStructuralLoadSingleForceWarping
  SUBTYPE OF (IfcStructuralLoadSingleForce);
    WarpingMoment : OPTIONAL IfcWarpingMomentMeasure;
END_ENTITY;

ENTITY IfcStructuralLoadStatic
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcStructuralLoadLinearForce
    , IfcStructuralLoadPlanarForce
    , IfcStructuralLoadSingleDisplacement
    , IfcStructuralLoadSingleForce
    , IfcStructuralLoadTemperature))
  SUBTYPE OF (IfcStructuralLoadOrResult);
END_ENTITY;

ENTITY IfcStructuralLoadTemperature
  SUBTYPE OF (IfcStructuralLoadStatic);
    DeltaTConstant : OPTIONAL IfcThermodynamicTemperatureMeasure;
    DeltaTY : OPTIONAL IfcThermodynamicTemperatureMeasure;
    DeltaTZ : OPTIONAL IfcThermodynamicTemperatureMeasure;
END_ENTITY;

ENTITY IfcStructuralMember
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcStructuralCurveMember
    , IfcStructuralSurfaceMember))
  SUBTYPE OF (IfcStructuralItem);
  INVERSE
    ConnectedBy : SET [0:?] OF IfcRelConnectsStructuralMember FOR
    RelatingStructuralMember;
END_ENTITY;

ENTITY IfcStructuralPlanarAction
  SUBTYPE OF (IfcStructuralSurfaceAction);
  WHERE
    SuitableLoadType : SIZEOF([' IFC4. IFCSTRUCTURALLOADPLANARFORCE',

```

---

```

' IFC4. IFCSTRUCTURALLOADTEMPERATURE' ]                                     *
TYPEOF(SELF\IfcStructuralActivity.AppliedLoad)) = 1;
    ConstPredefinedType : SELF\IfcStructuralSurfaceAction.PredefinedType =
IfcStructuralSurfaceActivityTypeEnum.CONST;
END_ENTITY;

ENTITY IfcStructuralPointAction
    SUBTYPE OF (IfcStructuralAction);
    WHERE
        SuitableLoadType : SIZEOF([' IFC4. IFCSTRUCTURALLOADSINGLEFORCE',
' IFC4. IFCSTRUCTURALLOADSINGLEDISPLACEMENT' ]                               *
TYPEOF(SELF\IfcStructuralActivity.AppliedLoad)) = 1;
END_ENTITY;

ENTITY IfcStructuralPointConnection
    SUBTYPE OF (IfcStructuralConnection);
    ConditionCoordinateSystem : OPTIONAL IfcAxis2Placement3D;
END_ENTITY;

ENTITY IfcStructuralPointReaction
    SUBTYPE OF (IfcStructuralReaction);
    WHERE
        SuitableLoadType : SIZEOF([' IFC4. IFCSTRUCTURALLOADSINGLEFORCE',
' IFC4. IFCSTRUCTURALLOADSINGLEDISPLACEMENT' ]                               *
TYPEOF(SELF\IfcStructuralActivity.AppliedLoad)) = 1;
END_ENTITY;

ENTITY IfcStructuralReaction
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcStructuralCurveReaction
        , IfcStructuralPointReaction
        , IfcStructuralSurfaceReaction))
    SUBTYPE OF (IfcStructuralActivity);
END_ENTITY;

ENTITY IfcStructuralResultGroup
    SUBTYPE OF (IfcGroup);
    TheoryType : IfcAnalysisTheoryTypeEnum;
    ResultForLoadGroup : OPTIONAL IfcStructuralLoadGroup;
    IsLinear : IfcBoolean;
    INVERSE
        ResultGroupFor : SET [0:1] OF IfcStructuralAnalysisModel FOR HasResults;
    WHERE
        HasObjectType : (TheoryType <> IfcAnalysisTheoryTypeEnum.USERDEFINED) OR

```

---

```

EXISTS(SELF\IfcObject.ObjectType);
END_ENTITY;

ENTITY IfcStructuralSurfaceAction
  SUPERTYPE OF (ONEOF
    (IfcStructuralPlanarAction))
  SUBTYPE OF (IfcStructuralAction);
    ProjectedOrTrue : OPTIONAL IfcProjectedOrTrueLengthEnum;
    PredefinedType : IfcStructuralSurfaceActivityTypeEnum;
  WHERE
    ProjectedIsGlobal : (NOT EXISTS(ProjectedOrTrue)) OR
    ((ProjectedOrTrue <> PROJECTED_LENGTH) OR
    (SELF\IfcStructuralActivity.GlobalOrLocal = GLOBAL_COORDS));
    HasObjectType : (PredefinedType <>
IfcStructuralSurfaceActivityTypeEnum.USERDEFINED) OR
EXISTS(SELF\IfcObject.ObjectType);
END_ENTITY;

ENTITY IfcStructuralSurfaceConnection
  SUBTYPE OF (IfcStructuralConnection);
END_ENTITY;

ENTITY IfcStructuralSurfaceMember
  SUPERTYPE OF (ONEOF
    (IfcStructuralSurfaceMemberVarying))
  SUBTYPE OF (IfcStructuralMember);
    PredefinedType : IfcStructuralSurfaceMemberTypeEnum;
    Thickness : OPTIONAL IfcPositiveLengthMeasure;
  WHERE
    HasObjectType : (PredefinedType <>
IfcStructuralSurfaceMemberTypeEnum.USERDEFINED) OR
EXISTS(SELF\IfcObject.ObjectType);
END_ENTITY;

ENTITY IfcStructuralSurfaceMemberVarying
  SUBTYPE OF (IfcStructuralSurfaceMember);
END_ENTITY;

ENTITY IfcStructuralSurfaceReaction
  SUBTYPE OF (IfcStructuralReaction);
    PredefinedType : IfcStructuralSurfaceActivityTypeEnum;
  WHERE
    HasPredefinedType : (PredefinedType <>
IfcStructuralSurfaceActivityTypeEnum.USERDEFINED) OR

```

---

```
EXISTS (SELF\IfcObject. ObjectType);
END_ENTITY;
```

```
ENTITY IfcStyleModel
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcStyledRepresentation))
  SUBTYPE OF (IfcRepresentation);
END_ENTITY;
```

```
ENTITY IfcStyledItem
  SUBTYPE OF (IfcRepresentationItem);
  Item : OPTIONAL IfcRepresentationItem;
  Styles : SET [1:?] OF IfcStyleAssignmentSelect;
  Name : OPTIONAL IfcLabel;
  WHERE
    ApplicableItem : NOT('IFC4.IFCSTYLEDITEM' IN TYPEOF(Item));
END_ENTITY;
```

```
ENTITY IfcStyledRepresentation
  SUBTYPE OF (IfcStyleModel);
  WHERE
    OnlyStyledItems : SIZEOF(QUERY(temp <* SELF\IfcRepresentation.Items |
      (NOT('IFC4.IFCSTYLEDITEM' IN TYPEOF(temp)))
    )) = 0;
END_ENTITY;
```

```
ENTITY IfcSubContractResource
  SUBTYPE OF (IfcConstructionResource);
  PredefinedType : OPTIONAL IfcSubContractResourceTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
      (PredefinedType <> IfcSubContractResourceTypeEnum.USERDEFINED) OR
      ((PredefinedType = IfcSubContractResourceTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject. ObjectType));
END_ENTITY;
```

```
ENTITY IfcSubContractResourceType
  SUBTYPE OF (IfcConstructionResourceType);
  PredefinedType : IfcSubContractResourceTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <>
      IfcSubContractResourceTypeEnum.USERDEFINED) OR
      ((PredefinedType = IfcSubContractResourceTypeEnum.USERDEFINED) AND
        EXISTS (SELF\IfcTypeResource. ResourceType));
```



---

END\_ENTITY;

ENTITY IfcSubedge  
  SUBTYPE OF (IfcEdge);  
    ParentEdge : IfcEdge;  
END\_ENTITY;

ENTITY IfcSurface  
  ABSTRACT SUPERTYPE OF (ONEOF  
    (IfcBoundedSurface  
      , IfcElementarySurface  
      , IfcSweptSurface))  
  SUBTYPE OF (IfcGeometricRepresentationItem);  
  DERIVE  
    Dim : IfcDimensionCount := 3;  
END\_ENTITY;

ENTITY IfcSurfaceCurve  
  SUPERTYPE OF (ONEOF  
    (IfcIntersectionCurve  
      , IfcSeamCurve))  
  SUBTYPE OF (IfcCurve);  
    Curve3D : IfcCurve;  
    AssociatedGeometry : LIST [1:2] OF IfcPcurve;  
    MasterRepresentation : IfcPreferredSurfaceCurveRepresentation;  
  DERIVE  
    BasisSurface : SET [1:2] OF IfcSurface := IfcGetBasisSurface(SELF);  
  WHERE  
    CurveIs3D : Curve3D.Dim = 3;  
    CurveIsNotPcurve : NOT (' IFC4. IFCPCURVE' IN TYPEOF(Curve3D));  
END\_ENTITY;

ENTITY IfcSurfaceCurveSweptAreaSolid  
  SUBTYPE OF (IfcSweptAreaSolid);  
    Directrix : IfcCurve;  
    StartParam : OPTIONAL IfcParameterValue;  
    EndParam : OPTIONAL IfcParameterValue;  
    ReferenceSurface : IfcSurface;  
  WHERE  
    DirectrixBounded : (EXISTS(StartParam) AND EXISTS(EndParam)) OR  
    (SIZEOF([' IFC4. IFCCONIC', ' IFC4. IFCBOUNDEDCURVE'] \* TYPEOF(Directrix)) = 1);  
END\_ENTITY;

ENTITY IfcSurfaceFeature

---

```

SUBTYPE OF (IfcFeatureElement);
  PredefinedType : OPTIONAL IfcSurfaceFeatureTypeEnum;
WHERE
  HasObjectType : NOT EXISTS(PredefinedType) OR (PredefinedType <>
IfcSurfaceFeatureTypeEnum.USERDEFINED) OR EXISTS(SELF\IfcObject.ObjectType);
END_ENTITY;

ENTITY IfcSurfaceOfLinearExtrusion
SUBTYPE OF (IfcSweptSurface);
  ExtrudedDirection : IfcDirection;
  Depth : IfcLengthMeasure;
DERIVE
  ExtrusionAxis : IfcVector := IfcRepresentationItem() ||
IfcGeometricRepresentationItem () || IfcVector (ExtrudedDirection, Depth);
WHERE
  DepthGreaterZero : Depth > 0.;
END_ENTITY;

ENTITY IfcSurfaceOfRevolution
SUBTYPE OF (IfcSweptSurface);
  AxisPosition : IfcAxis1Placement;
DERIVE
  AxisLine : IfcLine := IfcRepresentationItem() ||
IfcGeometricRepresentationItem () || IfcCurve() || IfcLine(AxisPosition.Location,
  IfcRepresentationItem() || IfcGeometricRepresentationItem () ||
IfcVector(AxisPosition.Z, 1.0));
END_ENTITY;

ENTITY IfcSurfaceReinforcementArea
SUBTYPE OF (IfcStructuralLoadOrResult);
  SurfaceReinforcement1 : OPTIONAL LIST [2:3] OF IfcLengthMeasure;
  SurfaceReinforcement2 : OPTIONAL LIST [2:3] OF IfcLengthMeasure;
  ShearReinforcement : OPTIONAL IfcRatioMeasure;
WHERE
  SurfaceAndOrShearAreaSpecified : EXISTS(SurfaceReinforcement1) OR
EXISTS(SurfaceReinforcement2) OR EXISTS(ShearReinforcement);
  NonnegativeArea1 : (NOT EXISTS(SurfaceReinforcement1)) OR (
  (SurfaceReinforcement1[1] >= 0.) AND
  (SurfaceReinforcement1[2] >= 0.) AND
  ((SIZEOF(SurfaceReinforcement1) = 1) OR (SurfaceReinforcement1[1] >= 0.))
);
  NonnegativeArea2 : (NOT EXISTS(SurfaceReinforcement2)) OR (
  (SurfaceReinforcement2[1] >= 0.) AND
  (SurfaceReinforcement2[2] >= 0.) AND

```

---

```

        ((SIZEOF(SurfaceReinforcement2) = 1) OR (SurfaceReinforcement2[1] >= 0.))
    );
    NonnegativeArea3 : (NOT EXISTS(ShearReinforcement)) OR (ShearReinforcement >=
0.);
END_ENTITY;

```

```

ENTITY IfcSurfaceStyle
    SUBTYPE OF (IfcPresentationStyle);
    Side : IfcSurfaceSide;
    Styles : SET [1:5] OF IfcSurfaceStyleElementSelect;
    WHERE
        MaxOneShading : SIZEOF(QUERY(Style <* SELF.Styles |
' IFC4. IFCSURFACESTYLESHADING' IN
        TYPEOF(Style)
    )) <= 1;
        MaxOneLighting : SIZEOF(QUERY(Style <* SELF.Styles |
' IFC4. IFCSURFACESTYLELIGHTING' IN
        TYPEOF(Style)
    )) <= 1;
        MaxOneRefraction : SIZEOF(QUERY(Style <* SELF.Styles |
' IFC4. IFCSURFACESTYLEREFRACTION' IN
        TYPEOF(Style)
    )) <= 1;
        MaxOneTextures : SIZEOF(QUERY(Style <* SELF.Styles |
' IFC4. IFCSURFACESTYLEWITHTEXTURES' IN
        TYPEOF(Style)
    )) <= 1;
        MaxOneExtDefined : SIZEOF(QUERY(Style <* SELF.Styles |
' IFC4. IFCEXTERNALLYDEFINEDSURFACESTYLE' IN
        TYPEOF(Style)
    )) <= 1;
END_ENTITY;

```

```

ENTITY IfcSurfaceStyleLighting
    SUBTYPE OF (IfcPresentationItem);
    DiffuseTransmissionColour : IfcColourRgb;
    DiffuseReflectionColour : IfcColourRgb;
    TransmissionColour : IfcColourRgb;
    ReflectanceColour : IfcColourRgb;
END_ENTITY;

```

```

ENTITY IfcSurfaceStyleRefraction
    SUBTYPE OF (IfcPresentationItem);
    RefractionIndex : OPTIONAL IfcReal;

```

---

```

        DispersionFactor : OPTIONAL IfcReal;
END_ENTITY;

ENTITY IfcSurfaceStyleRendering
    SUBTYPE OF (IfcSurfaceStyleShading);
        DiffuseColour : OPTIONAL IfcColourOrFactor;
        TransmissionColour : OPTIONAL IfcColourOrFactor;
        DiffuseTransmissionColour : OPTIONAL IfcColourOrFactor;
        ReflectionColour : OPTIONAL IfcColourOrFactor;
        SpecularColour : OPTIONAL IfcColourOrFactor;
        SpecularHighlight : OPTIONAL IfcSpecularHighlightSelect;
        ReflectanceMethod : IfcReflectanceMethodEnum;
END_ENTITY;

ENTITY IfcSurfaceStyleShading
    SUPERTYPE OF (ONEOF
        (IfcSurfaceStyleRendering))
    SUBTYPE OF (IfcPresentationItem);
        SurfaceColour : IfcColourRgb;
        Transparency : OPTIONAL IfcNormalisedRatioMeasure;
END_ENTITY;

ENTITY IfcSurfaceStyleWithTextures
    SUBTYPE OF (IfcPresentationItem);
        Textures : LIST [1:?] OF IfcSurfaceTexture;
END_ENTITY;

ENTITY IfcSurfaceTexture
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcBlobTexture
        , IfcImageTexture
        , IfcPixelTexture))
    SUBTYPE OF (IfcPresentationItem);
        RepeatS : IfcBoolean;
        RepeatT : IfcBoolean;
        Mode : OPTIONAL IfcIdentifier;
        TextureTransform : OPTIONAL IfcCartesianTransformationOperator2D;
        Parameter : OPTIONAL LIST [1:?] OF IfcIdentifier;
    INVERSE
        IsMappedBy : SET [0:?] OF IfcTextureCoordinate FOR Maps;
        UsedInStyles : SET [0:?] OF IfcSurfaceStyleWithTextures FOR Textures;
END_ENTITY;

ENTITY IfcSweptAreaSolid

```

---

```

ABSTRACT SUPERTYPE OF (ONEOF
    (IfcExtrudedAreaSolid
    , IfcFixedReferenceSweptAreaSolid
    , IfcRevolvedAreaSolid
    , IfcSurfaceCurveSweptAreaSolid))
SUBTYPE OF (IfcSolidModel);
    SweptArea : IfcProfileDef;
    Position : OPTIONAL IfcAxis2Placement3D;
WHERE
    SweptAreaType : SweptArea.ProfileType = IfcProfileTypeEnum.Area;
END_ENTITY;

ENTITY IfcSweptDiskSolid
    SUPERTYPE OF (ONEOF
        (IfcSweptDiskSolidPolygonal))
    SUBTYPE OF (IfcSolidModel);
        Directrix : IfcCurve;
        Radius : IfcPositiveLengthMeasure;
        InnerRadius : OPTIONAL IfcPositiveLengthMeasure;
        StartParam : OPTIONAL IfcParameterValue;
        EndParam : OPTIONAL IfcParameterValue;
WHERE
    DirectrixDim : Directrix.Dim = 3;
    InnerRadiusSize : (NOT EXISTS(InnerRadius)) OR (Radius > InnerRadius);
    DirectrixBounded : (EXISTS(StartParam) AND EXISTS(EndParam)) OR
    (SIZEOF([' IFC4. IFCCONIC', ' IFC4. IFCBOUNDED_CURVE'] * TYPEOF(Directrix)) = 1);
END_ENTITY;

ENTITY IfcSweptDiskSolidPolygonal
    SUBTYPE OF (IfcSweptDiskSolid);
        FilletRadius : OPTIONAL IfcPositiveLengthMeasure;
WHERE
    CorrectRadii : NOT(EXISTS(FilletRadius)) OR (FilletRadius >=
SELF\IfcSweptDiskSolid.Radius);
    DirectrixIsPolyline : (' IFC4. IFCPOLYLINE' IN
TYPEOF(SELF\IfcSweptDiskSolid.Directrix)) OR
((' IFC4. IFCINDEXEDPOLYCURVE' IN TYPEOF(SELF\IfcSweptDiskSolid.Directrix)) AND
NOT(EXISTS(SELF\IfcSweptDiskSolid.Directrix.Segments)));
END_ENTITY;

ENTITY IfcSweptSurface
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcSurfaceOfLinearExtrusion
        , IfcSurfaceOfRevolution))

```

---

```

SUBTYPE OF (IfcSurface);
    SweptCurve : IfcProfileDef;
    Position : OPTIONAL IfcAxis2Placement3D;
WHERE
    SweptCurveType : SweptCurve.ProfileType = IfcProfileTypeEnum.Curve;
END_ENTITY;

ENTITY IfcSwitchingDevice
SUBTYPE OF (IfcFlowController);
    PredefinedType : OPTIONAL IfcSwitchingDeviceTypeEnum;
WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcSwitchingDeviceTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcSwitchingDeviceTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    (' IFC4. IFCSWITCHINGDEVICETYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcSwitchingDeviceType
SUBTYPE OF (IfcFlowControllerType);
    PredefinedType : IfcSwitchingDeviceTypeEnum;
WHERE
    CorrectPredefinedType : (PredefinedType <>
    IfcSwitchingDeviceTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcSwitchingDeviceTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcSystem
SUPERTYPE OF (ONEOF
    (IfcBuildingSystem
    , IfcDistributionSystem
    , IfcStructuralAnalysisModel
    , IfcZone))
SUBTYPE OF (IfcGroup);
INVERSE
    ServicesBuildings : SET [0:1] OF IfcRelServicesBuildings FOR RelatingSystem;
END_ENTITY;

ENTITY IfcSystemFurnitureElement
SUBTYPE OF (IfcFurnishingElement);
    PredefinedType : OPTIONAL IfcSystemFurnitureElementTypeEnum;

```

---

```

WHERE
    CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
    (PredefinedType <> IfcSystemFurnitureElementTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcSystemFurnitureElementTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    (' IFC4. IFCSYSTEMFURNITUREELEMENTTYPE' IN
    TYPEOF (SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcSystemFurnitureElementType
    SUBTYPE OF (IfcFurnishingElementType);
    PredefinedType : OPTIONAL IfcSystemFurnitureElementTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <>
        IfcSystemFurnitureElementTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcSystemFurnitureElementTypeEnum.USERDEFINED) AND
        EXISTS (SELF\IfcElementType.ElementType));
    END_ENTITY;

```

```

ENTITY IfcTShapeProfileDef
    SUBTYPE OF (IfcParameterizedProfileDef);
    Depth : IfcPositiveLengthMeasure;
    FlangeWidth : IfcPositiveLengthMeasure;
    WebThickness : IfcPositiveLengthMeasure;
    FlangeThickness : IfcPositiveLengthMeasure;
    FilletRadius : OPTIONAL IfcNonNegativeLengthMeasure;
    FlangeEdgeRadius : OPTIONAL IfcNonNegativeLengthMeasure;
    WebEdgeRadius : OPTIONAL IfcNonNegativeLengthMeasure;
    WebSlope : OPTIONAL IfcPlaneAngleMeasure;
    FlangeSlope : OPTIONAL IfcPlaneAngleMeasure;
    WHERE
        ValidFlangeThickness : FlangeThickness < Depth;
        ValidWebThickness : WebThickness < FlangeWidth;
    END_ENTITY;

```

```

ENTITY IfcTable;
    Name : OPTIONAL IfcLabel;
    Rows : OPTIONAL LIST [1:?] OF IfcTableRow;
    Columns : OPTIONAL LIST [1:?] OF IfcTableColumn;
    DERIVE
        NumberOfCellsInRow : IfcInteger := HIINDEX (Rows[1].RowCells);
        NumberOfHeadings : IfcInteger := SIZEOF (QUERY (Temp <* Rows | Temp.IsHeading));
        NumberOfDataRows : IfcInteger := SIZEOF (QUERY (Temp <* Rows |

```

---

```

NOT(Temp.IsHeading)));
WHERE
    WR1 : SIZEOF(QUERY( Temp <* Rows | HIINDEX(Temp.RowCells) <>
HIINDEX(Rows[1].RowCells))) = 0;
    WR2 : { 0 <= NumberOfHeadings <= 1 };
END_ENTITY;

ENTITY IfcTableColumn;
    Identifier : OPTIONAL IfcIdentifier;
    Name : OPTIONAL IfcLabel;
    Description : OPTIONAL IfcText;
    Unit : OPTIONAL IfcUnit;
    ReferencePath : OPTIONAL IfcReference;
END_ENTITY;

ENTITY IfcTableRow;
    RowCells : OPTIONAL LIST [1:?] OF IfcValue;
    IsHeading : OPTIONAL IfcBoolean;
END_ENTITY;

ENTITY IfcTank
    SUBTYPE OF (IfcFlowStorageDevice);
    PredefinedType : OPTIONAL IfcTankTypeEnum;
    WHERE
        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
        (PredefinedType <> IfcTankTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcTankTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCTANKTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcTankType
    SUBTYPE OF (IfcFlowStorageDeviceType);
    PredefinedType : IfcTankTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <> IfcTankTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcTankTypeEnum.USERDEFINED) AND
        EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcTask
    SUBTYPE OF (IfcProcess);
    Status : OPTIONAL IfcLabel;

```



---

```

    WorkMethod : OPTIONAL IfcLabel;
    IsMilestone : IfcBoolean;
    Priority : OPTIONAL IfcInteger;
    TaskTime : OPTIONAL IfcTaskTime;
    PredefinedType : OPTIONAL IfcTaskTypeEnum;
WHERE
    HasName : EXISTS(SELF\IfcRoot.Name);
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR (PredefinedType <>
IfcTaskTypeEnum.USERDEFINED) OR ((PredefinedType = IfcTaskTypeEnum.USERDEFINED)
AND EXISTS(SELF\IfcObject.ObjectType));
END_ENTITY;

```

```

ENTITY IfcTaskTime
    SUPERTYPE OF (ONEOF
        (IfcTaskTimeRecurring))
    SUBTYPE OF (IfcSchedulingTime);
    DurationType : OPTIONAL IfcTaskDurationEnum;
    ScheduleDuration : OPTIONAL IfcDuration;
    ScheduleStart : OPTIONAL IfcDateTime;
    ScheduleFinish : OPTIONAL IfcDateTime;
    EarlyStart : OPTIONAL IfcDateTime;
    EarlyFinish : OPTIONAL IfcDateTime;
    LateStart : OPTIONAL IfcDateTime;
    LateFinish : OPTIONAL IfcDateTime;
    FreeFloat : OPTIONAL IfcDuration;
    TotalFloat : OPTIONAL IfcDuration;
    IsCritical : OPTIONAL IfcBoolean;
    StatusTime : OPTIONAL IfcDateTime;
    ActualDuration : OPTIONAL IfcDuration;
    ActualStart : OPTIONAL IfcDateTime;
    ActualFinish : OPTIONAL IfcDateTime;
    RemainingTime : OPTIONAL IfcDuration;
    Completion : OPTIONAL IfcPositiveRatioMeasure;
END_ENTITY;

```

```

ENTITY IfcTaskTimeRecurring
    SUBTYPE OF (IfcTaskTime);
    Recurrence : IfcRecurrencePattern;
END_ENTITY;

```

```

ENTITY IfcTaskType
    SUBTYPE OF (IfcTypeProcess);
    PredefinedType : IfcTaskTypeEnum;
    WorkMethod : OPTIONAL IfcLabel;

```

---

```

WHERE
    CorrectPredefinedType : (PredefinedType <> IfcTaskTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcTaskTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcTypeProcess.ProcessType));
END_ENTITY;

```

```

ENTITY IfcTelecomAddress
    SUBTYPE OF (IfcAddress);
    TelephoneNumbers : OPTIONAL LIST [1:?] OF IfcLabel;
    FacsimileNumbers : OPTIONAL LIST [1:?] OF IfcLabel;
    PagerNumber : OPTIONAL IfcLabel;
    ElectronicMailAddresses : OPTIONAL LIST [1:?] OF IfcLabel;
    WWWHomePageURL : OPTIONAL IfcURIReference;
    MessagingIDs : OPTIONAL LIST [1:?] OF IfcURIReference;
    WHERE
        MinimumDataProvided : EXISTS (TelephoneNumbers) OR
        EXISTS (FacsimileNumbers) OR
        EXISTS (PagerNumber) OR
        EXISTS (ElectronicMailAddresses) OR
        EXISTS (WWWHomePageURL) OR
        EXISTS (MessagingIDs);
    END_ENTITY;

```

```

ENTITY IfcTendon
    SUBTYPE OF (IfcReinforcingElement);
    PredefinedType : OPTIONAL IfcTendonTypeEnum;
    NominalDiameter : OPTIONAL IfcPositiveLengthMeasure;
    CrossSectionArea : OPTIONAL IfcAreaMeasure;
    TensionForce : OPTIONAL IfcForceMeasure;
    PreStress : OPTIONAL IfcPressureMeasure;
    FrictionCoefficient : OPTIONAL IfcNormalisedRatioMeasure;
    AnchorageSlip : OPTIONAL IfcPositiveLengthMeasure;
    MinCurvatureRadius : OPTIONAL IfcPositiveLengthMeasure;
    WHERE
        CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
        (PredefinedType <> IfcTendonTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcTendonTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject.ObjectType));
        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
        ('IFC4.IFCTENDONTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
    END_ENTITY;

```

```

ENTITY IfcTendonAnchor
    SUBTYPE OF (IfcReinforcingElement);

```

---

```

    PredefinedType : OPTIONAL IfcTendonAnchorTypeEnum;
WHERE
    CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
    (PredefinedType <> IfcTendonAnchorTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcTendonAnchorTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCTENDONANCHORTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcTendonAnchorType
    SUBTYPE OF (IfcReinforcingElementType);
    PredefinedType : IfcTendonAnchorTypeEnum;
WHERE
    CorrectPredefinedType : (PredefinedType <>
    IfcTendonAnchorTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcTendonAnchorTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcTendonType
    SUBTYPE OF (IfcReinforcingElementType);
    PredefinedType : IfcTendonTypeEnum;
    NominalDiameter : OPTIONAL IfcPositiveLengthMeasure;
    CrossSectionArea : OPTIONAL IfcAreaMeasure;
    SheathDiameter : OPTIONAL IfcPositiveLengthMeasure;
WHERE
    CorrectPredefinedType : (PredefinedType <> IfcTendonTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcTendonTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcTessellatedFaceSet
    ABSTRACT SUPERTYPE OF (ONEOF
        (IfcPolygonalFaceSet
        , IfcTriangulatedFaceSet))
    SUBTYPE OF (IfcTessellatedItem);
    Coordinates : IfcCartesianPointList3D;
DERIVE
    Dim : IfcDimensionCount := 3;
INVERSE
    HasColours : SET [0:1] OF IfcIndexedColourMap FOR MappedTo;
    HasTextures : SET [0:?] OF IfcIndexedTextureMap FOR MappedTo;

```

---

END\_ENTITY;

ENTITY IfcTessellatedItem  
ABSTRACT SUPERTYPE OF (ONEOF  
    (IfcIndexedPolygonalFace  
      , IfcTessellatedFaceSet))  
SUBTYPE OF (IfcGeometricRepresentationItem);  
END\_ENTITY;

ENTITY IfcTextLiteral  
SUPERTYPE OF (ONEOF  
    (IfcTextLiteralWithExtent))  
SUBTYPE OF (IfcGeometricRepresentationItem);  
    Literal : IfcPresentableText;  
    Placement : IfcAxis2Placement;  
    Path : IfcTextPath;  
END\_ENTITY;

ENTITY IfcTextLiteralWithExtent  
SUBTYPE OF (IfcTextLiteral);  
    Extent : IfcPlanarExtent;  
    BoxAlignment : IfcBoxAlignment;  
WHERE  
    WR31 : NOT(' IFC4. IFCPLANARBOX' IN TYPEOF(Extent));  
END\_ENTITY;

ENTITY IfcTextStyle  
SUBTYPE OF (IfcPresentationStyle);  
    TextCharacterAppearance : OPTIONAL IfcTextStyleForDefinedFont;  
    TextStyle : OPTIONAL IfcTextStyleTextModel;  
    TextFontStyle : IfcTextFontSelect;  
    ModelOrDraughting : OPTIONAL IfcBoolean;  
END\_ENTITY;

ENTITY IfcTextStyleFontModel  
SUBTYPE OF (IfcPreDefinedTextFont);  
    FontFamily : LIST [1:?] OF IfcTextFontName;  
    FontStyle : OPTIONAL IfcFontStyle;  
    FontVariant : OPTIONAL IfcFontVariant;  
    FontWeight : OPTIONAL IfcFontWeight;  
    FontSize : IfcSizeSelect;  
WHERE  
    MeasureOfFontSize : (' IFC4. IFCLENGTHMEASURE' IN TYPEOF(SELF.FontSize)) AND  
    (SELF.FontSize > 0.);

---

END\_ENTITY;

ENTITY IfcTextStyleForDefinedFont  
 SUBTYPE OF (IfcPresentationItem);  
 Colour : IfcColour;  
 BackgroundColour : OPTIONAL IfcColour;  
END\_ENTITY;

ENTITY IfcTextStyleTextModel  
 SUBTYPE OF (IfcPresentationItem);  
 TextIndent : OPTIONAL IfcSizeSelect;  
 TextAlign : OPTIONAL IfcTextAlignment;  
 TextDecoration : OPTIONAL IfcTextDecoration;  
 LetterSpacing : OPTIONAL IfcSizeSelect;  
 WordSpacing : OPTIONAL IfcSizeSelect;  
 TextTransform : OPTIONAL IfcTextTransformation;  
 LineHeight : OPTIONAL IfcSizeSelect;  
END\_ENTITY;

ENTITY IfcTextureCoordinate  
 ABSTRACT SUPERTYPE OF (ONEOF  
 (IfcIndexedTextureMap  
 , IfcTextureCoordinateGenerator  
 , IfcTextureMap))  
 SUBTYPE OF (IfcPresentationItem);  
 Maps : LIST [1:?] OF IfcSurfaceTexture;  
END\_ENTITY;

ENTITY IfcTextureCoordinateGenerator  
 SUBTYPE OF (IfcTextureCoordinate);  
 Mode : IfcLabel;  
 Parameter : OPTIONAL LIST [1:?] OF IfcReal;  
END\_ENTITY;

ENTITY IfcTextureMap  
 SUBTYPE OF (IfcTextureCoordinate);  
 Vertices : LIST [3:?] OF IfcTextureVertex;  
 MappedTo : IfcFace;  
END\_ENTITY;

ENTITY IfcTextureVertex  
 SUBTYPE OF (IfcPresentationItem);  
 Coordinates : LIST [2:2] OF IfcParameterValue;  
END\_ENTITY;

---

```

ENTITY IfcTextureVertexList
  SUBTYPE OF (IfcPresentationItem);
    TexCoordsList : LIST [1:?] OF LIST [2:2] OF IfcParameterValue;
END_ENTITY;

ENTITY IfcTimePeriod;
  StartTime : IfcTime;
  EndTime : IfcTime;
END_ENTITY;

ENTITY IfcTimeSeries
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcIrregularTimeSeries
    , IfcRegularTimeSeries));
  Name : IfcLabel;
  Description : OPTIONAL IfcText;
  StartTime : IfcDateTime;
  EndTime : IfcDateTime;
  TimeSeriesDataType : IfcTimeSeriesDataTypeEnum;
  DataOrigin : IfcDataOriginEnum;
  UserDefinedDataOrigin : OPTIONAL IfcLabel;
  Unit : OPTIONAL IfcUnit;
  INVERSE
    HasExternalReference : SET [1:?] OF IfcExternalReferenceRelationship FOR
RelatedResourceObjects;
END_ENTITY;

ENTITY IfcTimeSeriesValue;
  ListValues : LIST [1:?] OF IfcValue;
END_ENTITY;

ENTITY IfcTopologicalRepresentationItem
  ABSTRACT SUPERTYPE OF (ONEOF
    (IfcConnectedFaceSet
    , IfcEdge
    , IfcFace
    , IfcFaceBound
    , IfcLoop
    , IfcPath
    , IfcVertex))
  SUBTYPE OF (IfcRepresentationItem);
END_ENTITY;

```

---

```

ENTITY IfcTopologyRepresentation
  SUBTYPE OF (IfcShapeModel);
  WHERE
    WR21 : SIZEOF(QUERY(temp <* SELF\IfcRepresentation.Items |
      NOT(' IFC4. IFCTOPOLOGICALREPRESENTATIONITEM' IN TYPEOF(temp))
    )) = 0;
    WR22 : EXISTS(SELF\IfcRepresentation.RepresentationType);
    WR23 :
IfcTopologyRepresentationTypes(SELF\IfcRepresentation.RepresentationType,
SELF\IfcRepresentation.Items);
END_ENTITY;

ENTITY IfcToroidalSurface
  SUBTYPE OF (IfcElementarySurface);
  MajorRadius : IfcPositiveLengthMeasure;
  MinorRadius : IfcPositiveLengthMeasure;
  WHERE
    MajorLargerMinor : MinorRadius < MajorRadius;
END_ENTITY;

ENTITY IfcTransformer
  SUBTYPE OF (IfcEnergyConversionDevice);
  PredefinedType : OPTIONAL IfcTransformerTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcTransformerTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcTransformerTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    (' IFC4. IFCTRANFORMERTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcTransformerType
  SUBTYPE OF (IfcEnergyConversionDeviceType);
  PredefinedType : IfcTransformerTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcTransformerTypeEnum.USERDEFINED)
OR
    ((PredefinedType = IfcTransformerTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcTransportElement

```

---

```

SUBTYPE OF (IfcElement);
    PredefinedType : OPTIONAL IfcTransportElementTypeEnum;
WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcTransportElementTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcTransportElementTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCTRANSPORTELEMENTTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcTransportElementType
    SUBTYPE OF (IfcElementType);
    PredefinedType : IfcTransportElementTypeEnum;
WHERE
    CorrectPredefinedType : (PredefinedType <>
    IfcTransportElementTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcTransportElementTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcTrapeziumProfileDef
    SUBTYPE OF (IfcParameterizedProfileDef);
    BottomXDim : IfcPositiveLengthMeasure;
    TopXDim : IfcPositiveLengthMeasure;
    YDim : IfcPositiveLengthMeasure;
    TopXOffset : IfcLengthMeasure;
END_ENTITY;

```

```

ENTITY IfcTriangulatedFaceSet
    SUBTYPE OF (IfcTessellatedFaceSet);
    Normals : OPTIONAL LIST [1:?] OF LIST [3:3] OF IfcParameterValue;
    Closed : OPTIONAL IfcBoolean;
    CoordIndex : LIST [1:?] OF LIST [3:3] OF IfcPositiveInteger;
    PnIndex : OPTIONAL LIST [1:?] OF IfcPositiveInteger;
    DERIVE
        NumberOfTriangles : IfcInteger := SIZEOF(CoordIndex);
END_ENTITY;

```

```

ENTITY IfcTrimmedCurve
    SUBTYPE OF (IfcBoundedCurve);
    BasisCurve : IfcCurve;
    Trim1 : SET [1:2] OF IfcTrimmingSelect;

```



---

```

    Trim2 : SET [1:2] OF IfcTrimmingSelect;
    SenseAgreement : IfcBoolean;
    MasterRepresentation : IfcTrimmingPreference;
WHERE
    Trim1ValuesConsistent : (HIINDEX(Trim1) = 1) OR (TYPEOF(Trim1[1]) <>
    TYPEOF(Trim1[2]));
    Trim2ValuesConsistent : (HIINDEX(Trim2) = 1) OR (TYPEOF(Trim2[1]) <>
    TYPEOF(Trim2[2]));
    NoTrimOfBoundedCurves : NOT(' IFC4. IFCBOUNDEDCURVE' IN TYPEOF(BasisCurve));
END_ENTITY;

```

```

ENTITY IfcTubeBundle
    SUBTYPE OF (IfcEnergyConversionDevice);
    PredefinedType : OPTIONAL IfcTubeBundleTypeEnum;
WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcTubeBundleTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcTubeBundleTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    (' IFC4. IFCTUBEBundleType' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcTubeBundleType
    SUBTYPE OF (IfcEnergyConversionDeviceType);
    PredefinedType : IfcTubeBundleTypeEnum;
WHERE
    CorrectPredefinedType : (PredefinedType <> IfcTubeBundleTypeEnum.USERDEFINED)
OR
    ((PredefinedType = IfcTubeBundleTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcTypeObject
    SUPERTYPE OF (ONEOF
        (IfcTypeProcess
        , IfcTypeProduct
        , IfcTypeResource))
    SUBTYPE OF (IfcObjectDefinition);
    ApplicableOccurrence : OPTIONAL IfcIdentifier;
    HasPropertySets : OPTIONAL SET [1:?] OF IfcPropertySetDefinition;
INVERSE
    Types : SET [0:1] OF IfcRelDefinesByType FOR RelatingType;

```

---

```

WHERE
    NameRequired : EXISTS (SELF\IfcRoot.Name);
    UniquePropertySetNames      :      (NOT (EXISTS (HasPropertySets)))      OR
IfcUniquePropertySetNames (HasPropertySets);
END_ENTITY;

```

```

ENTITY IfcTypeProcess
ABSTRACT SUPERTYPE OF (ONEOF
    (IfcEventType
    , IfcProcedureType
    , IfcTaskType))
SUBTYPE OF (IfcTypeObject);
    Identification : OPTIONAL IfcIdentifier;
    LongDescription : OPTIONAL IfcText;
    ProcessType : OPTIONAL IfcLabel;
INVERSE
    OperatesOn : SET [0:?] OF IfcRelAssignsToProcess FOR RelatingProcess;
END_ENTITY;

```

```

ENTITY IfcTypeProduct
SUPERTYPE OF (ONEOF
    (IfcDoorStyle
    , IfcElementType
    , IfcSpatialElementType
    , IfcWindowStyle))
SUBTYPE OF (IfcTypeObject);
    RepresentationMaps : OPTIONAL LIST [1:?] OF UNIQUE IfcRepresentationMap;
    Tag : OPTIONAL IfcLabel;
INVERSE
    ReferencedBy : SET [0:?] OF IfcRelAssignsToProduct FOR RelatingProduct;
WHERE
    ApplicableOccurrence : NOT (EXISTS (SELF\IfcTypeObject.Types[1])) OR
(SIZEOF(QUERY(temp <* SELF\IfcTypeObject.Types[1].RelatedObjects |
    NOT(' IFC4. IFCPRODUCT' IN TYPEOF(temp)))
) = 0);
END_ENTITY;

```

```

ENTITY IfcTypeResource
ABSTRACT SUPERTYPE OF (ONEOF
    (IfcConstructionResourceType))
SUBTYPE OF (IfcTypeObject);
    Identification : OPTIONAL IfcIdentifier;
    LongDescription : OPTIONAL IfcText;
    ResourceType : OPTIONAL IfcLabel;

```

---

```

INVERSE
    ResourceOf : SET [0:?] OF IfcRelAssignsToResource FOR RelatingResource;
END_ENTITY;

ENTITY IfcUShapeProfileDef
    SUBTYPE OF (IfcParameterizedProfileDef);
    Depth : IfcPositiveLengthMeasure;
    FlangeWidth : IfcPositiveLengthMeasure;
    WebThickness : IfcPositiveLengthMeasure;
    FlangeThickness : IfcPositiveLengthMeasure;
    FilletRadius : OPTIONAL IfcNonNegativeLengthMeasure;
    EdgeRadius : OPTIONAL IfcNonNegativeLengthMeasure;
    FlangeSlope : OPTIONAL IfcPlaneAngleMeasure;
    WHERE
        ValidFlangeThickness : FlangeThickness < (Depth / 2.);
        ValidWebThickness : WebThickness < FlangeWidth;
END_ENTITY;

ENTITY IfcUnitAssignment;
    Units : SET [1:?] OF IfcUnit;
    WHERE
        WR01 : IfcCorrectUnitAssignment(Units);
END_ENTITY;

ENTITY IfcUnitaryControlElement
    SUBTYPE OF (IfcDistributionControlElement);
    PredefinedType : OPTIONAL IfcUnitaryControlElementTypeEnum;
    WHERE
        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
        (PredefinedType <> IfcUnitaryControlElementTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcUnitaryControlElementTypeEnum.USERDEFINED) AND EXISTS
        (SELF\IfcObject.ObjectType));
        CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
        (' IFC4. IFCUNITARYCONTROLELEMENTTYPE' IN
        TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcUnitaryControlElementType
    SUBTYPE OF (IfcDistributionControlElementType);
    PredefinedType : IfcUnitaryControlElementTypeEnum;
    WHERE
        CorrectPredefinedType : (PredefinedType <>
        IfcUnitaryControlElementTypeEnum.USERDEFINED) OR
        ((PredefinedType = IfcUnitaryControlElementTypeEnum.USERDEFINED) AND

```

---

```

EXISTS (SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcUnitaryEquipment
  SUBTYPE OF (IfcEnergyConversionDevice);
    PredefinedType : OPTIONAL IfcUnitaryEquipmentTypeEnum;
  WHERE
    CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
    (PredefinedType <> IfcUnitaryEquipmentTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcUnitaryEquipmentTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    (' IFC4. IFCUNITARYEQUIPMENTTYPE' IN
    TYPEOF (SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcUnitaryEquipmentType
  SUBTYPE OF (IfcEnergyConversionDeviceType);
    PredefinedType : IfcUnitaryEquipmentTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <>
    IfcUnitaryEquipmentTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcUnitaryEquipmentTypeEnum.USERDEFINED) AND
    EXISTS (SELF\IfcElementType.ElementType));
END_ENTITY;

```

```

ENTITY IfcValve
  SUBTYPE OF (IfcFlowController);
    PredefinedType : OPTIONAL IfcValveTypeEnum;
  WHERE
    CorrectPredefinedType : NOT (EXISTS (PredefinedType)) OR
    (PredefinedType <> IfcValveTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcValveTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    (' IFC4. IFCVALVETYPE' IN TYPEOF (SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

```

```

ENTITY IfcValveType
  SUBTYPE OF (IfcFlowControllerType);
    PredefinedType : IfcValveTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcValveTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcValveTypeEnum.USERDEFINED) AND

```

---

```
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;
```

```
ENTITY IfcVector
  SUBTYPE OF (IfcGeometricRepresentationItem);
    Orientation : IfcDirection;
    Magnitude : IfcLengthMeasure;
  DERIVE
    Dim : IfcDimensionCount := Orientation.Dim;
  WHERE
    MagGreaterOrEqualZero : Magnitude >= 0.0;
END_ENTITY;
```

```
ENTITY IfcVertex
  SUPERTYPE OF (ONEOF
    (IfcVertexPoint))
  SUBTYPE OF (IfcTopologicalRepresentationItem);
END_ENTITY;
```

```
ENTITY IfcVertexLoop
  SUBTYPE OF (IfcLoop);
    LoopVertex : IfcVertex;
END_ENTITY;
```

```
ENTITY IfcVertexPoint
  SUBTYPE OF (IfcVertex);
    VertexGeometry : IfcPoint;
END_ENTITY;
```

```
ENTITY IfcVibrationIsolator
  SUBTYPE OF (IfcElementComponent);
    PredefinedType : OPTIONAL IfcVibrationIsolatorTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcVibrationIsolatorTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcVibrationIsolatorTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    ('IFC4.IFCVIBRATIONISOLATORTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;
```

```
ENTITY IfcVibrationIsolatorType
  SUBTYPE OF (IfcElementComponentType);
```

---

```

        PredefinedType : IfcVibrationIsolatorTypeEnum;
WHERE
        CorrectPredefinedType          :          (PredefinedType          <>
IfcVibrationIsolatorTypeEnum.USERDEFINED) OR
((PredefinedType          =          IfcVibrationIsolatorTypeEnum.USERDEFINED)          AND
EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcVirtualElement
    SUBTYPE OF (IfcElement);
END_ENTITY;

ENTITY IfcVirtualGridIntersection;
    IntersectingAxes : LIST [2:2] OF UNIQUE IfcGridAxis;
    OffsetDistances : LIST [2:3] OF IfcLengthMeasure;
END_ENTITY;

ENTITY IfcVoidingFeature
    SUBTYPE OF (IfcFeatureElementSubtraction);
    PredefinedType : OPTIONAL IfcVoidingFeatureTypeEnum;
WHERE
    HasObjectType : NOT EXISTS(PredefinedType) OR (PredefinedType <>
IfcVoidingFeatureTypeEnum.USERDEFINED) OR EXISTS(SELF\IfcObject.ObjectType);
END_ENTITY;

ENTITY IfcWall
    SUPERTYPE OF (ONEOF
        (IfcWallElementedCase
        , IfcWallStandardCase))
    SUBTYPE OF (IfcBuildingElement);
    PredefinedType : OPTIONAL IfcWallTypeEnum;
WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
(PredefinedType <> IfcWallTypeEnum.USERDEFINED) OR
((PredefinedType          =          IfcWallTypeEnum.USERDEFINED)          AND          EXISTS
(SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
(' IFC4. IFCWALLTYPE' IN TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcWallElementedCase
    SUBTYPE OF (IfcWall);
WHERE
    HasDecomposition : HIINDEX(SELF\IfcObjectDefinition.IsDecomposedBy) > 0;

```

---

```

END_ENTITY;

ENTITY IfcWallStandardCase
  SUBTYPE OF (IfcWall);
  WHERE
    HasMaterialLayerSetUsage : SIZEOF (QUERY(temp < * USEDIN(SELF,
' IFC4. IFCRELAASSOCIATES.RELATEDOBJECTS' ) |
      (' IFC4. IFCRELAASSOCIATESMATERIAL' IN TYPEOF(temp)) AND
      (' IFC4. IFCMATERIALLAYERSETUSAGE' IN TYPEOF(temp.RelatingMaterial))
    )) = 1;
END_ENTITY;

ENTITY IfcWallType
  SUBTYPE OF (IfcBuildingElementType);
  PredefinedType : IfcWallTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <> IfcWallTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcWallTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcWasteTerminal
  SUBTYPE OF (IfcFlowTerminal);
  PredefinedType : OPTIONAL IfcWasteTerminalTypeEnum;
  WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR
    (PredefinedType <> IfcWasteTerminalTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcWasteTerminalTypeEnum.USERDEFINED) AND EXISTS
    (SELF\IfcObject.ObjectType));
    CorrectTypeAssigned : (SIZEOF(IsTypedBy) = 0) OR
    (' IFC4. IFCWASTETERMINALTYPE' IN
    TYPEOF(SELF\IfcObject.IsTypedBy[1].RelatingType));
END_ENTITY;

ENTITY IfcWasteTerminalType
  SUBTYPE OF (IfcFlowTerminalType);
  PredefinedType : IfcWasteTerminalTypeEnum;
  WHERE
    CorrectPredefinedType : (PredefinedType <>
    IfcWasteTerminalTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcWasteTerminalTypeEnum.USERDEFINED) AND
    EXISTS(SELF\IfcElementType.ElementType));
END_ENTITY;

```

---

```

ENTITY IfcWindow
  SUPERTYPE OF (ONEOF
    (IfcWindowStandardCase))
  SUBTYPE OF (IfcBuildingElement);
    OverallHeight : OPTIONAL IfcPositiveLengthMeasure;
    OverallWidth : OPTIONAL IfcPositiveLengthMeasure;
    PredefinedType : OPTIONAL IfcWindowTypeEnum;
    PartitioningType : OPTIONAL IfcWindowTypePartitioningEnum;
    UserDefinedPartitioningType : OPTIONAL IfcLabel;
  WHERE
    CorrectStyleAssigned : (SIZEOF(IsTypedBy) = 0)
OR (' IFC4. IFCWINDOWTYPE' IN TYPEOF(SELf\IfcObject. IsTypedBy[1]. RelatingType));
END_ENTITY;

```

```

ENTITY IfcWindowLiningProperties
  SUBTYPE OF (IfcPreDefinedPropertySet);
    LiningDepth : OPTIONAL IfcPositiveLengthMeasure;
    LiningThickness : OPTIONAL IfcNonNegativeLengthMeasure;
    TransomThickness : OPTIONAL IfcNonNegativeLengthMeasure;
    MullionThickness : OPTIONAL IfcNonNegativeLengthMeasure;
    FirstTransomOffset : OPTIONAL IfcNormalisedRatioMeasure;
    SecondTransomOffset : OPTIONAL IfcNormalisedRatioMeasure;
    FirstMullionOffset : OPTIONAL IfcNormalisedRatioMeasure;
    SecondMullionOffset : OPTIONAL IfcNormalisedRatioMeasure;
    ShapeAspectStyle : OPTIONAL IfcShapeAspect;
    LiningOffset : OPTIONAL IfcLengthMeasure;
    LiningToPanelOffsetX : OPTIONAL IfcLengthMeasure;
    LiningToPanelOffsetY : OPTIONAL IfcLengthMeasure;
  WHERE
    WR31 : NOT (EXISTS(LiningDepth) AND NOT (EXISTS(LiningThickness)));
    WR32 : NOT (NOT (EXISTS(FirstTransomOffset)) AND EXISTS(SecondTransomOffset));
    WR33 : NOT (NOT (EXISTS(FirstMullionOffset)) AND EXISTS(SecondMullionOffset));
    WR34 : (EXISTS(SELf\IfcPropertySetDefinition. DefinesType[1]))
AND
(
  (' IFC4. IFCWINDOWTYPE' IN TYPEOF(SELf\IfcPropertySetDefinition. DefinesType[1]))
  OR
  (' IFC4. IFCWINDOWSTYLE' IN TYPEOF(SELf\IfcPropertySetDefinition. DefinesType[1]))
);
END_ENTITY;

```

```

ENTITY IfcWindowPanelProperties
  SUBTYPE OF (IfcPreDefinedPropertySet);
    OperationType : IfcWindowPanelOperationEnum;

```



---

```

    PanelPosition : IfcWindowPanelPositionEnum;
    FrameDepth : OPTIONAL IfcPositiveLengthMeasure;
    FrameThickness : OPTIONAL IfcPositiveLengthMeasure;
    ShapeAspectStyle : OPTIONAL IfcShapeAspect;
WHERE
    ApplicableToType : (EXISTS (SELF\IfcPropertySetDefinition.DefinesType[1]))
AND
(
    (' IFC4. IFCWINDOWTYPE' IN TYPEOF (SELF\IfcPropertySetDefinition.DefinesType[1]))
    OR
    (' IFC4. IFCWINDOWSTYLE' IN TYPEOF (SELF\IfcPropertySetDefinition.DefinesType[1]))
);
END_ENTITY;

ENTITY IfcWindowStandardCase
    SUBTYPE OF (IfcWindow);
END_ENTITY;

ENTITY IfcWindowStyle
    SUBTYPE OF (IfcTypeProduct);
    ConstructionType : IfcWindowStyleConstructionEnum;
    OperationType : IfcWindowStyleOperationEnum;
    ParameterTakesPrecedence : IfcBoolean;
    Sizeable : IfcBoolean;
END_ENTITY;

ENTITY IfcWindowType
    SUBTYPE OF (IfcBuildingElementType);
    PredefinedType : IfcWindowTypeEnum;
    PartitioningType : IfcWindowTypePartitioningEnum;
    ParameterTakesPrecedence : OPTIONAL IfcBoolean;
    UserDefinedPartitioningType : OPTIONAL IfcLabel;
WHERE
    CorrectPredefinedType : (PredefinedType <> IfcWindowTypeEnum.USERDEFINED) OR
    ((PredefinedType = IfcWindowTypeEnum.USERDEFINED) AND
    EXISTS (SELF\IfcElementType.ElementType));
END_ENTITY;

ENTITY IfcWorkCalendar
    SUBTYPE OF (IfcControl);
    WorkingTimes : OPTIONAL SET [1:?] OF IfcWorkTime;
    ExceptionTimes : OPTIONAL SET [1:?] OF IfcWorkTime;
    PredefinedType : OPTIONAL IfcWorkCalendarTypeEnum;
WHERE

```

---

```

        CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR (PredefinedType <>
IfcWorkCalendarTypeEnum.USERDEFINED) OR
((PredefinedType = IfcWorkCalendarTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcObject.ObjectType));
END_ENTITY;

```

```

ENTITY IfcWorkControl
ABSTRACT SUPERTYPE OF (ONEOF
    (IfcWorkPlan
    , IfcWorkSchedule))
SUBTYPE OF (IfcControl);
    CreationDate : IfcDateTime;
    Creators : OPTIONAL SET [1:?] OF IfcPerson;
    Purpose : OPTIONAL IfcLabel;
    Duration : OPTIONAL IfcDuration;
    TotalFloat : OPTIONAL IfcDuration;
    StartTime : IfcDateTime;
    FinishTime : OPTIONAL IfcDateTime;
END_ENTITY;

```

```

ENTITY IfcWorkPlan
SUBTYPE OF (IfcWorkControl);
    PredefinedType : OPTIONAL IfcWorkPlanTypeEnum;
WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR (PredefinedType <>
IfcWorkPlanTypeEnum.USERDEFINED) OR
((PredefinedType = IfcWorkPlanTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcObject.ObjectType));
END_ENTITY;

```

```

ENTITY IfcWorkSchedule
SUBTYPE OF (IfcWorkControl);
    PredefinedType : OPTIONAL IfcWorkScheduleTypeEnum;
WHERE
    CorrectPredefinedType : NOT(EXISTS(PredefinedType)) OR (PredefinedType <>
IfcWorkScheduleTypeEnum.USERDEFINED) OR
((PredefinedType = IfcWorkScheduleTypeEnum.USERDEFINED) AND
EXISTS(SELF\IfcObject.ObjectType));
END_ENTITY;

```

```

ENTITY IfcWorkTime
SUBTYPE OF (IfcSchedulingTime);
    RecurrencePattern : OPTIONAL IfcRecurrencePattern;
    Start : OPTIONAL IfcDate;

```

---

```

    Finish : OPTIONAL IfcDate;
END_ENTITY;

ENTITY IfcZShapeProfileDef
  SUBTYPE OF (IfcParameterizedProfileDef);
    Depth : IfcPositiveLengthMeasure;
    FlangeWidth : IfcPositiveLengthMeasure;
    WebThickness : IfcPositiveLengthMeasure;
    FlangeThickness : IfcPositiveLengthMeasure;
    FilletRadius : OPTIONAL IfcNonNegativeLengthMeasure;
    EdgeRadius : OPTIONAL IfcNonNegativeLengthMeasure;
  WHERE
    ValidFlangeThickness : FlangeThickness < (Depth / 2.);
END_ENTITY;

ENTITY IfcZone
  SUBTYPE OF (IfcSystem);
    LongName : OPTIONAL IfcLabel;
  WHERE
    WR1 : (SIZEOF(SELF\IfcGroup.IsGroupedBy) = 0) OR
    (SIZEOF (QUERY (temp <* SELF\IfcGroup.IsGroupedBy[1].RelatedObjects |
      NOT(('IFC4.IFCZONE' IN TYPEOF(temp)) OR
        ('IFC4.IFCSPACE' IN TYPEOF(temp)) OR
        ('IFC4.IFCSPATIALZONE' IN TYPEOF(temp))
      ))) = 0);
END_ENTITY;

(*
*****
Gangzhou ifcEx
*****
*)

TYPE IfcSptType_GD = ENUMERATION OF
  (SPT_XTKWDZZ
   , SPT_JLQ
   , SPT_Beam
   , SPT_Column);
END_TYPE;

ENTITY IfcSteelTieinSumSet_GD;
  Level : IfcString;
  Diameter : IfcString;
  TieinType : IfcInteger;

```

---

```

        TieinNum : IfcInteger;
END_ENTITY;

ENTITY IfcBQSumItem_GD;
    Code : IfcString;
    Name : IfcString;
    Attr : IfcString;
    Content : IfcString;
    Type : IfcInteger;
    Unit : IfcString;
    Quantity : IfcReal;
    Memo : IfcString;
END_ENTITY;

ENTITY IfcBQSumSet_GD;
    Items : LIST [0:?] OF IfcBQSumItem_GD;
END_ENTITY;

ENTITY IfcBQElementQtyItem_GD;
    Code : IfcString;
    Name : IfcString;
    Attr : IfcString;
    Content : IfcString;
    Type : IfcInteger;
    Unit : IfcString;
    Quantity : IfcReal;
    Memo : IfcString;
END_ENTITY;

ENTITY IfcBQElementQtySet_GD;
    Items : LIST [0:?] OF IfcBQElementQtyItem_GD;
END_ENTITY;

ENTITY IfcBeamHangBar_GD;
    SpanID : IfcInteger;
    Param : IfcReal;
    BarInfo : IfcString;
    AddBar : IfcString;
END_ENTITY;

ENTITY IfcBeamSpan_GD;
    ID : IfcInteger;
    ESptParam : IfcReal;
    SSptType : OPTIONAL IfcSptType_GD;

```

---

```

    SSptLeft : OPTIONAL IfcReal;
    SSptRight : OPTIONAL IfcReal;
    ESptType : IfcSptType_GD;
    ESptLeft : OPTIONAL IfcReal;
    ESptRight : OPTIONAL IfcReal;
    SectionWidth : OPTIONAL IfcReal;
    SectionHeight : OPTIONAL IfcReal;
    StartPtTopElev : OPTIONAL IfcReal;
    EndPtTopElev : OPTIONAL IfcReal;
    LeftPileBar : IfcString;
    SpanBar : IfcString;
    RightPileBar : IfcString;
    BottomBar : IfcString;
    SideBar : IfcString;
    HoopBar : IfcString;
END_ENTITY;

ENTITY IfcColumnSectionHoopBar_GD;
    ID : IfcInteger;
    Radius : OPTIONAL IfcReal;
    BarInfo : IfcString;
    VertBarPts : LIST [0:?] OF IfcColumnSectionVertBar_GD;
END_ENTITY;

ENTITY IfcColumnSectionVertBar_GD;
    ID : IfcInteger;
    Pt : IfcCartesianPoint;
    BarInfo : IfcString;
END_ENTITY;

ENTITY IfcElementQtyItem_GD;
    Unit : IfcString;
    Expression : IfcString;
    Quantity : IfcReal;
    Memo : IfcString;
END_ENTITY;

ENTITY IfcElementQtySet_GD;
    Type : IfcInteger;
    Items : LIST [0:?] OF IfcElementQtyItem_GD;
END_ENTITY;

ENTITY IfcElementSteelItem_GD;
    Name : IfcString;

```

---

```

    Level : IfcString;
    Diameter : IfcInteger;
    Length : IfcInteger;
    Num : IfcInteger;
    SingleWeight : IfcReal;
    TotalWeight : IfcReal;
    TieinNum : IfcInteger;
    TieinType : IfcString;
END_ENTITY;

ENTITY IfcElementSteelSet_GD;
    Items : LIST [0:?] OF IfcElementSteelItem_GD;
END_ENTITY;

ENTITY IfcElement_GD
    SUBTYPE OF (IfcProduct);
END_ENTITY;

ENTITY IfcRelVoidsElement_GD
    SUBTYPE OF (IfcRelDecomposes);
    RelatingBuildingElement : IfcElement_GD;
    RelatedOpeningElement : IfcElement_GD;
END_ENTITY;

ENTITY IfcRelConnectsElements_GD
    SUBTYPE OF (IfcRelConnects);
    RelatingElement : IfcElement_GD;
    RelatedElement : IfcElement_GD;
END_ENTITY;

ENTITY IfcSteelSumItem_GD;
    ElementType : IfcString;
    Level : IfcString;
    Diameter : IfcInteger;
    Weight : IfcReal;
    Unit : IfcString;
END_ENTITY;

ENTITY IfcSteelSumSet_GD;
    Items : LIST [0:?] OF IfcSteelSumItem_GD;
END_ENTITY;

ENTITY IfcInsulatingLayer_GD
    SUBTYPE OF (IfcElement_GD);

```

---

```
    Material : IfcInteger;
    Thickness : IfcInteger;
    AirLayerThickness : IfcInteger;
    StartPtBottomElev : IfcReal;
    EndPtBottomElev : IfcReal;
    StartPtTopElev : IfcReal;
    EndPtTopElev : IfcReal;
END_ENTITY;
```

```
ENTITY IfcForceWall_GD
  SUBTYPE OF (IfcElement_GD);
    HoriDistributionBar : IfcString;
    VertDistributionBar : IfcString;
    TieBar : IfcString;
    ConcGradeID : IfcInteger;
    InnerOuterFlag : IfcInteger;
    AseismaticLevel : IfcInteger;
    AnchorJoint : IfcString;
    ProtectiveThickness : IfcInteger;
END_ENTITY;
```

```
ENTITY IfcAloneBeamFinish_GD
  SUBTYPE OF (IfcElement_GD);
    MaterialThickness : IfcInteger;
END_ENTITY;
```

```
ENTITY IfcCoping_GD
  SUBTYPE OF (IfcElement_GD);
    Material : IfcInteger;
    MortarTypeID : IfcInteger;
    MortarGradeID : IfcInteger;
    ConcGradeID : IfcInteger;
END_ENTITY;
```

```
ENTITY IfcFootStep_GD
  SUBTYPE OF (IfcElement_GD);
    Height : IfcInteger;
    StepHeight : IfcInteger;
    Material : IfcInteger;
    ConcGradeID : IfcInteger;
    MortarTypeID : IfcInteger;
    MortarGradeID : IfcInteger;
    TopElev : IfcReal;
END_ENTITY;
```

---

```
ENTITY IfcHungCeiling_GD
  SUBTYPE OF (IfcElement_GD);
    AboveFloorHeight : IfcInteger;
END_ENTITY;

ENTITY IfcRingBeam_GD
  SUBTYPE OF (IfcElement_GD);
    TopHoriBar : IfcString;
    BottomHoriBar : IfcString;
    HoopBar : IfcString;
    Limb : IfcInteger;
    ConcGradeID : IfcInteger;
    ProtectiveThickness : IfcInteger;
    AseismaticLevel : IfcInteger;
    AnchorJoint : IfcString;
END_ENTITY;

ENTITY IfcBedding_GD
  SUBTYPE OF (IfcElement_GD);
    Thickness : IfcInteger;
    ConcGradeID : IfcInteger;
    TopElev : IfcReal;
END_ENTITY;

ENTITY IfcPitEarth_GD
  SUBTYPE OF (IfcElement_GD);
    EarthTypeID : IfcInteger;
    WetSoilThickness : IfcInteger;
    ExcavatedTypeID : IfcString;
    TransportDistance : IfcString;
END_ENTITY;

ENTITY IfcDitchEarth_GD
  SUBTYPE OF (IfcElement_GD);
    EarthTypeID : IfcInteger;
    WetSoilThickness : IfcInteger;
    ExcavatedTypeID : IfcString;
    TransportDistance : IfcString;
END_ENTITY;

ENTITY IfcFDBeam_GD
  SUBTYPE OF (IfcElement_GD);
    Type : IfcInteger;
```



---

```
SpanCount : IfcInteger;
HoopBar : IfcString;
Limb : IfcInteger;
TopThroughBar : IfcString;
BottomThroughBar : IfcString;
SideBar : IfcString;
TieBar : IfcString;
Spans : LIST [0:?] OF IfcBeamSpan_GD;
HangBars : LIST [0:?] OF IfcBeamHangBar_GD;
ConcGradeID : IfcInteger;
ProtectiveThickness : IfcInteger;
AseismaticLevel : IfcInteger;
AnchorJoint : IfcString;
END_ENTITY;
```

```
ENTITY IfcWallHole_GD
  SUBTYPE OF (IfcElement_GD);
    ReinforcedBar : IfcString;
    ObliqueReinforcedBar : IfcString;
    EmbedBeamHeigth : IfcInteger;
    EmbedBeamVertBar : IfcString;
    EmbedBeamHoopReinf : IfcString;
END_ENTITY;
```

```
ENTITY IfcDado_GD
  SUBTYPE OF (IfcElement_GD);
    Height : IfcInteger;
    Thickness : IfcInteger;
    StartPtBottomElev : IfcReal;
    EndPtBottomElev : IfcReal;
END_ENTITY;
```

```
ENTITY IfcWallFaceFinish_GD
  SUBTYPE OF (IfcElement_GD);
    Thickness : IfcInteger;
    Material : IfcInteger;
    InnerOuterFlag : IfcInteger;
    StartPtBottomElev : IfcReal;
    EndPtBottomElev : IfcReal;
    StartPtTopElev : IfcReal;
    EndPtTopElev : IfcReal;
END_ENTITY;
```

```
ENTITY IfcAlcove_GD
```

---

```
SUBTYPE OF (IfcElement_GD);
    FrameThickness : IfcInteger;
END_ENTITY;

ENTITY IfcExcavatedEarth_GD
SUBTYPE OF (IfcElement_GD);
    EarthTypeID : IfcInteger;
    WetSoilThickness : IfcInteger;
    ExcavatedTypeID : IfcString;
    TransportDistance : IfcString;
END_ENTITY;

ENTITY IfcRoof_GD
SUBTYPE OF (IfcElement_GD);
    TopElev : IfcReal;
END_ENTITY;

ENTITY IfcRibbonWindow_GD
SUBTYPE OF (IfcElement_GD);
    FrameOffset : IfcInteger;
    FillingHeight : IfcInteger;
    FillingWidth : IfcInteger;
END_ENTITY;

ENTITY IfcCurtainWall_GD
SUBTYPE OF (IfcElement_GD);
    Material : IfcInteger;
    StructType : IfcInteger;
    InnerOuterFlag : IfcInteger;
END_ENTITY;

ENTITY IfcSiteLeveling_GD
SUBTYPE OF (IfcElement_GD);
    SiteLevelingType : IfcInteger;
END_ENTITY;

ENTITY IfcFloorArea_GD
SUBTYPE OF (IfcElement_GD);
    FloorAreaCalcStyle : IfcInteger;
    BottomElev : IfcReal;
END_ENTITY;

ENTITY IfcApron_GD
SUBTYPE OF (IfcElement_GD);
```

---

```

        Thickness : IfcInteger;
        ConcGradeID : IfcInteger;
END_ENTITY;

ENTITY IfcEmbedBeam_GD
    SUBTYPE OF (IfcElement_GD);
        HoriBar : IfcString;
        TopHoriBar : IfcString;
        BottomHoriBar : IfcString;
        HoopBar : IfcString;
        Limb : IfcInteger;
        TieBar : IfcString;
        ConcGradeID : IfcInteger;
        ProtectiveThickness : IfcInteger;
        AseismaticLevel : IfcInteger;
        AnchorJoint : IfcString;
END_ENTITY;

ENTITY IfcReinforcement_GD
    SUBTYPE OF (IfcElement_GD);
        Position : IfcInteger;
        BarInfo : IfcString;
        LeftLabel : IfcInteger;
        RightLabel : IfcInteger;
        LabelPosition : IfcInteger;
        LeftHook : IfcInteger;
        RightHook : IfcInteger;
        DistributedBar : IfcString;
        Anchor : IfcString;
        Joint : IfcString;
END_ENTITY;

ENTITY IfcSlabHole_GD
    SUBTYPE OF (IfcElement_GD);
        ShortSpanReinforcedBar : IfcString;
        LongSpanReinforcedBar : IfcString;
        CircleReinforcedBar : IfcString;
END_ENTITY;

ENTITY IfcNegReinforcement_GD
    SUBTYPE OF (IfcElement_GD);
        BarInfo : IfcString;
        LeftLabel : IfcInteger;
        RightLabel : IfcInteger;

```

---

```
    LabelPosition : IfcInteger;
    IncludeSupport : IfcInteger;
    LeftHook : IfcInteger;
    RightHook : IfcInteger;
    DistributedBar : IfcString;
    Anchor : IfcString;
    Joint : IfcString;
END_ENTITY;
```

```
ENTITY IfcTieColumn_GD
  SUBTYPE OF (IfcElement_GD);
    Type : IfcInteger;
    HorseTeethCroucher : IfcBoolean;
    IndentingWidth : IfcInteger;
    IndentingHeight : IfcInteger;
    ConcGradeID : IfcInteger;
    AseismaticLevel : IfcInteger;
    AnchorJoint : IfcString;
    ProtectiveThickness : IfcInteger;
    VertBar : IfcString;
    CornerBar : IfcString;
    BSideBar : IfcString;
    HSideBar : IfcString;
    HoopBar : IfcString;
END_ENTITY;
```

```
ENTITY IfcColumn_GD
  SUBTYPE OF (IfcElement_GD);
    Type : IfcInteger;
    NodeHoopBar : IfcString;
    HoopBarType : IfcInteger;
    Position : IfcInteger;
    ConcGradeID : IfcInteger;
    AseismaticLevel : IfcInteger;
    AnchorJoint : IfcString;
    ProtectiveThickness : IfcInteger;
    SectionVertBar : LIST [0:?] OF IfcColumnSectionVertBar_GD;
    SectionHoopBar : LIST [0:?] OF IfcColumnSectionHoopBar_GD;
END_ENTITY;
```

```
ENTITY IfcColumnBase_GD
  SUBTYPE OF (IfcElement_GD);
    ConcGradeID : IfcInteger;
END_ENTITY;
```

---

```
ENTITY IfcColumnCap_GD
  SUBTYPE OF (IfcElement_GD);
    ConcGradeID : IfcInteger;
END_ENTITY;
```

```
ENTITY IfcParapet_GD
  SUBTYPE OF (IfcElement_GD);
    Material : IfcInteger;
    HoriDistributionBar : IfcString;
    VertDistributionBar : IfcString;
    TieBar : IfcString;
    ConcGradeID : IfcInteger;
    ProtectiveThickness : IfcInteger;
    AseismaticLevel : IfcInteger;
    AnchorJoint : IfcString;
    MortarTypeID : IfcInteger;
    MortarGradeID : IfcInteger;
END_ENTITY;
```

```
ENTITY IfcBeam_GD
  SUBTYPE OF (IfcElement_GD);
    Type : IfcInteger;
    SpanCount : IfcInteger;
    HoopBar : IfcString;
    Limb : IfcInteger;
    TopThroughBar : IfcString;
    BottomThroughBar : IfcString;
    SideBar : IfcString;
    TieBar : IfcString;
    Spans : LIST [0:?] OF IfcBeamSpan_GD;
    HangBars : LIST [0:?] OF IfcBeamHangBar_GD;
    ConcGradeID : IfcInteger;
    ProtectiveThickness : IfcInteger;
    AseismaticLevel : IfcInteger;
    AnchorJoint : IfcString;
END_ENTITY;
```

```
ENTITY IfcFloorFinish_GD
  SUBTYPE OF (IfcElement_GD);
    MaterialThickness : IfcInteger;
    Waterproof : IfcInteger;
    RolledEdgeFactor : IfcString;
    TopElev : IfcReal;
```

---

END\_ENTITY;

ENTITY IfcSlabStrip\_GD

SUBTYPE OF (IfcElement\_GD);  
    ConcGradeID : IfcInteger;  
    Thickness : IfcInteger;  
    Width : IfcInteger;  
    AxisOffset : IfcInteger;  
    Type : IfcInteger;  
    BottomReinforcement : IfcString;  
    TopReinforcement : IfcString;  
    HoopBar : IfcString;  
    Limb : IfcInteger;  
    TopElev : IfcReal;  
    SideBeamHoopBarWidth : IfcInteger;  
    SideBeamAxisOffset : IfcInteger;  
    SideBeamBottomReinforcement : IfcString;  
    SideBeamTopReinforcement : IfcString;  
    SideBeamHoopBar : IfcString;  
    SideBeamLimb : IfcInteger;  
    IsThroughHoopBar : IfcInteger;  
    ProtectiveThickness : IfcInteger;  
    AseismaticLevel : IfcInteger;  
    AnchorJoint : IfcString;

END\_ENTITY;

ENTITY IfcStair\_GD

SUBTYPE OF (IfcElement\_GD);  
    FloorAreaCalcStyle : IfcInteger;  
    EDOShape : IfcInteger;  
    ConcGradeID : IfcInteger;

END\_ENTITY;

ENTITY IfcIsolatedColumnFinish\_GD

SUBTYPE OF (IfcElement\_GD);  
    MaterialThickness : IfcInteger;  
    BottomElev : IfcReal;  
    TopElev : IfcReal;

END\_ENTITY;

ENTITY IfcSlab\_GD

SUBTYPE OF (IfcElement\_GD);  
    Thickness : IfcInteger;  
    ConcGradeID : IfcInteger;

---

```

        TopElev : IfcReal;
END_ENTITY;

ENTITY IfcFlight_GD
    SUBTYPE OF (IfcElement_GD);
        SlabThickness : IfcInteger;
        StepHeight : IfcInteger;
        FlightHeight : IfcInteger;
        FloorAreaCalcStyle : IfcInteger;
        ConcGradeID : IfcInteger;
        BottomElev : IfcReal;
END_ENTITY;

ENTITY IfcBrickWall_GD
    SUBTYPE OF (IfcElement_GD);
        InnerOuterFlag : IfcInteger;
        Material : IfcInteger;
        MortarTypeID : IfcInteger;
        MortarGradeID : IfcInteger;
        MasonryThroughBar : IfcString;
        TransverseBar : IfcString;
END_ENTITY;

ENTITY IfcMasonryColumn_GD
    SUBTYPE OF (IfcElement_GD);
        Material : IfcInteger;
        MortarTypeID : IfcInteger;
        MortarGradeID : IfcInteger;
END_ENTITY;

ENTITY IfcWindow_GD
    SUBTYPE OF (IfcElement_GD);
        FrameOffset : IfcInteger;
        FillingHeight : IfcInteger;
        FillingWidth : IfcInteger;
END_ENTITY;

ENTITY IfcRaftFDReinf_GD
    SUBTYPE OF (IfcElement_GD);
        Position : IfcInteger;
        BarInfo : IfcString;
        LeftLabel : IfcInteger;
        RightLabel : IfcInteger;
        LabelPosition : IfcInteger;

```

---

```
    LeftHook : IfcInteger;  
    RightHook : IfcInteger;  
    DistributedBar : IfcString;  
    Anchor : IfcString;  
    Joint : IfcString;  
END_ENTITY;
```

```
ENTITY IfcRaftFD_GD  
  SUBTYPE OF (IfcElement_GD);  
    Thickness : IfcInteger;  
    ConcGradeID : IfcInteger;  
    Type : IfcInteger;  
END_ENTITY;
```

```
ENTITY IfcRaftFDNegReinf_GD  
  SUBTYPE OF (IfcElement_GD);  
    BarInfo : IfcString;  
    LeftLabel : IfcInteger;  
    RightLabel : IfcInteger;  
    LabelPosition : IfcInteger;  
    IncludeSupport : IfcInteger;  
    LeftHook : IfcInteger;  
    RightHook : IfcInteger;  
    DistributedBar : IfcString;  
    Anchor : IfcString;  
    Joint : IfcString;  
END_ENTITY;
```

```
ENTITY IfcDormer_GD  
  SUBTYPE OF (IfcElement_GD);  
    ConcGradeID : IfcInteger;  
END_ENTITY;
```

```
ENTITY IfcSpiralSlab_GD  
  SUBTYPE OF (IfcElement_GD);  
    SectionWidth : IfcInteger;  
    Thickness : IfcInteger;  
    InsideRadius : IfcInteger;  
    RotationDirection : IfcInteger;  
    RotationAngle : IfcInteger;  
    HorzBarPosition : IfcInteger;  
    BottomHorzBar : IfcString;  
    BottomVertBar : IfcString;  
    TopHorzBar : IfcString;
```



---

```
    TopVertBar : IfcString;
    ConcGradeID : IfcInteger;
    BottomElev : IfcReal;
    TopElev : IfcReal;
    ProtectiveThickness : IfcInteger;
    AseismaticLevel : IfcInteger;
    AnchorJoint : IfcString;
END_ENTITY;
```

```
ENTITY IfcSpiralFlight_GD
  SUBTYPE OF (IfcElement_GD);
    StairSlabThickness : IfcInteger;
    FlightWidth : IfcInteger;
    StepHeight : IfcInteger;
    TotalStepHeight : IfcInteger;
    InnerRadius : IfcInteger;
    Clockwise : IfcInteger;
    RotationAngle : IfcInteger;
    FloorAreaCalcStyle : IfcInteger;
    ConcGradeID : IfcInteger;
    BottomElev : IfcReal;
END_ENTITY;
```

```
ENTITY IfcSkirt_GD
  SUBTYPE OF (IfcElement_GD);
    Height : IfcInteger;
    Thickness : IfcInteger;
    StartPtBottomElev : IfcReal;
    EndPtBottomElev : IfcReal;
END_ENTITY;
```

```
ENTITY IfcLintel_GD
  SUBTYPE OF (IfcElement_GD);
    HoriBar : IfcString;
    TopHoriBar : IfcString;
    BottomHoriBar : IfcString;
    HoopBar : IfcString;
    Limb : IfcInteger;
    ConcGradeID : IfcInteger;
    StartPtLenInWall : IfcInteger;
    EndPtLenInWall : IfcInteger;
    ProtectiveThickness : IfcInteger;
    AseismaticLevel : IfcInteger;
    AnchorJoint : IfcString;
```

---

END\_ENTITY;

ENTITY IfcLinkBeam\_GD

SUBTYPE OF (IfcElement\_GD);  
SpanCount : IfcInteger;  
HoopBar : IfcString;  
Limb : IfcInteger;  
TopThroughBar : IfcString;  
BottomThroughBar : IfcString;  
SideBar : IfcString;  
TieBar : IfcString;  
ConcGradeID : IfcInteger;  
ProtectiveThickness : IfcInteger;  
AseismaticLevel : IfcInteger;  
AnchorJoint : IfcString;  
TopLinkBeam : IfcInteger;  
FoldBar : IfcString;  
EmbedBracingHeight : IfcInteger;  
EmbedBracingWidth : IfcInteger;  
EmbedBracingHeight11G : IfcInteger;  
EmbedBracingBar : IfcString;  
EmbedBracingHoopBar : IfcString;

END\_ENTITY;

ENTITY IfcDoor\_GD

SUBTYPE OF (IfcElement\_GD);  
FrameOffset : IfcInteger;  
FillingHeight : IfcInteger;  
FillingWidth : IfcInteger;

END\_ENTITY;

ENTITY IfcDoorWin\_GD

SUBTYPE OF (IfcElement\_GD);  
OpeningWidth : IfcInteger;  
OpeningHeight : IfcInteger;  
WinDoorWidth : IfcInteger;  
AboveFloorHeight : IfcInteger;  
WinDoorFloorHeight : IfcInteger;  
DoorWinLocation : IfcInteger;  
FrameThickness : IfcInteger;  
FrameOffset : IfcInteger;  
DoorFillingHeight : IfcInteger;  
DoorFillingWidth : IfcInteger;  
WinFillingHeight : IfcInteger;

---

```
        WinFillingWidth : IfcInteger;
END_ENTITY;

ENTITY IfcSump_GD
  SUBTYPE OF (IfcElement_GD);
    Outside : IfcInteger;
    BottomSlabThick : IfcInteger;
    SumpElev : IfcReal;
    SlopeInput : IfcInteger;
    BottomWide : IfcInteger;
    Angle : IfcInteger;
    TransverseBottomBar : IfcString;
    TransverseBar : IfcString;
    VertBottomBar : IfcString;
    VertBar : IfcString;
    HoleWallBar : IfcString;
    SlopeBar : IfcString;
    SlopeYBar : IfcString;
    ConcGradeID : IfcInteger;
    GetRaftBar : IfcInteger;
    GetRFDBar : IfcInteger;
    ProtectiveThickness : IfcInteger;
    AseismaticLevel : IfcInteger;
    AnchorJoint : IfcString;
END_ENTITY;

ENTITY IfcCanopy_GD
  SUBTYPE OF (IfcElement_GD);
    Thickness : IfcInteger;
    ConcGradeID : IfcInteger;
    TopElev : IfcReal;
END_ENTITY;

ENTITY IfcSwingWin_GD
  SUBTYPE OF (IfcElement_GD);
    ConcGradeID : IfcInteger;
    FloorAreaCalcStyle : IfcInteger;
END_ENTITY;

ENTITY IfcInsulatingWall_GD
  SUBTYPE OF (IfcElement_GD);
END_ENTITY;

ENTITY IfcPostCastStrip_GD
```

---

```
SUBTYPE OF (IfcElement_GD);
END_ENTITY;

ENTITY IfcPitEarthBackfill_GD
  SUBTYPE OF (IfcElement_GD);
END_ENTITY;

ENTITY IfcDitchEarthBackfill_GD
  SUBTYPE OF (IfcElement_GD);
END_ENTITY;

ENTITY IfcFDSlabStrip_GD
  SUBTYPE OF (IfcElement_GD);
END_ENTITY;

ENTITY IfcPier_GD
  SUBTYPE OF (IfcElement_GD);
END_ENTITY;

ENTITY IfcExcavatedEarthBackfill_GD
  SUBTYPE OF (IfcElement_GD);
END_ENTITY;

ENTITY IfcCeiling_GD
  SUBTYPE OF (IfcElement_GD);
END_ENTITY;

ENTITY IfcRibbonOpening_GD
  SUBTYPE OF (IfcElement_GD);
END_ENTITY;

ENTITY IfcRoomBackFill_GD
  SUBTYPE OF (IfcElement_GD);
END_ENTITY;

ENTITY IfcEave_GD
  SUBTYPE OF (IfcElement_GD);
END_ENTITY;

ENTITY IfcStripFD_GD
  SUBTYPE OF (IfcElement_GD);
END_ENTITY;

ENTITY IfcPile_GD
```

---

```

    SUBTYPE OF (IfcElement_GD);
END_ENTITY;

ENTITY IfcPileCap_GD
    SUBTYPE OF (IfcElement_GD);
END_ENTITY;

ENTITY IfcIsolatedFD_GD
    SUBTYPE OF (IfcElement_GD);
END_ENTITY;

ENTITY IfcMasonryReinf_GD
    SUBTYPE OF (IfcElement_GD);
END_ENTITY;

FUNCTION IfcAssociatedSurface
(Arg : IfcPcurve) : IfcSurface;

    LOCAL
        Surf : IfcSurface;
    END_LOCAL;

    Surf := Arg\IfcPcurve.BasisSurface;

    RETURN(Surf);
END_FUNCTION;

FUNCTION IfcBaseAxis
(Dim : INTEGER;
 Axis1, Axis2, Axis3 : IfcDirection)
: LIST [2:3] OF IfcDirection;

LOCAL
    U : LIST [2:3] OF IfcDirection;
    Factor : REAL;
    D1, D2 : IfcDirection;
END_LOCAL;

    IF (Dim = 3) THEN
        D1      :=      NVL(IfcNormalise(Axis3),      IfcRepresentationItem()      ||
IfcGeometricRepresentationItem () || IfcDirection([0.0,0.0,1.0]));
        D2 := IfcFirstProjAxis(D1, Axis1);
        U  := [D2, IfcSecondProjAxis(D1, D2, Axis2), D1];
    ELSE

```

---

```

    IF EXISTS(Axis1) THEN
        D1 := IfcNormalise(Axis1);
        U := [D1, IfcOrthogonalComplement(D1)];
        IF EXISTS(Axis2) THEN
            Factor := IfcDotProduct(Axis2, U[2]);
            IF (Factor < 0.0) THEN
                U[2].DirectionRatios[1] := -U[2].DirectionRatios[1];
                U[2].DirectionRatios[2] := -U[2].DirectionRatios[2];
            END_IF;
        END_IF;
    ELSE
        IF EXISTS(Axis2) THEN
            D1 := IfcNormalise(Axis2);
            U := [IfcOrthogonalComplement(D1), D1];
            U[1].DirectionRatios[1] := -U[1].DirectionRatios[1];
            U[1].DirectionRatios[2] := -U[1].DirectionRatios[2];
        ELSE
            U := [IfcRepresentationItem() || IfcGeometricRepresentationItem () ||
IfcDirection([1.0, 0.0]),
                IfcRepresentationItem() || IfcGeometricRepresentationItem () ||
IfcDirection([0.0, 1.0])];
        END_IF;
    END_IF;
    RETURN(U);
END_FUNCTION;

FUNCTION IfcBooleanChoose
(B : BOOLEAN ;
    Choice1, Choice2 : GENERIC : Item) : GENERIC : Item;
    IF B THEN
        RETURN (Choice1);
    ELSE
        RETURN (Choice2);
    END_IF;
END_FUNCTION;

FUNCTION IfcBuild2Axes
(RefDirection : IfcDirection)
: LIST [2:2] OF IfcDirection;
LOCAL
    D : IfcDirection := NVL(IfcNormalise(RefDirection),
        IfcRepresentationItem() || IfcGeometricRepresentationItem () ||
IfcDirection([1.0, 0.0]));

```

---

```

END_LOCAL;
  RETURN([D, IfcOrthogonalComplement(D)]);
END_FUNCTION;

FUNCTION IfcBuildAxes
(Axis, RefDirection : IfcDirection)
  : LIST [3:3] OF IfcDirection;
LOCAL
  D1, D2 : IfcDirection;
END_LOCAL;
  D1      :=      NVL(IfcNormalise(Axis),      IfcRepresentationItem()      ||
IfcGeometricRepresentationItem () || IfcDirection([0.0,0.0,1.0]));
  D2 := IfcFirstProjAxis(D1, RefDirection);
  RETURN ([D2, IfcNormalise(IfcCrossProduct(D1,D2))\IfcVector.Orientation, D1]);
END_FUNCTION;

FUNCTION IfcConsecutiveSegments
(Segments : LIST [1:?] OF IfcSegmentIndexSelect)
  : BOOLEAN;

LOCAL
  Result : BOOLEAN := TRUE;
END_LOCAL;

  REPEAT i := 1 TO (HIINDEX(Segments)-1);
    IF Segments[i][HIINDEX(Segments[i])] <> Segments[i+1][1] THEN
      BEGIN
        Result := FALSE;
        ESCAPE;
      END;
    END_IF;
  END_REPEAT;

  RETURN (Result);
END_FUNCTION;

FUNCTION IfcConstraintsParamBSpline
( Degree, UpKnots, UpCp : INTEGER;
  KnotMult : LIST OF INTEGER;
  Knots : LIST OF IfcParameterValue )
: BOOLEAN;

LOCAL

```

---

```

    Result : BOOLEAN := TRUE;
    K, Sum : INTEGER;
END_LOCAL;

(* Find sum of knot multiplicities. *)
Sum := KnotMult[1];
REPEAT i := 2 TO UpKnots;
    Sum := Sum + KnotMult[i];
END_REPEAT;

(* Check limits holding for all B-spline parametrisations *)
IF (Degree < 1) OR (UpKnots < 2) OR (UpCp < Degree) OR
    (Sum <> (Degree + UpCp + 2)) THEN
    Result := FALSE;
    RETURN(Result);
END_IF;

K := KnotMult[1];
IF (K < 1) OR (K > Degree + 1) THEN
    Result := FALSE;
    RETURN(Result);
END_IF;

REPEAT i := 2 TO UpKnots;
    IF (KnotMult[i] < 1) OR (Knots[i] <= Knots[i-1]) THEN
        Result := FALSE;
        RETURN(Result);
    END_IF;
    K := KnotMult[i];
    IF (i < UpKnots) AND (K > Degree) THEN
        Result := FALSE;
        RETURN(Result);
    END_IF;
    IF (i = UpKnots) AND (K > Degree + 1) THEN
        Result := FALSE;
        RETURN(Result);
    END_IF;
END_REPEAT;

RETURN(result);
END_FUNCTION;

FUNCTION IfcConvertDirectionInto2D
(Direction : IfcDirection)

```



---

```

        : IfcDirection;

LOCAL
    Direction2D      :      IfcDirection      :=      IfcRepresentationItem()      ||
IfcGeometricRepresentationItem () || IfcDirection([0.,1.]);
    END_LOCAL;

    Direction2D.DirectionRatios[1] := Direction.DirectionRatios[1];
    Direction2D.DirectionRatios[2] := Direction.DirectionRatios[2];

    RETURN (Direction2D);
END_FUNCTION;

FUNCTION IfcCorrectDimensions
(m      : IfcUnitEnum; Dim : IfcDimensionalExponents) : LOGICAL;
CASE m OF
    LENGTHUNIT : IF
        Dim = (IfcDimensionalExponents (1, 0, 0, 0, 0, 0, 0))
            THEN RETURN(TRUE);
            ELSE RETURN(FALSE);
        END_IF;
    MASSUNIT : IF
        Dim = (IfcDimensionalExponents (0, 1, 0, 0, 0, 0, 0))
            THEN RETURN(TRUE);
            ELSE RETURN(FALSE);
        END_IF;
    TIMEUNIT : IF
        Dim = (IfcDimensionalExponents (0, 0, 1, 0, 0, 0, 0))
            THEN RETURN(TRUE);
            ELSE RETURN(FALSE);
        END_IF;
    ELECTRICCURRENTUNIT : IF
        Dim = (IfcDimensionalExponents (0, 0, 0, 1, 0, 0, 0))
            THEN RETURN(TRUE);
            ELSE RETURN(FALSE);
        END_IF;
    THERMODYNAMICTEMPERATUREUNIT : IF
        Dim = (IfcDimensionalExponents (0, 0, 0, 0, 1, 0, 0))
            THEN RETURN(TRUE);
            ELSE RETURN(FALSE);
        END_IF;
    AMOUNTOFSUBSTANCEUNIT : IF
        Dim = (IfcDimensionalExponents (0, 0, 0, 0, 0, 1, 0))
            THEN RETURN(TRUE);

```

---

```

        ELSE RETURN(FALSE);
    END_IF;
LUMINOUSINTENSITYUNIT : IF
    Dim = (IfcDimensionalExponents (0, 0, 0, 0, 0, 0, 1))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
    END_IF;
PLANEANGLEUNIT : IF
    Dim = (IfcDimensionalExponents (0, 0, 0, 0, 0, 0, 0))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
    END_IF;
SOLIDANGLEUNIT : IF
    Dim = (IfcDimensionalExponents (0, 0, 0, 0, 0, 0, 0))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
    END_IF;
AREAUNIT : IF
    Dim = (IfcDimensionalExponents (2, 0, 0, 0, 0, 0, 0))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
    END_IF;
VOLUMEUNIT : IF
    Dim = (IfcDimensionalExponents (3, 0, 0, 0, 0, 0, 0))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
    END_IF;

ABSORBEDDOSEUNIT : IF
    Dim = (IfcDimensionalExponents (2, 0, -2, 0, 0, 0, 0))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
    END_IF;
RADIOACTIVITYUNIT : IF
    Dim = (IfcDimensionalExponents (0, 0, -1, 0, 0, 0, 0))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
    END_IF;
ELECTRICCAPACITANCEUNIT : IF
    Dim = (IfcDimensionalExponents (-2, -1, 4, 2, 0, 0, 0))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
    END_IF;
DOSEEQUIVALENTUNIT : IF

```

---

```

    Dim = (IfcDimensionalExponents (2, 0, -2, 0, 0, 0, 0))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
END_IF;
ELECTRICCHARGEUNIT : IF
    Dim = (IfcDimensionalExponents (0, 0, 1, 1, 0, 0, 0))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
END_IF;
ELECTRICCONDUCTANCEUNIT : IF
    Dim = (IfcDimensionalExponents (-2, -1, 3, 2, 0, 0, 0))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
END_IF;
ELECTRICVOLTAGEUNIT : IF
    Dim = (IfcDimensionalExponents (2, 1, -3, -1, 0, 0, 0))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
END_IF;
ELECTRICRESISTANCEUNIT : IF
    Dim = (IfcDimensionalExponents (2, 1, -3, -2, 0, 0, 0))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
END_IF;
ENERGYUNIT : IF
    Dim = (IfcDimensionalExponents (2, 1, -2, 0, 0, 0, 0))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
END_IF;
FORCEUNIT : IF
    Dim = (IfcDimensionalExponents (1, 1, -2, 0, 0, 0, 0))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
END_IF;
FREQUENCYUNIT : IF
    Dim = (IfcDimensionalExponents (0, 0, -1, 0, 0, 0, 0))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
END_IF;
INDUCTANCEUNIT : IF
    Dim = (IfcDimensionalExponents (2, 1, -2, -2, 0, 0, 0))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
END_IF;

```

---

```

ILLUMINANCEUNIT : IF
    Dim = (IfcDimensionalExponents (-2, 0, 0, 0, 0, 0, 1))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
END_IF;
LUMINOUSFLUXUNIT : IF
    Dim = (IfcDimensionalExponents (0, 0, 0, 0, 0, 0, 1))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
END_IF;
MAGNETICFLUXUNIT : IF
    Dim = (IfcDimensionalExponents (2, 1, -2, -1, 0, 0, 0))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
END_IF;
MAGNETICFLUXDENSITYUNIT : IF
    Dim = (IfcDimensionalExponents (0, 1, -2, -1, 0, 0, 0))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
END_IF;
POWERUNIT : IF
    Dim = (IfcDimensionalExponents (2, 1, -3, 0, 0, 0, 0))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
END_IF;
PRESSUREUNIT : IF
    Dim = (IfcDimensionalExponents (-1, 1, -2, 0, 0, 0, 0))
    THEN RETURN(TRUE);
    ELSE RETURN(FALSE);
END_IF;

OTHERWISE :
    RETURN (UNKNOWN);
END_CASE;
END_FUNCTION;

FUNCTION IfcCorrectFillAreaStyle
(Styles : SET[1:?] OF IfcFillStyleSelect)
:LOGICAL;

LOCAL
    Hatching : INTEGER := 0;
    Tiles      : INTEGER := 0;
    Colour     : INTEGER := 0;

```

---

```

    External : INTEGER := 0;
END_LOCAL;

External := SIZEOF(QUERY(Style <* Styles |
    ' IFC4. IFCEXTERNALLYDEFINEDHATCHSTYLE' IN
    TYPEOF(Style)));

Hatching := SIZEOF(QUERY(Style <* Styles |
    ' IFC4. IFCFILLAREASTYLEHATCHING' IN
    TYPEOF(Style)));

Tiles      := SIZEOF(QUERY(Style <* Styles |
    ' IFC4. IFCFILLAREASTYLETILES' IN
    TYPEOF(Style)));

Colour     := SIZEOF(QUERY(Style <* Styles |
    ' IFC4. IFCCOLOUR' IN
    TYPEOF(Style)));

IF (External > 1) THEN
    RETURN (FALSE);
END_IF;

IF ((External = 1) AND ((Hatching > 0) OR (Tiles > 0) OR (Colour > 0))) THEN
    RETURN (FALSE);
END_IF;

IF (Colour > 1) THEN
    RETURN (FALSE);
END_IF;

IF ((Hatching > 0) AND (Tiles > 0)) THEN
    RETURN (FALSE);
END_IF;

RETURN(TRUE);
END_FUNCTION;

FUNCTION IfcCorrectLocalPlacement
(AxisPlacement:IfcAxis2Placement;

```

---

```

    RelPlacement : IfcObjectPlacement):LOGICAL;

IF (EXISTS(RelPlacement)) THEN
    IF ( ' IFC4. IFCGRIDPLACEMENT' IN TYPEOF(RelPlacement)) THEN
        RETURN(?);
    END_IF;
    IF ( ' IFC4. IFCLOCALPLACEMENT' IN TYPEOF(RelPlacement)) THEN
        IF ( ' IFC4. IFCAXIS2PLACEMENT2D' IN TYPEOF(AxisPlacement)) THEN
            RETURN(TRUE);
        END_IF;
        IF ( ' IFC4. IFCAXIS2PLACEMENT3D' IN TYPEOF(AxisPlacement)) THEN
            IF (RelPlacement\IfcLocalPlacement.RelativePlacement.Dim = 3) THEN
                RETURN(TRUE);
            ELSE
                RETURN(FALSE);
            END_IF;
        END_IF;
    END_IF;
ELSE
    RETURN(TRUE);
END_IF;
RETURN(?);
END_FUNCTION;

FUNCTION IfcCorrectObjectAssignment
(Constraint: IfcObjectTypeEnum; Objects : SET[1:?] OF IfcObjectDefinition)
: LOGICAL ;

LOCAL
    Count : INTEGER := 0;
END_LOCAL;

IF NOT(EXISTS(Constraint)) THEN
    RETURN(TRUE);
END_IF;

CASE Constraint OF
    IfcObjectTypeEnum.NOTDEFINED : RETURN(TRUE);
    IfcObjectTypeEnum.PRODUCT :
        BEGIN
            Count := SIZEOF(QUERY(temp <* Objects | NOT(' IFC4. IFCPRODUCT' IN
TYPEOF(temp))));
            RETURN(Count = 0);
        END;

```

---

```

    IfcObjectTypeEnum.PROCESS :
    BEGIN
        Count := SIZEOF(QUERY(temp <* Objects | NOT(' IFC4. IFCPROCESS' IN
TYPEOF(temp))));
        RETURN(Count = 0);
    END;
    IfcObjectTypeEnum.CONTROL :
    BEGIN
        Count := SIZEOF(QUERY(temp <* Objects | NOT(' IFC4. IFCCONTROL' IN
TYPEOF(temp))));
        RETURN(Count = 0);
    END;
    IfcObjectTypeEnum.RESOURCE :
    BEGIN
        Count := SIZEOF(QUERY(temp <* Objects | NOT(' IFC4. IFCRESOURCE' IN
TYPEOF(temp))));
        RETURN(Count = 0);
    END;
    IfcObjectTypeEnum.ACTOR :
    BEGIN
        Count := SIZEOF(QUERY(temp <* Objects | NOT(' IFC4. IFCACTOR' IN
TYPEOF(temp))));
        RETURN(Count = 0);
    END;
    IfcObjectTypeEnum.GROUP :
    BEGIN
        Count := SIZEOF(QUERY(temp <* Objects | NOT(' IFC4. IFCGROUP' IN
TYPEOF(temp))));
        RETURN(Count = 0);
    END;
    IfcObjectTypeEnum.PROJECT :
    BEGIN
        Count := SIZEOF(QUERY(temp <* Objects | NOT(' IFC4. IFCPROJECT' IN
TYPEOF(temp))));
        RETURN(Count = 0);
    END;
    OTHERWISE : RETURN(?);
END_CASE;
END_FUNCTION;

FUNCTION IfcCorrectUnitAssignment
(Units : SET [1:?] OF IfcUnit)
: LOGICAL;

```

---

```

LOCAL
  NamedUnitNumber    : INTEGER := 0;
  DerivedUnitNumber   : INTEGER := 0;
  MonetaryUnitNumber : INTEGER := 0;
  NamedUnitNames      : SET OF IfcUnitEnum := [];
  DerivedUnitNames    : SET OF IfcDerivedUnitEnum := [];
END_LOCAL;

  NamedUnitNumber := SIZEOF(QUERY(temp <* Units | (' IFC4. IFCNAMEDUNIT' IN
  TYPEOF(temp)) AND NOT(temp\IfcNamedUnit.UnitType = IfcUnitEnum.USERDEFINED)));
  DerivedUnitNumber := SIZEOF(QUERY(temp <* Units | (' IFC4. IFCDERIVEDUNIT' IN
  TYPEOF(temp)) AND NOT(temp\IfcDerivedUnit.UnitType =
  IfcDerivedUnitEnum.USERDEFINED)));
  MonetaryUnitNumber := SIZEOF(QUERY(temp <* Units | ' IFC4. IFCMONETARYUNIT' IN
  TYPEOF(temp)));

  REPEAT i := 1 TO SIZEOF(Units);
    IF ((' IFC4. IFCNAMEDUNIT' IN TYPEOF(Units[i])) AND
  NOT(Units[i]\IfcNamedUnit.UnitType = IfcUnitEnum.USERDEFINED)) THEN
      NamedUnitNames := NamedUnitNames + Units[i]\IfcNamedUnit.UnitType;
    END_IF;
    IF ((' IFC4. IFCDERIVEDUNIT' IN TYPEOF(Units[i])) AND
  NOT(Units[i]\IfcDerivedUnit.UnitType = IfcDerivedUnitEnum.USERDEFINED)) THEN
      DerivedUnitNames := DerivedUnitNames + Units[i]\IfcDerivedUnit.UnitType;
    END_IF;
  END_REPEAT;

  RETURN((SIZEOF(NamedUnitNames) = NamedUnitNumber) AND (SIZEOF(DerivedUnitNames)
  = DerivedUnitNumber) AND (MonetaryUnitNumber <= 1));
END_FUNCTION;

FUNCTION IfcCrossProduct
(Arg1, Arg2 : IfcDirection)
  : IfcVector;
LOCAL
  Mag : REAL;
  Res : IfcDirection;
  V1,V2 : LIST[3:3] OF REAL;
  Result : IfcVector;
END_LOCAL;

  IF (NOT EXISTS (Arg1) OR (Arg1.Dim = 2)) OR (NOT EXISTS (Arg2) OR (Arg2.Dim = 2))
  THEN
    RETURN(?);

```



---

```

ELSE
  BEGIN
    V1 := IfcNormalise(Arg1)\IfcDirection.DirectionRatios;

    V2 := IfcNormalise(Arg2)\IfcDirection.DirectionRatios;
    Res := IfcRepresentationItem() || IfcGeometricRepresentationItem ()
          || IfcDirection([(V1[2]*V2[3] - V1[3]*V2[2]), (V1[3]*V2[1] -
V1[1]*V2[3]), (V1[1]*V2[2] - V1[2]*V2[1])]);
    Mag := 0.0;
    REPEAT i := 1 TO 3;
      Mag := Mag + Res.DirectionRatios[i]*Res.DirectionRatios[i];
    END_REPEAT;
    IF (Mag > 0.0) THEN
      Result := IfcRepresentationItem() || IfcGeometricRepresentationItem () ||
IfcVector(Res, SQRT(Mag));
    ELSE
      Result := IfcRepresentationItem() || IfcGeometricRepresentationItem () ||
IfcVector(Arg1, 0.0);
    END_IF;
    RETURN(Result);
  END;
END_IF;
END_FUNCTION;

```

```

FUNCTION IfcCurveDim
(Curve : IfcCurve)
  : IfcDimensionCount;

  IF (' IFC4. IFCLINE' IN TYPEOF(Curve))
    THEN RETURN(Curve\IfcLine.Pnt.Dim);
  END_IF;
  IF (' IFC4. IFCCONIC' IN TYPEOF(Curve))
    THEN RETURN(Curve\IfcConic.Position.Dim);
  END_IF;
  IF (' IFC4. IFCPOLYLINE' IN TYPEOF(Curve))
    THEN RETURN(Curve\IfcPolyline.Points[1].Dim);
  END_IF;
  IF (' IFC4. IFCTRIMMEDCURVE' IN TYPEOF(Curve))
    THEN RETURN(IfcCurveDim(Curve\IfcTrimmedCurve.BasisCurve));
  END_IF;
  IF (' IFC4. IFCCOMPOSITECURVE' IN TYPEOF(Curve))
    THEN RETURN(Curve\IfcCompositeCurve.Segments[1].Dim);
  END_IF;
  IF (' IFC4. IFCBSPLINECURVE' IN TYPEOF(Curve))

```

---

```

        THEN RETURN(Curve\IfcBSplineCurve.ControlPointsList[1].Dim);
    END_IF;
    IF (' IFC4. IFCOFFSETCURVE2D' IN TYPEOF(Curve))
        THEN RETURN(2);
    END_IF;
    IF (' IFC4. IFCOFFSETCURVE3D' IN TYPEOF(Curve))
        THEN RETURN(3);
    END_IF;
    IF (' IFC4. IFCPCURVE' IN TYPEOF(Curve))
        THEN RETURN(3);
    END_IF;
    IF (' IFC4. IFCINDEXEDPOLYCURVE' IN TYPEOF(Curve))
        THEN RETURN(Curve\IfcIndexedPolyCurve.Points.Dim);
    END_IF;
    RETURN (?);
END_FUNCTION;

```

```

FUNCTION IfcCurveWeightsPositive
( B: IfcRationalBSplineCurveWithKnots)
: BOOLEAN;

    LOCAL
        Result : BOOLEAN := TRUE;
    END_LOCAL;

    REPEAT i := 0 TO B.UpperIndexOnControlPoints;
        IF B.Weights[i] <= 0.0 THEN
            Result := FALSE;
            RETURN(Result);
        END_IF;
    END_REPEAT;
    RETURN(Result);
END_FUNCTION;

```

```

FUNCTION IfcDeriveDimensionalExponents
(UnitElements : SET [1:?] OF IfcDerivedUnitElement)
: IfcDimensionalExponents;
    LOCAL
        Result : IfcDimensionalExponents :=
            IfcDimensionalExponents(0, 0, 0, 0, 0, 0, 0);
    END_LOCAL;
    REPEAT i := LOINDEX(UnitElements) TO HIINDEX(UnitElements);
        Result.LengthExponent := Result.LengthExponent +
            (UnitElements[i].Exponent *

```

---

```

        UnitElements[i].Unit.Dimensions.LengthExponent);
Result.MassExponent := Result.MassExponent +
    (UnitElements[i].Exponent *
    UnitElements[i].Unit.Dimensions.MassExponent);
Result.TimeExponent := Result.TimeExponent +
    (UnitElements[i].Exponent *
    UnitElements[i].Unit.Dimensions.TimeExponent);
Result.ElectricCurrentExponent := Result.ElectricCurrentExponent +
    (UnitElements[i].Exponent *
    UnitElements[i].Unit.Dimensions.ElectricCurrentExponent);
Result.ThermodynamicTemperatureExponent :=
Result.ThermodynamicTemperatureExponent +
    (UnitElements[i].Exponent *
    UnitElements[i].Unit.Dimensions.ThermodynamicTemperatureExponent);
Result.AmountOfSubstanceExponent := Result.AmountOfSubstanceExponent +
    (UnitElements[i].Exponent *
    UnitElements[i].Unit.Dimensions.AmountOfSubstanceExponent);
Result.LuminousIntensityExponent := Result.LuminousIntensityExponent +
    (UnitElements[i].Exponent *
    UnitElements[i].Unit.Dimensions.LuminousIntensityExponent);
END_REPEAT;
RETURN (Result);
END_FUNCTION;

```

```

FUNCTION IfcDimensionsForSiUnit
(n : IfcSiUnitName )      : IfcDimensionalExponents;
CASE n OF
    METRE          : RETURN (IfcDimensionalExponents
        (1, 0, 0, 0, 0, 0, 0));
    SQUARE_METRE   : RETURN (IfcDimensionalExponents
        (2, 0, 0, 0, 0, 0, 0));
    CUBIC_METRE    : RETURN (IfcDimensionalExponents
        (3, 0, 0, 0, 0, 0, 0));
    GRAM           : RETURN (IfcDimensionalExponents
        (0, 1, 0, 0, 0, 0, 0));
    SECOND         : RETURN (IfcDimensionalExponents
        (0, 0, 1, 0, 0, 0, 0));
    AMPERE         : RETURN (IfcDimensionalExponents
        (0, 0, 0, 1, 0, 0, 0));
    KELVIN         : RETURN (IfcDimensionalExponents
        (0, 0, 0, 0, 1, 0, 0));
    MOLE           : RETURN (IfcDimensionalExponents
        (0, 0, 0, 0, 0, 1, 0));
    CANDELA        : RETURN (IfcDimensionalExponents

```

---

```

                                (0, 0, 0, 0, 0, 0, 1));
RADIAN      : RETURN (IfcDimensionalExponents
                                (0, 0, 0, 0, 0, 0, 0));
STERADIAN   : RETURN (IfcDimensionalExponents
                                (0, 0, 0, 0, 0, 0, 0));
HERTZ       : RETURN (IfcDimensionalExponents
                                (0, 0, -1, 0, 0, 0, 0));
NEWTON      : RETURN (IfcDimensionalExponents
                                (1, 1, -2, 0, 0, 0, 0));
PASCAL      : RETURN (IfcDimensionalExponents
                                (-1, 1, -2, 0, 0, 0, 0));
JOULE       : RETURN (IfcDimensionalExponents
                                (2, 1, -2, 0, 0, 0, 0));
WATT        : RETURN (IfcDimensionalExponents
                                (2, 1, -3, 0, 0, 0, 0));
COULOMB     : RETURN (IfcDimensionalExponents
                                (0, 0, 1, 1, 0, 0, 0));
VOLT        : RETURN (IfcDimensionalExponents
                                (2, 1, -3, -1, 0, 0, 0));
FARAD       : RETURN (IfcDimensionalExponents
                                (-2, -1, 4, 2, 0, 0, 0));
OHM         : RETURN (IfcDimensionalExponents
                                (2, 1, -3, -2, 0, 0, 0));
SIEMENS     : RETURN (IfcDimensionalExponents
                                (-2, -1, 3, 2, 0, 0, 0));
WEBER       : RETURN (IfcDimensionalExponents
                                (2, 1, -2, -1, 0, 0, 0));
TESLA       : RETURN (IfcDimensionalExponents
                                (0, 1, -2, -1, 0, 0, 0));
HENRY       : RETURN (IfcDimensionalExponents
                                (2, 1, -2, -2, 0, 0, 0));
DEGREE_CELSIUS : RETURN (IfcDimensionalExponents
                                (0, 0, 0, 0, 1, 0, 0));
LUMEN       : RETURN (IfcDimensionalExponents
                                (0, 0, 0, 0, 0, 0, 1));
LUX         : RETURN (IfcDimensionalExponents
                                (-2, 0, 0, 0, 0, 0, 1));
BECQUEREL   : RETURN (IfcDimensionalExponents
                                (0, 0, -1, 0, 0, 0, 0));
GRAY        : RETURN (IfcDimensionalExponents
                                (2, 0, -2, 0, 0, 0, 0));
SIEVERT     : RETURN (IfcDimensionalExponents
                                (2, 0, -2, 0, 0, 0, 0));
OTHERWISE   : RETURN (IfcDimensionalExponents

```

---

```

                                (0, 0, 0, 0, 0, 0, 0));

    END_CASE;
END_FUNCTION;

FUNCTION IfcDotProduct
(Arg1, Arg2 : IfcDirection)
    : REAL;
LOCAL
    Scalar : REAL;
    Vec1, Vec2 : IfcDirection;
    Ndim    : INTEGER;
END_LOCAL;

    IF NOT EXISTS (Arg1) OR NOT EXISTS (Arg2) THEN
        Scalar := ?;
    ELSE
        IF (Arg1.Dim <> Arg2.Dim) THEN
            Scalar := ?;
        ELSE
            BEGIN
                Vec1 := IfcNormalise(Arg1);
                Vec2 := IfcNormalise(Arg2);
                Ndim := Arg1.Dim;
                Scalar := 0.0;
                REPEAT i := 1 TO Ndim;
                    Scalar := Scalar + Vec1.DirectionRatios[i]*Vec2.DirectionRatios[i];
                END_REPEAT;
            END;
        END_IF;
    END_IF;
    RETURN (Scalar);
END_FUNCTION;

FUNCTION IfcFirstProjAxis
(ZAxis, Arg : IfcDirection) : IfcDirection;
LOCAL
    XAxis : IfcDirection;
    V      : IfcDirection;
    Z      : IfcDirection;
    XVec   : IfcVector;
END_LOCAL;

    IF (NOT EXISTS(ZAxis)) THEN
        RETURN (?) ;
    
```

---

```

ELSE
  Z := IfcNormalise(ZAxis);
  IF NOT EXISTS(Arg) THEN
    IF (Z.DirectionRatios <> [1.0, 0.0, 0.0]) THEN
      V := IfcRepresentationItem() || IfcGeometricRepresentationItem () ||
IfcDirection([1.0, 0.0, 0.0]);
    ELSE
      V := IfcRepresentationItem() || IfcGeometricRepresentationItem () ||
IfcDirection([0.0, 1.0, 0.0]);
    END_IF;
  ELSE
    IF (Arg.Dim <> 3) THEN
      RETURN (?);
    END_IF;
    IF ((IfcCrossProduct(Arg, Z).Magnitude) = 0.0) THEN
      RETURN (?);
    ELSE
      V := IfcNormalise(Arg);
    END_IF;
  END_IF;
  XVec := IfcScalarTimesVector(IfcDotProduct(V, Z), Z);
  XAxis := IfcVectorDifference(V, XVec).Orientation;
  XAxis := IfcNormalise(XAxis);
END_IF;
RETURN(XAxis);
END_FUNCTION;

FUNCTION IfcGetBasisSurface
(C : IfcCurveOnSurface) : SET[0:2] OF IfcSurface;

LOCAL
  Surfs : SET[0:2] OF IfcSurface;
  N : INTEGER;
END_LOCAL;

Surfs := [];
IF 'IFC4.IFCPCURVE' IN TYPEOF (C) THEN
  Surfs := [C\IfcPCurve.BasisSurface];
ELSE
  IF 'IFC4.IFCSURFACECURVE' IN TYPEOF (C) THEN
    N := SIZEOF(C\IfcSurfaceCurve.AssociatedGeometry);
    REPEAT i := 1 TO N;
      Surfs
      :=
      Surfs
      +
IfcAssociatedSurface(C\IfcSurfaceCurve.AssociatedGeometry[i]);
    END_REPEAT;
  END_IF;
END_IF;

```

---

```

        END_REPEAT;
    END_IF;
END_IF;
IF 'IFC4.IFCCOMPOSITECURVEONSURFACE' IN TYPEOF (C) THEN

    (* For an IfcCompositeCurveOnSurface the BasisSurface is the intersection of
    the BasisSurface of all the segments. *)

    N := SIZEOF(C\IfcCompositeCurve.Segments);
    Surfs := IfcGetBasisSurface(C\IfcCompositeCurve.Segments[1].ParentCurve);
    IF N > 1 THEN
        REPEAT i := 2 TO N;
            Surfs := Surfs *
IfcGetBasisSurface(C\IfcCompositeCurve.Segments[i].ParentCurve);
        END_REPEAT;
    END_IF;
END_IF;
RETURN(Surfs);
END_FUNCTION;

FUNCTION IfcListToArray
(Lis : LIST [0:?] OF GENERIC : T;
    Low,U : INTEGER) : ARRAY OF GENERIC : T;
LOCAL
    N : INTEGER;
    Res : ARRAY [Low:U] OF GENERIC : T;
END_LOCAL;

N := SIZEOF(Lis);
IF (N <> (U-Low +1)) THEN
    RETURN(?);
ELSE
    Res := [Lis[1] : N];
    REPEAT i := 2 TO N;
        Res[Low+i-1] := Lis[i];
    END_REPEAT;
    RETURN(Res);
END_IF;
END_FUNCTION;

FUNCTION IfcLoopHeadToTail
(ALoop : IfcEdgeLoop) : LOGICAL;
LOCAL
    N : INTEGER;

```

---

```

    P : LOGICAL := TRUE;
END_LOCAL;

    N := SIZEOF (ALoop.EdgeList);
    REPEAT i := 2 TO N;
        P := P AND (ALoop.EdgeList[i-1].EdgeEnd :=:
                    ALoop.EdgeList[i].EdgeStart);
    END_REPEAT;
    RETURN (P);
END_FUNCTION;

FUNCTION IfcMakeArrayOfArray
(Lis : LIST[1:?] OF LIST [1:?] OF GENERIC : T;
 Low1, U1, Low2, U2 : INTEGER):
ARRAY [Low1:U1] OF ARRAY [Low2:U2] OF GENERIC : T;

    LOCAL
        Res : ARRAY[Low1:U1] OF ARRAY [Low2:U2] OF GENERIC : T;
    END_LOCAL;

    (* Check input dimensions for consistency *)
    IF (U1-Low1+1) <> SIZEOF(Lis) THEN
        RETURN (?);
    END_IF;
    IF (U2 - Low2 + 1 ) <> SIZEOF(Lis[1]) THEN
        RETURN (?) ;
    END_IF;

    (* Initialise Res with values from Lis[1] *)
    Res := [IfcListToArray(Lis[1], Low2, U2) : (U1-Low1 + 1)];
    REPEAT i := 2 TO HIINDEX(Lis);
        IF (U2-Low2+1) <> SIZEOF(Lis[i]) THEN
            RETURN (?);
        END_IF;
        Res[Low1+i-1] := IfcListToArray(Lis[i], Low2, U2);
    END_REPEAT;
    RETURN (Res);
END_FUNCTION;

FUNCTION IfcMlsTotalThickness
(LayerSet : IfcMaterialLayerSet) : IfcLengthMeasure;
    LOCAL
        Max : IfcLengthMeasure := LayerSet.MaterialLayers[1].LayerThickness;
    END_LOCAL;

```



---

```

IF SIZEOF(LayerSet.MaterialLayers) > 1 THEN
  REPEAT i := 2 TO HIINDEX(LayerSet.MaterialLayers);
    Max := Max + LayerSet.MaterialLayers[i].LayerThickness;
  END_REPEAT;
END_IF;
RETURN (Max);
END_FUNCTION;

FUNCTION IfcNormalise
(Arg : IfcVectorOrDirection)
  : IfcVectorOrDirection;
LOCAL
  Ndim : INTEGER;
  V    : IfcDirection
      := IfcRepresentationItem() || IfcGeometricRepresentationItem () ||
IfcDirection([1., 0.]);
  Vec  : IfcVector
      := IfcRepresentationItem() || IfcGeometricRepresentationItem () ||
IfcVector (
      IfcRepresentationItem() || IfcGeometricRepresentationItem () ||
IfcDirection([1., 0.]), 1.);
  Mag  : REAL;
  Result : IfcVectorOrDirection
      := V;
END_LOCAL;

IF NOT EXISTS (Arg) THEN
  RETURN (?);
ELSE
  IF 'IFC4.IFCVECTOR' IN TYPEOF(Arg) THEN
    BEGIN
      Ndim := Arg\IfcVector.Dim;
      V.DirectionRatios := Arg\IfcVector.Orientation.DirectionRatios;
      Vec.Magnitude := Arg\IfcVector.Magnitude;
      Vec.Orientation := V;
      IF Arg\IfcVector.Magnitude = 0.0 THEN
        RETURN(?);
      ELSE
        Vec.Magnitude := 1.0;
      END_IF;
    END;
  ELSE
    BEGIN

```

---

```

        Ndim := Arg\IfcDirection.Dim;
        V.DirectionRatios := Arg\IfcDirection.DirectionRatios;
    END;
END_IF;

Mag := 0.0;
REPEAT i := 1 TO Ndim;
    Mag := Mag + V.DirectionRatios[i]*V.DirectionRatios[i];
END_REPEAT;
IF Mag > 0.0 THEN
    Mag := SQRT(Mag);
    REPEAT i := 1 TO Ndim;
        V.DirectionRatios[i] := V.DirectionRatios[i]/Mag;
    END_REPEAT;
    IF 'IFC4.IFCVECTOR' IN TYPEOF(arg) THEN
        Vec.Orientation := V;
        Result := Vec;
    ELSE
        Result := V;
    END_IF;
ELSE
    RETURN(?);
END_IF;
END_IF;
RETURN (Result);
END_FUNCTION;

FUNCTION IfcOrthogonalComplement
(Vec : IfcDirection)
    : IfcDirection;
LOCAL
    Result : IfcDirection ;
END_LOCAL;
IF NOT EXISTS (Vec) OR (Vec.Dim <> 2) THEN
    RETURN(?);
ELSE
    Result := IfcRepresentationItem() || IfcGeometricRepresentationItem () ||
IfcDirection([-Vec.DirectionRatios[2], Vec.DirectionRatios[1]]);
    RETURN(Result);
END_IF;
END_FUNCTION;

FUNCTION IfcPathHeadToTail
(APath : IfcPath) : LOGICAL;

```

---

```

LOCAL
  N : INTEGER := 0;
  P : LOGICAL := UNKNOWN;
END_LOCAL;
  N := SIZEOF (APath.EdgeList);
REPEAT i := 2 TO N;
  P := P AND (APath.EdgeList[i-1].EdgeEnd :=:
              APath.EdgeList[i].EdgeStart);
END_REPEAT;
RETURN (P);
END_FUNCTION;

FUNCTION IfcPointListDim
(PointList : IfcCartesianPointList)
  : IfcDimensionCount;

  IF ('IFC4.IFCCARTESIANPOINTLIST2D' IN TYPEOF(PointList))
    THEN RETURN(2);
  END_IF;
  IF ('IFC4.IFCCARTESIANPOINTLIST3D' IN TYPEOF(PointList))
    THEN RETURN(3);
  END_IF;
  RETURN (?);
END_FUNCTION;

FUNCTION IfcSameAxis2Placement
(ap1, ap2 : IfcAxis2Placement; Epsilon : REAL)
  : LOGICAL ;

  RETURN (IfcSameDirection(ap1.P[1], ap2.P[1], Epsilon) AND
          IfcSameDirection(ap1.P[2], ap2.P[2], Epsilon) AND
          IfcSameCartesianPoint(ap1.Location, ap1.Location, Epsilon));
END_FUNCTION;

FUNCTION IfcSameCartesianPoint
(cp1, cp2 : IfcCartesianPoint; Epsilon : REAL)
  : LOGICAL;

LOCAL
  cp1x : REAL := cp1.Coordinates[1];
  cp1y : REAL := cp1.Coordinates[2];
  cp1z : REAL := 0;
  cp2x : REAL := cp2.Coordinates[1];
  cp2y : REAL := cp2.Coordinates[2];

```

---

```

    cp2z : REAL := 0;
END_LOCAL;

IF (SIZEOF(cp1.Coordinates) > 2) THEN
    cp1z := cp1.Coordinates[3];
END_IF;

IF (SIZEOF(cp2.Coordinates) > 2) THEN
    cp2z := cp2.Coordinates[3];
END_IF;

RETURN (IfcSameValue(cp1x, cp2x, Epsilon) AND
        IfcSameValue(cp1y, cp2y, Epsilon) AND
        IfcSameValue(cp1z, cp2z, Epsilon));
END_FUNCTION;

FUNCTION IfcSameDirection
(dir1, dir2 : IfcDirection; Epsilon : REAL)
: LOGICAL;
LOCAL
    dir1x : REAL := dir1.DirectionRatios[1];
    dir1y : REAL := dir1.DirectionRatios[2];
    dir1z : REAL := 0;
    dir2x : REAL := dir2.DirectionRatios[1];
    dir2y : REAL := dir2.DirectionRatios[2];
    dir2z : REAL := 0;
END_LOCAL;

IF (SIZEOF(dir1.DirectionRatios) > 2) THEN
    dir1z := dir1.DirectionRatios[3];
END_IF;

IF (SIZEOF(dir2.DirectionRatios) > 2) THEN
    dir2z := dir2.DirectionRatios[3];
END_IF;

RETURN (IfcSameValue(dir1x, dir2x, Epsilon) AND
        IfcSameValue(dir1y, dir2y, Epsilon) AND
        IfcSameValue(dir1z, dir2z, Epsilon));
END_FUNCTION;

FUNCTION IfcSameValidPrecision
(Epsilon1, Epsilon2 : REAL) : LOGICAL ;
LOCAL

```

---

```

    ValidEps1, ValidEps2 : REAL;
    DefaultEps           : REAL := 0.000001;
    DerivationOfEps      : REAL := 1.001;
    UpperEps             : REAL := 1.0;
END_LOCAL;

    ValidEps1 := NVL(Epsilon1, DefaultEps);
    ValidEps2 := NVL(Epsilon2, DefaultEps);
    RETURN ((0.0 < ValidEps1) AND (ValidEps1 <= (DerivationOfEps * ValidEps2)) AND
            (ValidEps2 <= (DerivationOfEps * ValidEps1)) AND (ValidEps2 <
UpperEps));
END_FUNCTION;

FUNCTION IfcSameValue
(Value1, Value2 : REAL; Epsilon : REAL)
: LOGICAL;
LOCAL
    ValidEps    : REAL;
    DefaultEps   : REAL := 0.000001;
END_LOCAL;

    ValidEps := NVL(Epsilon, DefaultEps);
    RETURN ((Value1 + ValidEps > Value2) AND (Value1 < Value2 + ValidEps));
END_FUNCTION;

FUNCTION IfcScalarTimesVector
(Scalar : REAL; Vec : IfcVectorOrDirection)
: IfcVector;
LOCAL
    V : IfcDirection;
    Mag : REAL;
    Result : IfcVector;
END_LOCAL;

    IF NOT EXISTS (Scalar) OR NOT EXISTS (Vec) THEN
        RETURN (?) ;
    ELSE
        IF 'IFC4.IFCVECTOR' IN TYPEOF (Vec) THEN
            V := Vec\IfcVector.Orientation;
            Mag := Scalar * Vec\IfcVector.Magnitude;
        ELSE
            V := Vec;
            Mag := Scalar;
        END_IF;
    END_IF;

```

---

```

    IF (Mag < 0.0 ) THEN
        REPEAT i := 1 TO SIZEOF(V.DirectionRatios);
            V.DirectionRatios[i] := -V.DirectionRatios[i];
        END_REPEAT;
        Mag := -Mag;
    END_IF;
    Result := IfcRepresentationItem() || IfcGeometricRepresentationItem () ||
IfcVector(IfcNormalise(V), Mag);
    END_IF;
    RETURN (Result);
END_FUNCTION;

```

```

FUNCTION IfcSecondProjAxis
(ZAxis, XAxis, Arg: IfcDirection)
    : IfcDirection;
LOCAL
    YAxis : IfcVector;
    V      : IfcDirection;
    Temp   : IfcVector;
END_LOCAL;

```

```

    IF NOT EXISTS(Arg) THEN
        V := IfcRepresentationItem() || IfcGeometricRepresentationItem () ||
IfcDirection([0.0, 1.0, 0.0]);
    ELSE
        V := Arg;
    END_IF;
    Temp := IfcScalarTimesVector(IfcDotProduct(V, ZAxis), ZAxis);
    YAxis := IfcVectorDifference(V, Temp);
    Temp := IfcScalarTimesVector(IfcDotProduct(V, XAxis), XAxis);
    YAxis := IfcVectorDifference(YAxis, Temp);
    YAxis := IfcNormalise(YAxis);
    RETURN(YAxis.Orientation);
END_FUNCTION;

```

```

FUNCTION IfcShapeRepresentationTypes
(RepType : IfcLabel; Items : SET OF IfcRepresentationItem) : LOGICAL;

```

```

    LOCAL
        Count : INTEGER := 0;
    END_LOCAL;

```

```

    CASE RepType OF
        'Point' :

```

---

```

BEGIN
    Count := SIZEOF(QUERY(temp <* Items | (' IFC4. IFCPOINT' IN TYPEOF(temp))));
END;

'PointCloud' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | (' IFC4. IFCCARTESIANPOINTLIST3D' IN
TYPEOF(temp))));
END;

'Curve' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | (' IFC4. IFCCURVE' IN TYPEOF(temp))));
END;

'Curve2D' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | (' IFC4. IFCCURVE' IN TYPEOF(temp))
AND (temp\IfcCurve.Dim = 2)));
END;

'Curve3D' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | (' IFC4. IFCCURVE' IN TYPEOF(temp))
AND (temp\IfcCurve.Dim = 3)));
END;

'Surface' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | (' IFC4. IFCSURFACE' IN
TYPEOF(temp))));
END;

'Surface2D' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | (' IFC4. IFCSURFACE' IN TYPEOF(temp))
AND (temp\IfcSurface.Dim = 2)));
END;

'Surface3D' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | (' IFC4. IFCSURFACE' IN TYPEOF(temp))
AND (temp\IfcSurface.Dim = 3)));
END;

```

---

```

'FillArea' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | (' IFC4. IFCANNOTATIONFILLAREA' IN
    TYPEOF(temp))));
END;

'Text' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | (' IFC4. IFCTEXTLITERAL' IN
    TYPEOF(temp))));
END;

'AdvancedSurface' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | ' IFC4. IFCBSPLINESURFACE' IN
    TYPEOF(temp))));
END;

'Annotation2D' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | (
        SIZEOF(TYPEOF(temp) * [
            ' IFC4. IFCPOINT',
            ' IFC4. IFCCURVE',
            ' IFC4. IFCGEOMETRICCURVESET',
            ' IFC4. IFCANNOTATIONFILLAREA',
            ' IFC4. IFCTEXTLITERAL' ]) = 1)
    ));
END;

'GeometricSet' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | (' IFC4. IFCGEOMETRICSET' IN
    TYPEOF(temp))
        OR (' IFC4. IFCPOINT' IN TYPEOF(temp))
        OR (' IFC4. IFCCURVE' IN TYPEOF(temp))
        OR (' IFC4. IFCSURFACE' IN TYPEOF(temp))));
END;

'GeometricCurveSet' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | (' IFC4. IFCGEOMETRICCURVESET' IN
    TYPEOF(temp))

```



---

```

        OR ('IFC4.IFCGEOMETRICSET' IN TYPEOF(temp))
        OR ('IFC4.IFCPOINT' IN TYPEOF(temp))
        OR ('IFC4.IFCCURVE' IN TYPEOF(temp))));
REPEAT i:=1 TO HIINDEX(Items);
    IF ('IFC4.IFCGEOMETRICSET' IN TYPEOF(Items[i]))
    THEN
        IF (SIZEOF(QUERY(temp <* Items[i]\IfcGeometricSet.Elements |
'IFC4.IFCSURFACE' IN TYPEOF(temp))) > 0)
        THEN
            Count := Count - 1;
        END_IF;
    END_IF;
END_REPEAT;
END;

'Tessellation' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | 'IFC4.IFCTESSELLATEDITEM' IN
TYPEOF(temp)));
END;

'SurfaceOrSolidModel' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | SIZEOF([
        'IFC4.IFCTESSELLATEDITEM',
        'IFC4.IFCSHELLBASEDSURFACEMODEL',
        'IFC4.IFCFACEBASEDSURFACEMODEL',
        'IFC4.IFCSOLIDMODEL' ] * TYPEOF(temp)) >= 1
    ));
END;

'SurfaceModel' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | SIZEOF([
        'IFC4.IFCTESSELLATEDITEM',
        'IFC4.IFCSHELLBASEDSURFACEMODEL',
        'IFC4.IFCFACEBASEDSURFACEMODEL' ] * TYPEOF(temp)) >= 1
    ));
END;

'SolidModel' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | ('IFC4.IFCSOLIDMODEL' IN
TYPEOF(temp))));

```

---

```

END;

'SweptSolid' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | (SIZEOF([
        'IFC4. IFCEXTRUDEDAREASOLID',
        'IFC4. IFCREVOLVEDAREASOLID' ] * TYPEOF(temp)) >= 1
    ) AND (SIZEOF([
        'IFC4. IFCEXTRUDEDAREASOLIDTAPERED',
        'IFC4. IFCREVOLVEDAREASOLIDTAPERED' ] * TYPEOF(temp)) = 0
    )
    ));
END;

'AdvancedSweptSolid' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | SIZEOF([
        'IFC4. IFCSWEPTAREASOLID',
        'IFC4. IFCSWEPTDISKSOLID' ] * TYPEOF(temp)) >= 1
    ));
END;

'CSG' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | SIZEOF([
        'IFC4. IFCBOOLEANRESULT',
        'IFC4. IFCCSGPRIMITIVE3D',
        'IFC4. IFCCSGSOLID' ] * TYPEOF(temp)) >= 1
    ));
END;

'Clipping' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | SIZEOF([
        'IFC4. IFCCSGSOLID',
        'IFC4. IFCBOOLEANCLIPPINGRESULT' ] * TYPEOF(temp)) >= 1
    ));
END;

'Brep' :
BEGIN
    Count := SIZEOF(QUERY(temp <* Items | ('IFC4. IFCFACETEDBREP' IN
    TYPEOF(temp))));
END;

```

---

```

    'AdvancedBrep' :
    BEGIN
        Count := SIZEOF(QUERY(temp <* Items | (' IFC4. IFCMANIFOLDSOLIDBREP' IN
        TYPEOF(temp))));
    END;

    'BoundingBox' :
    BEGIN
        Count := SIZEOF(QUERY(temp <* Items | (' IFC4. IFCBOUNDINGBOX' IN
        TYPEOF(temp))));
        IF (SIZEOF(Items) > 1)
        THEN
            Count := 0;
        END_IF;
    END;

    'SectionedSpine' :
    BEGIN
        Count := SIZEOF(QUERY(temp <* Items | (' IFC4. IFCSECTIONEDSPINE' IN
        TYPEOF(temp))));
    END;

    'LightSource' :
    BEGIN
        Count := SIZEOF(QUERY(temp <* Items | (' IFC4. IFCLIGHTSOURCE' IN
        TYPEOF(temp))));
    END;

    'MappedRepresentation' :
    BEGIN
        Count := SIZEOF(QUERY(temp <* Items | (' IFC4. IFCMAPPEDITEM' IN
        TYPEOF(temp))));
    END;

    OTHERWISE : RETURN(?);
END_CASE;
RETURN (Count = SIZEOF(Items));
END_FUNCTION;

FUNCTION IfcSurfaceWeightsPositive
( B: IfcRationalBSplineSurfaceWithKnots)
: BOOLEAN;

```

---

```

LOCAL
    Result : BOOLEAN := TRUE;
END_LOCAL;

REPEAT i := 0 TO B\IfcBSplineSurface.UUpper;
    REPEAT j := 0 TO B\IfcBSplineSurface.VUpper;
        IF (B.Weights[i][j] <= 0.0) THEN
            Result := FALSE;
            RETURN(Result);
        END_IF;
    END_REPEAT;
END_REPEAT;
RETURN(Result);
END_FUNCTION;

FUNCTION IfcTaperedSweptAreaProfiles
(StartArea, EndArea : IfcProfileDef)
: LOGICAL;

LOCAL
    Result : LOGICAL := FALSE;
END_LOCAL;

IF (' IFC4. IFCPARAMETERIZEDPROFILEDEF' IN TYPEOF(StartArea)) THEN
    IF (' IFC4. IFCDERIVEDPROFILEDEF' IN TYPEOF(EndArea)) THEN
        Result := (StartArea :=: EndArea\IfcDerivedProfileDef.ParentProfile);
    ELSE
        Result := (TYPEOF(StartArea) = TYPEOF(EndArea));
    END_IF;
ELSE
    IF (' IFC4. IFCDERIVEDPROFILEDEF' IN TYPEOF(EndArea)) THEN
        Result := (StartArea :=: EndArea\IfcDerivedProfileDef.ParentProfile);
    ELSE
        Result := FALSE;
    END_IF;
END_IF;

RETURN(Result);
END_FUNCTION;

FUNCTION IfcTopologyRepresentationTypes
(RepType : IfcLabel; Items : SET OF IfcRepresentationItem) : LOGICAL;

LOCAL

```

---

```

    Count : INTEGER := 0;
END_LOCAL;

CASE RepType OF
'Vertex' :
    BEGIN
        Count := SIZEOF(QUERY(temp <* Items |
                                (' IFC4. IFCVERTEX' IN TYPEOF(temp))));
    END;
'Edge' :
    BEGIN
        Count := SIZEOF(QUERY(temp <* Items |
                                (' IFC4. IFCEDGE' IN TYPEOF(temp))));
    END;
'Path' :
    BEGIN
        Count := SIZEOF(QUERY(temp <* Items |
                                (' IFC4. IFCPATH' IN TYPEOF(temp))));
    END;
'Face' :
    BEGIN
        Count := SIZEOF(QUERY(temp <* Items |
                                (' IFC4. IFCFACE' IN TYPEOF(temp))));
    END;
'Shell' :
    BEGIN
        Count := SIZEOF(QUERY(temp <* Items |
                                (' IFC4. IFCOPENSHELL' IN TYPEOF(temp))
                                OR (' IFC4. IFCCLOSEDSHELL' IN TYPEOF(temp))));
    END;
'Undefined' : RETURN(TRUE);
OTHERWISE : RETURN(?);
END_CASE;
RETURN (Count = SIZEOF(Items));
END_FUNCTION;

```

```

FUNCTION IfcUniqueDefinitionNames
(Relations : SET [1:?] OF IfcRelDefinesByProperties)
:LOGICAL;

```

```

LOCAL
    Definition : IfcPropertySetDefinitionSelect;
    DefinitionSet : IfcPropertySetDefinitionSet;
    Properties : SET OF IfcPropertySetDefinition := [];

```

---

```

    Result : LOGICAL;
END_LOCAL;

IF SIZEOF(Relations) = 0 THEN
    RETURN(TRUE);
END_IF;

REPEAT i:=1 TO HIINDEX(Relations);
    Definition := Relations[i].RelatingPropertyDefinition;
    IF 'IFC4.IFCPROPERTYSETDEFINITION' IN TYPEOF(Definition) THEN
        Properties := Properties + Definition;
    ELSE
        IF 'IFC4.IFCPROPERTYSETDEFINITIONSET' IN TYPEOF(Definition) THEN
            BEGIN
                DefinitionSet := Definition;
                REPEAT j:= 1 TO HIINDEX(DefinitionSet);
                    Properties := Properties + DefinitionSet[j];
                END_REPEAT;
            END;
        END_IF;
    END_IF;
END_REPEAT;

Result := IfcUniquePropertySetNames(Properties);
RETURN (Result);
END_FUNCTION;

FUNCTION IfcUniquePropertyName
(Properties : SET [1:?] OF IfcProperty)
:LOGICAL;

LOCAL
    Names : SET OF IfcIdentifier := [];
END_LOCAL;

REPEAT i:=1 TO HIINDEX(Properties);
    Names := Names + Properties[i].Name;
END_REPEAT;

RETURN (SIZEOF(Names) = SIZEOF(Properties));
END_FUNCTION;

FUNCTION IfcUniquePropertySetNames
(Properties : SET [1:?] OF IfcPropertySetDefinition)

```

---

```

:LOGICAL;

LOCAL
  Names : SET OF IfcLabel := [];
  Unnamed : INTEGER := 0;
END_LOCAL;

REPEAT i:=1 TO HIINDEX(Properties);
  IF 'IFC4.IFCPROPERTYSET' IN TYPEOF(Properties[i]) THEN
    Names := Names + Properties[i]\IfcRoot.Name;
  ELSE
    Unnamed := Unnamed + 1;
  END_IF;
END_REPEAT;

RETURN (SIZEOF(Names) + Unnamed = SIZEOF(Properties));
END_FUNCTION;

FUNCTION IfcUniquePropertyTemplateName
(Properties : SET [1:?] OF IfcPropertyTemplate)
:LOGICAL;

LOCAL
  Names : SET OF IfcLabel := [];
END_LOCAL;

REPEAT i:=1 TO HIINDEX(Properties);
  Names := Names + Properties[i].Name;
END_REPEAT;
RETURN (SIZEOF(Names) = SIZEOF(Properties));
END_FUNCTION;

FUNCTION IfcUniqueQuantityNames
(Properties : SET [1:?] OF IfcPhysicalQuantity)
:LOGICAL;

LOCAL
  Names : SET OF IfcLabel := [];
END_LOCAL;

REPEAT i:=1 TO HIINDEX(Properties);
  Names := Names + Properties[i].Name;
END_REPEAT;
RETURN (SIZEOF(Names) = SIZEOF(Properties));

```

---

```

END_FUNCTION;

FUNCTION IfcVectorDifference
(Arg1, Arg2 : IfcVectorOrDirection)
  : IfcVector;
LOCAL
  Result : IfcVector;
  Res, Vec1, Vec2 : IfcDirection;
  Mag, Mag1, Mag2 : REAL;
  Ndim : INTEGER;
END_LOCAL;

IF ((NOT EXISTS (Arg1)) OR (NOT EXISTS (Arg2))) OR (Arg1.Dim <> Arg2.Dim) THEN
  RETURN (?) ;
ELSE
  BEGIN
    IF 'IFC4.IFCVECTOR' IN TYPEOF(Arg1) THEN
      Mag1 := Arg1\IfcVector.Magnitude;
      Vec1 := Arg1\IfcVector.Orientation;
    ELSE
      Mag1 := 1.0;
      Vec1 := Arg1;
    END_IF;
    IF 'IFC4.IFCVECTOR' IN TYPEOF(Arg2) THEN
      Mag2 := Arg2\IfcVector.Magnitude;
      Vec2 := Arg2\IfcVector.Orientation;
    ELSE
      Mag2 := 1.0;
      Vec2 := Arg2;
    END_IF;
    Vec1 := IfcNormalise (Vec1);
    Vec2 := IfcNormalise (Vec2);
    Ndim := SIZEOF(Vec1.DirectionRatios);
    Mag := 0.0;
    Res := IfcRepresentationItem() || IfcGeometricRepresentationItem () ||
IfcDirection([0.0:Ndim]);

    REPEAT i := 1 TO Ndim;
      Res.DirectionRatios[i] := Mag1*Vec1.DirectionRatios[i] -
Mag2*Vec2.DirectionRatios[i];
      Mag := Mag + (Res.DirectionRatios[i]*Res.DirectionRatios[i]);
    END_REPEAT;

    IF (Mag > 0.0 ) THEN

```



---

```

        Result := IfcRepresentationItem() || IfcGeometricRepresentationItem () ||
IfcVector( Res, Sqrt(Mag));
    ELSE
        Result := IfcRepresentationItem() || IfcGeometricRepresentationItem () ||
IfcVector( Vec1, 0.0);
    END_IF;
END;
END_IF;
RETURN (Result);
END_FUNCTION;

FUNCTION IfcVectorSum
(Arg1, Arg2 : IfcVectorOrDirection)
: IfcVector;
LOCAL
    Result : IfcVector;
    Res, Vec1, Vec2 : IfcDirection;
    Mag, Mag1, Mag2 : REAL;
    Ndim : INTEGER;
END_LOCAL;

IF ((NOT EXISTS (Arg1)) OR (NOT EXISTS (Arg2))) OR (Arg1.Dim <> Arg2.Dim) THEN
    RETURN (?);
ELSE
    BEGIN
        IF 'IFC4.IFCVECTOR' IN TYPEOF(Arg1) THEN
            Mag1 := Arg1\IfcVector.Magnitude;
            Vec1 := Arg1\IfcVector.Orientation;
        ELSE
            Mag1 := 1.0;
            Vec1 := Arg1;
        END_IF;
        IF 'IFC4.IFCVECTOR' IN TYPEOF(Arg2) THEN
            Mag2 := Arg2\IfcVector.Magnitude;
            Vec2 := Arg2\IfcVector.Orientation;
        ELSE
            Mag2 := 1.0;
            Vec2 := Arg2;
        END_IF;
        Vec1 := IfcNormalise (Vec1);
        Vec2 := IfcNormalise (Vec2);
        Ndim := SIZEOF(Vec1.DirectionRatios);
        Mag := 0.0;
        Res := IfcRepresentationItem() || IfcGeometricRepresentationItem () ||

```

---

```

IfcDirection([0.0:Ndim]);

    REPEAT i := 1 TO Ndim;
        Res.DirectionRatios[i]      :=      Mag1*Vec1.DirectionRatios[i]      +
Mag2*Vec2.DirectionRatios[i];
        Mag := Mag + (Res.DirectionRatios[i]*Res.DirectionRatios[i]);
    END_REPEAT;

    IF (Mag > 0.0 ) THEN
        Result := IfcRepresentationItem() || IfcGeometricRepresentationItem () ||
IfcVector( Res, SQRT(Mag));
    ELSE
        Result := IfcRepresentationItem() || IfcGeometricRepresentationItem () ||
IfcVector( Vec1, 0.0);
    END_IF;
END;
END_IF;
RETURN (Result);
END_FUNCTION;

RULE IfcRepresentationContextSameWCS FOR
    (IfcGeometricRepresentationContext);
LOCAL
    IsDifferent : LOGICAL := FALSE;
END_LOCAL;
IF (SIZEOF(IfcGeometricRepresentationContext) > 1)
THEN
    REPEAT i := 2 TO HIINDEX(IfcGeometricRepresentationContext);
        IF      (IfcGeometricRepresentationContext[1].WorldCoordinateSystem      :<>:
IfcGeometricRepresentationContext[i].WorldCoordinateSystem)
            THEN
                IsDifferent
                    :=
(NOT(IfcSameValidPrecision(IfcGeometricRepresentationContext[1].Precision,
IfcGeometricRepresentationContext[i].Precision)))
                    OR
(NOT(IfcSameAxis2Placement(IfcGeometricRepresentationContext[1].WorldCoordinate
System,
IfcGeometricRepresentationContext[i].WorldCoordinateSystem,
IfcGeometricRepresentationContext[1].Precision)));
            IF (IsDifferent = TRUE) THEN
                ESCAPE;

```

---

```
        END_IF;
    END_IF;
END_REPEAT;
END_IF;
WHERE
    WR1 : IsDifferent = FALSE;
END_RULE;

RULE IfcSingleProjectInstance FOR
    (IfcProject);

WHERE
    WR1 : SIZEOF(IfcProject) <= 1;
END_RULE;

END_SCHEMA;
```