

Egenskaper och egenskapsuppsättningar (Property och PropertySet)

Säkerställ att varje objekt tilldelas relevant information.

För att så långt som möjligt skapa ett enhetligt arbetssätt gällande egenskaper och egenskapsuppsättningar föreslås här principer för val av och benämning av egenskaper och egenskapsuppsättningar.

Varför?

En effektiv hantering av information förutsätter att egenskaper är definierade och använda på ett strukturerat och enhetligt sätt.

En beställare av information ska kunna ta emot, validera och använda informationen. Det är uppdragsgivarens (den som frågar efter information) ansvar att bestämma strukturen och beskriva den för uppdragstagaren (den som ska leverera informationen).

Detta görs lämpligtvis genom att skapa informationsutbyteskrav (EIR) enligt ISO 19650 för aktuell leverans. Denna EIR bör bland annat beskriva

- vilka egenskaper som ska beskriva olika typer av objekt och
- i vilka egenskapsuppsättningar egenskaperna ska placeras.

Om detta tydligt framgår för uppdragstagaren finns det goda förutsättningar för att informationsutbytet kommer att hålla hög kvalitet och att informationen därmed kommer till nytta. Om krav saknas eller är undermåliga, minskar möjligheterna för att uppdragsgivaren på ett rationellt sätt kunna använda informationen som levereras.

Vad?

En egenskap (*Property*) är något som tilldelas ett objekt – eller “entitet” som de benämns i IFC – för att förtydliga vad det är eller vad det har för funktion. Varje objekt kan ha obegränsat antal egenskaper. Dessa kan grupperas i en eller flera egenskapsuppsättningar (*PropertySet*). Därigenom underlättas hanteringen för att hitta information som hänger ihop för ett specifikt objekt.

Egenskapsuppsättningar inom IFC-standarden

Objektspecifika egenskapsuppsättningar

IFC-standarden innehåller mängder av definierade egenskaper och egenskapsuppsättningar. För de flesta objekt har specifika egenskapsuppsättningar skapats med egenskaper som gäller för respektive typ av objekt. Dessa egenskapsuppsättningar benämns enligt formen *Pset_[Entitet]Common*.

Exempel: *Pset_CoveringCommon* för täckande objekt och *Pset_ColumnCommon* för pelare. Det finns även en del egenskapsuppsättningar för objekttyper (entiteter med en definierad *PredefinedType*), exempelvis *Pset_CoveringFlooring* för typen *FLOORING* (golvbeläggning).

För varje typ av objektredovisas tillhörande egenskapsuppsättningar. Nedan visas ett exempel på egenskapsuppsättningar för *IfcCovering* (täckande objekt).

Pset_Condition AssessmentDate AssessmentCondition AssessmentDescription	Pset_ConstructionAdministration ProcurementMethod SpecificationSectionNumber SubmitalIdentifier	Pset_ConstructionOccurrence InstallationDate ModelNumber TapNumber
Pset_CoveringCommon Reference Status AcousticRating	Pset_CoveringFlooring FLOORING HasNonSkidSurface HasAntiStaticSurface	Pset_CoveringTypeMembrane MEMBRANE NominalInstallationDepth NominalTransverseInclination
Pset_ElementKinematics CyclicPath CyclicRange LinearPath	Pset_EnvironmentalCondition ReferenceAirRelativeHumidity ReferenceEnvironmentTemperature MaximumAtmosphericPressure	Pset_EnvironmentallImpactIndicators Reference FunctionalUnitReference IndicatorsLink
Pset_EnvironmentallImpactValues TotalPrimaryEnergyConsumption WaterConsumption HazardousWaste	Pset_InstallationOccurrence InstallationDate AcceptanceDate PutIntoOperationDate	Pset_MaintenanceStrategy AssetCriticality AssetFrailty AssetPriority
Pset_MaintenanceTriggerCondition ConditionTargetPerformance ConditionMaintenanceLevel ConditionReplacementLevel	Pset_MaintenanceTriggerDuration DurationTargetPerformance DurationMaintenanceLevel DurationReplacementLevel	Pset_MaintenanceTriggerPerformance TargetPerformance PerformanceMaintenanceLevel ReplacementLevel
Pset_ManufacturerOccurrence AcquisitionDate BarCode SerialNumber	Pset_ManufacturerTypeInformation GlobalTradeItemNumber ArticleNumber ModelReference	Pset_RepairOccurrence RepairContent RepairDate MeanTimeToRepair
Pset_Risk RiskName RiskType NatureOfRisk	Pset_ServiceLife ServiceLifeDuration MeanTimeBetweenFailure	Pset_Tiling Permeability TileLength TileWidth
Pset_Tolerance ToleranceDescription ToleranceBasis DiurnalTolerance	Pset_Uncertainty UncertaintyBasis UncertaintyDescription HorizontalUncertainty	Pset_Warranty WarrantyIdentifier WarrantyStartDate IsExtendedWarranty

Figur: Egenskapsuppsättningar som får användas för att beskriva *IfcCovering* (täckande objekt).

Generella egenskapsuppsättningar

Utöver dessa objektspecifika finns generella egenskapsuppsättningar som innehåller egenskaper som gäller övergripande för flera typer av objekt, till exempel för indikatorer för miljöpåverkan (*Pset_EnvironmentalImpactIndicators*), riskhantering (*Pset_Risk*), och garantier (*Pset_Warranty*).

Användardefinierade egenskapsuppsättningar (*User-defined Property Sets*)

Det är även möjligt att skapa användardefinierade egenskapsuppsättningar (*User Defined Property Sets*). Dessa får benämnas valfritt enligt IFC, så länge de inte inleds med prefixet *Pset_*. Detta för att särskilja dem från de egenskapsuppsättningar som ingår i IFC-standarderna. Namngivning av egenskapsuppsättningar bör ändå följa en namnkonvention med en viss struktur, vilket beskrivs längre ner i detta avsnitt. Detta för att hålla en viss praxis i branschen och ge en förutsättning för förståelse för egenskapsuppsättningens ursprung och innehåll.

Egenskaper

Egenskaper inom IFC-standarderna

Varje egenskapsuppsättning består av en eller flera egenskaper. Varje egenskap kan förekomma i en eller flera egenskapsuppsättningar. Egenskapen Status ingår exempelvis i alla egenskapsuppsättningar *Pset_[Entitet]Common*.

Varje egenskap definieras med en

- benämning (*Name*),
- egenskapstyp (*Property Type*),
- datatyp (*Data Type*)
- beskrivning (*Description*).

Se exempel för egenskapen Status nedan.

Tabell 8: Beskrivning av egenskaper.

Name	Property Type	Data Type	Description
Status	IfcPropertyEnumeratedValue	PEnum_ElementStatus	Status of the element, predominately used in renovation or retrofitting projects. The status can be assigned to as "New" - element designed as new addition, "Existing" - element exists and remains, "Demolish" - element existed but is to be demolished, "Temporary" - element will exists only temporary (like a temporary support structure).

Benämning (*Name*)

Benämning av egenskaper anges alltid med så kallad Pascalnotation, det vill säga att orden skrivs ihop med versal (stor) första bokstav, exempelvis **AcousticRating**, **IsExternal** och **FireRating**.

Egenskapstyp (*Property Type*)

Egenskapstyp anger vilken typ av egenskapsvärde som får anges för aktuell egenskap, exempelvis om värdet ska vara ett enskilt värde, ett värde med en undre och övregräns, eller enligt en värdelista. Sex typer finns definierade i IFC-standarderna:

- IfcPropertySingleValue
- IfcPropertyBoundedValue
- IfcPropertyEnumeratedValue
- IfcPropertyListValue
- IfcPropertyReferenceValue
- IfcPropertyTableValue

Nedan beskrivs egenskapstyperna i detalj:

IfcPropertySingleValue är en av de mest grundläggande och ofta använda egenskapstyperna inom IFC. Den används för att definiera en egenskap som har ett enda, specifikt värde kopplat till ett objekt. Denna typ är enkel men mycket flexibel och används i många olika sammanhang.

Definition	Komponenter	Användning	Exempel
Används för att representera en egenskap som har ett enskilt värde. Detta kan vara ett numeriskt värde, en textsträng, en boolesk (sant/falskt) värde,	NominalValue: Det faktiska värdet på egenskapen. Detta kan vara ett tal (heltal, decimaltal), en text, en boolesk variabel, etc. Unit (valfritt): Enheten för värdet, om tillämpligt (t.ex., meter, kilogram,	Används när en egenskap har ett klart definierat, enda värde. Det är den vanligaste typen av egenskaper och används för allt från dimensioner till prestandavärden och materialbeskrivningar. Den används även för att specificera konstanta värden som inte	Bredden på ett fönster, t.ex. 1200 millimeter. Brandklassen på en vägg, t.ex. EI30. Temperaturen i ett rum, t.ex. 22 grader Celsius. Definition

Definition	Komponenter	Användning	Exempel
eller någon annan enkel datatyp.	sekund). Om enheten inte anges, antas en standardenhet beroende på egenskapen.	varierar över tid eller under olika förhållanden.	om en dörr är låsbar eller ej, True/False.

IfcPropertyBoundedValue används för att definiera en egenskap med ett värdeintervall, alltså ett minimum och ett maximum. Denna typ är användbar när man behöver ange tillåtna gränser för en viss egenskap, vilket är viktigt för att representera toleranser eller begränsade värden som ett objekt kan anta.

Definition	Komponenter	Användning	Exempel
Används för att representera egenskaper med värden som ligger inom ett visst intervall. Den definierar alltså både lägsta och högsta tillåtna värde för en specifik egenskap.	UpperBoundValue: Det högsta tillåtna värdet för egenskapen. LowerBoundValue: Det lägsta tillåtna värdet för egenskapen. SetPointValue (valfritt): Ett rekommenderat eller typiskt värde inom det angivna intervallet, men inte ett krav. Unit: Enheten för de värden som anges (t.ex. meter, grader Celsius).	Vanlig i scenarier där värden inte är fasta utan har en viss grad av variation, till exempel dimensionstoleranser, acceptabla temperaturintervall, eller styrkan hos ett material inom specificerade gränser. Används ofta för parametrar i byggnadsfysik, som t.ex. intervall för luftfuktighet eller temperatur som ett rum ska hålla, eller för att specificera tillåtna variationer i materialegenskaper.	Ett fönster kan ha ett U-värde som måste ligga mellan 0,8 och 1,2 W/m ² K, där dessa gränser definieras med <i>IfcPropertyBoundedValue</i> . En vägg kan specificera tillåtna tjockleksvariationer, t.ex. att tjockleken får vara 100–120 mm.

IfcPropertyEnumeratedValue används för att definiera en egenskap vars värde väljs från en fördefinierad lista av alternativ. Denna typ är särskilt användbar när värdet på en egenskap behöver begränsas till specifika, accepterade alternativ, vilket säkerställer konsistens och standardisering av data.

Definition	Komponenter	Användning	Exempel
Används för att representera egenskaper vars värde hämtas från en lista med fördefinierade alternativ, d.v.s. en värdelista. Detta innebär att användaren måste välja värdet från denna lista, vilket förhindrar inmatning av felaktiga eller ogiltiga värden.	EnumerationValues: En lista över ett eller flera valda värden från värdelistan. EnumerationReference: En referens till värdelistan som definierar de tillåtna värdena. Detta är ofta en separat definierad lista som specificerar alla möjliga alternativ. Unit (valfritt): Enheten för värdet om det är en numerisk typ, vilket inte alltid är fallet för värdelistor.	Används när en egenskap ska ha ett värde som strikt ska väljas från en uppsättning fördefinierade alternativ. Denna typ av egenskap är viktig för att säkerställa att data följer specifika standarder och är konsekventa. Den används ofta för val som relaterar till specifikationer eller standarder, såsom materialtyper,	Materialtyp: För ett väggobjekt kan en värdelista ange att materialet måste väljas från alternativen "Betong", "Stål", "Trä". Färg: För ett rum kan en värdelista definiera färgen som en av "Blå", "Grön", "Röd". Status: För en byggkomponent kan status definieras som en av "Nytt", "Befintligt", "Rives". Ytbehandling: En dörr kan ha en ytbehandling definierad som "Målad",

Definition	Komponenter	Användning	Exempel
		färgkoder, och klassificeringar.	"Laminerad", eller "Natur".

IfcPropertyListValue används för att definiera en egenskap som kan ha flera värden i form av en lista. Denna typ är särskilt användbar när en egenskap inte begränsas till ett enda värde utan kan anta flera värden samtidigt, vilket ger flexibilitet för att representera komplexa datauppsättningar.

Definition	Komponenter	Användning	Exempel
Används för att representera egenskaper som har en serie av värden, snarare än ett enda. Denna egenskapstyp tillåter att flera relaterade värden i en organiserad lista anges, vilket kan vara numeriska, textbaserade, booleska eller av andra datatyper.	ListValues: En sekvens av värden som anges som en lista. Listan kan innehålla flera värden av samma typ (t.ex. flera tal eller strängar). Unit (valfritt): Enheten för de angivna värdena om de är av numerisk typ. Om enheten inte specificeras används standarden för den specifika egenskapen.	Används när man behöver representera flera värden för en och samma egenskap, vilket är vanligt när data ska uttrycka en serie av liknande mätningar, tillstånd eller attribut. Används även för att representera samlingar av data som naturligt passar i en lista, såsom mätvärden över tid eller flera möjliga konfigurationer.	Dimensioner: Olika tillgängliga glasdimensioner för ett fönster, t.ex. [0,5 m, 1,0 m, 1,5 m]. Temperaturvärden: För ett klimatsystem kan en lista specificera olika driftstemperaturer, som [18°C, 22°C, 26°C]. Tillgängliga kulörer: För ett väggmaterial kan man ange flera alternativ som ["Blå", "Grön", "Vit"]. Materialtjocklek: En skiva kan ha flera tillåtna tjocklekar definierade som [10 mm, 12 mm, 15 mm].

IfcPropertyReferenceValue används för att definiera en egenskap vars värde är en referens till en annan entitet eller objekt i IFC-modellen. Istället för att direkt ange ett värde, pekar **IfcPropertyReferenceValue** på ett externt objekt, vilket möjliggör mer komplexa relationer och informationskopplingar mellan olika delar av en modell.

Definition	Komponenter	Användning	Exempel
Används för att representera en egenskap vars värde är en referens till en annan entitet, som exempelvis ett material, en klassificering, en extern fil eller någon annan IFC-objekttyp. Denna typ av egenskap skapar en länk mellan objektet som har egenskapen och den refererade entiteten.	PropertyReference: En referens till det objekt eller den entitet som egenskapen pekar på. Detta kan vara vilket IFC-objekt som helst, såsom IfcMaterial , IfcDocumentReference , IfcClassificationReference . UsageName (valfritt): Ett beskrivande namn som anger hur den refererade egenskapen används i sitt sammanhang.	Används när det är mer lämpligt att referera till en annan datakälla eller objekt snarare än att duplicera information. Detta är särskilt användbart för att undvika redundans och för att säkerställa att information hålls uppdaterad och konsekvent i modellen. Den används också för att skapa relationer mellan objekten i en modell, vilket gör det möjligt att spåra och strukturera data på ett	Materialreferens: En vägg kan ha en referens som pekar på ett IfcMaterial , vilket specificerar det exakta materialet som används. Klassificering: Ett byggobjekt kan ha en referens till en IfcClassificationReference som anger vilken klassificering objektet har enligt en standard som t.ex. CoClass . Dokumentlänkning: En byggdel kan referera till en IfcDocumentReference som pekar på en teknisk ritning, produktmanual eller certifikat. Externa data: Ett VVS-system kan referera till en extern datakälla som beskriver dess prestandaparametrar.

Definition	Komponenter	Användning	Exempel
		mer sammanhängande sätt.	

IfcPropertyTableValue används för att definiera en egenskap där värdet representeras som ett par av relaterade värdelistor, vanligtvis i form av en tabell. Denna typ är särskilt användbar när det finns behov av att uttrycka en relation mellan två uppsättningar data, till exempel vid parametriska värden eller funktioner som beskriver en variabel beroende på en annan.

Definition	Komponenter	Användning	Exempel
Används för att representera en egenskap som har relaterade värden organiserade i tabellform, med en lista av ingångsvärden (input values) och motsvarande utgångsvärden (output values). Detta gör det möjligt att beskriva komplexa samband mellan olika parametrar.	<p>DefiningValues: En lista över definierande värden (ofta x-axeln i en funktion eller tabell), exempelvis temperaturer, tidpunkter eller specifika belastningsfall.</p> <p>DefinedValues: En lista över definierade värden som motsvarar de definierande värdena (ofta y-axeln), exempelvis tryck, hastigheter eller krafter.</p> <p>Expression (valfritt): En text som beskriver sambandet eller regeln mellan de definierande och definierade värdena.</p> <p>DefiningUnit och DefinedUnit (valfritt): Enheterna för de definierande respektive definierade värdena, om de är av numerisk typ.</p>	Används för att uttrycka data där en egenskap förändras beroende på en annan, såsom resultat av beräkningar, empiriska mätningar, eller andra samband. Den är särskilt användbar i tekniska och prestandabaserade sammanhang där kurvor eller samband måste beskrivas explicit.	Tryck-temperatursamband: Egenskapen kan beskriva hur trycket i en tank varierar beroende på temperaturen, med temperaturen som definierande värde och trycket som definierat värde. Flödes hastighet över tid: Ett rörsystem kan ha en tabell som visar flödes hastighet vid olika tidpunkter. Effekt mot hastighet: En fläkt kan ha en tabell som beskriver effekten (watt) vid olika rotationshastigheter (RPM). Materialegenskaper: En materialtyp kan beskriva olika mekaniska egenskaper (som elasticitetsmodul) vid olika belastningsnivåer.

Datotyp (Data Type)

Datotyp definierar vilka typer av data som kan användas för egenskaper samt specificerar de grundläggande formaten för information, vilket är avgörande för att säkerställa korrekthet, interoperabilitet och användbarhet av data mellan olika system och applikationer. Dessa grupperas enligt tabellen nedan:

Tabell: Gruppering av egenskapstyper.

Gruppering av egenskapstyp

<i>IfcSimpleValue</i>	Består av 15 allmänna datatyper. Nedan beskrivs de mest vanligt förekommande. • <i>IfcBoolean</i> – Logiskt värde som antingen kan vara sann (TRUE) eller falsk (FALSE) • <i>IfcDate</i> – Datum med formen YYYY-MM-DD • <i>IfcIdentifier</i> – Maskinläsbar alfanumerisk sträng (STRING med max 255 tecken) för identifikation • <i>IfcInteger</i> – Heltal (INTEGER) • <i>IfcLabel</i> – Mänskligt läsbar sträng (STRING med max 255 tecken) för identifikation •
-----------------------	---

Gruppering av egenskapstyp

	<i>IfcReal</i> – Reellt tal (heltal, decimaltal och bråkta)l) • <i>IfcText</i> – Mänskligt läsbar alfanumerisk text (STRING) enbart avsett för informationsnyfte
<i>IfcMeasureValue</i>	Består av de vanligaste datatyperna som definieras inom ISO 10303-41. Dessa är exempelvis längd, area, volym, och antal
<i>IfcDerivedMeasureValue</i>	Består av specifika datatyper som är beräknade, exempelvis värmemotstånd, flöden, och tryck

Vissa egenskaper beskrivs även med en enhet (*IfcUnit*). Enhet ska aldrig anges som en del av egenskapsvärdet. Egenskapsvärdet för luftljudsisolering (*AcousticRating*) ska exempelvis anges enbart med värdet (exempelvis 40), inte inklusive enhet (exempelvis 40 dB).

Användardefinierade egenskaper (*User-defined Properties*)

Det går även att skapa användardefinierade egenskaper. Dessa får benämnas valfritt enligt IFC och placeras i en användardefinierad egenskapsuppsättning. Namngivning av egenskaper bör ändå följa en namnkonvention med en viss struktur, vilket beskrivs längre ner i detta avsnitt. Detta för att hålla en viss praxis i branschen och ge en förutsättning för förståelse för egenskapens härkomst och innehåll. Vid skapande av användardefinierade egenskaper är det viktigt att definiera dem med en Benämning (*Name*), Egenskapstyp (*Property Type*), och Datatyp (*Data Type*) som beskrivits ovan.

Principer för val av egenskaper och egenskapsuppsättningar

Processen för att kravställa, producera, validera och leverera information via IFC-formatet beskrivs i avsnittet "Arbetsflöde openBIM". I denna process ingår att ställa krav på informationsutbyte med hjälp av informationsutbyteskrav (EIR) enligt ISO 19650 med IFC. Som nämnts ovan innehåller dessa krav på egenskaper och egenskapsuppsättningar vilka även har beskrivits ovan. I detta avsnitt beskrivs principer och rekommendationer för val av egenskaper och egenskapsuppsättningar.

Som beskrivits ovan består IFC-standarden av en mängd egenskaper och egenskapsuppsättningar. Förutom dessa finns det även en uppsjö av andra egenskaper, exempelvis inom olika typer av standarder, så som ETIM och CoClass, liksom egenskaper som skapats genom olika branschinitiativ, så som BIP och BEAst. Vidare skapas det företags- och projektspecifika egenskaper .

Detta medför att det finns många egenskaper som benämns olika men som syftar på samma typ av information. Detta medför onödig hantering och gör informationsflödet mindre effektivt. För att minimera den risken rekommenderas följande ordning vid val av egenskaper:

1. Enligt IFC-standarden

I första hand bör förbestämda egenskaper och egenskapsuppsättningar från IFC-standarden användas för att beskriva valda entiteter. Dessa egenskaper är redan definierade i form av egenskaps- och datatyper samt med en beskrivning. Uppdragsgivare behöver dock fortfarande beskriva vilka av dessa egenskaper som ska levereras vilket görs med informationsutbyteskrav (EIR) med IFC.

2. Enligt CoClass

I andra hand bör egenskaper väljas från egenskapstabellen inom CoClass. CoClass saknar dock egenskapsuppsättningar vilket då måste definieras. Detta beskrivs längre ner i detta avsnitt.

3. Användardefinierad

I sista hand kan användardefinierade egenskaper och egenskapsuppsättningar nyttjas, dock enbart om nödvändiga egenskaper inte kan erhållas från varken IFC-standarden eller CoClass. Dessa egenskaper kan exempelvis härstamma från ETIM, BIP, BEAst, ett specifikt företag eller projekt.

Vid dessa fall måste både egenskaper och egenskapsuppsättningar definieras vilket beskrivs längre ner i detta avsnitt.

Enligt IFC-standarden

Så fort informationen i en IFC-modell ska läsas av en mottagande programvara bör egenskaper och egenskapsuppsättningar följa IFC-standarden. På så vis underlättas samverkan mellan programvarorna eftersom programmering kan ske enligt IFC-formatet och programvarorna kan certifieras enligt IFC-standarden.

Motsatt kan det upplevas svårt och ineffektivt att hitta kravställda egenskaper för användare när de ligger utspridda i flera olika egenskapsuppsättningar. Detta kan lösas på två följande sätt:

1. Genom att välja en programvara för att titta på informationen där det finns möjlighet att välja att inte gruppera egenskaperna per egenskapsgrupp. Denna funktion varierar bland dagens programvaror och i vissa fall kanske det inte finns möjlighet till detta val.
2. Genom att ställa krav på en extra, användardefinierad, egenskapsuppsättning som redovisar alla kravställda egenskaper i en och samma egenskapsuppsättning. Det kommer dock innebära att alla egenskaper redovisas dubbelt, det vill säga både i egenskapsuppsättningarna enligt IFC och i en användardefinierad egenskapsuppsättning.

Vid detta val kommer alla benämningar och beskrivningar av egenskaper och egenskapsuppsättningar att vara på engelska. Detta leder lättare till missförstånd av dess betydelse vilket ökar risken för felkällor. Därför bör informationsutbyteskraven även innehålla egenskapsdefinitioner som tydligt beskriver egenskaperna på svenska. Egenskaper på engelska kan även vara en fördel om det finns utländska projektdeltagare. Detta kommer troligtvis att lösa sig med tiden via *buildingSMART Data Dictionary* (bSDD) men i dagsläget finns inte denna på svenska. När denna är mer utvecklad kommer troligtvis språket inte att vara något problem då de flesta språk kommer att finnas representerade i databasen.

Enligt CoClass

CoClass egenskaper delas in i kulturella respektive materiella egenskaper. Dessa bryts även ner i mindre grupper i ytterligare två nivåer innan egenskaperna redovisas. Varje egenskap beskrivs med en klasskod, benämning, etikett, och definition, se figuren nedan.

The screenshot shows a software interface with a sidebar on the left and a main content area on the right. The sidebar has a search bar and a list of properties: 'NESZ Storlekskod', 'NEMF Tillverkare' (highlighted in blue), and 'NEMN Tillverkare nummer'. The main content area displays the details for 'NEMF Tillverkare'. It includes a status 'Fastställd', a description 'Beskrivning' and 'Klasser', an 'Etikett: ccNameManufacturer', and a 'Definition: benämning som anger tillverkare'.

- **Klasskod** (exempelvis *NEMF* i figuren ovan) redovisar egenskapens unika identifikation och kan nyttjas för maskinläsning.
- **Benämning** (exempelvis *Tillverkare* i figuren ovan) redovisar egenskapens namn på svenska och engelska.
- **Etikett** (exempelvis *ccNameManufacturer* i figuren ovan) redovisar egenskapens unika etikett och kan nyttjas inom maskinläsning.

- **Definition** (exempelvis ”benämning som anger tillverkare” i figuren ovan) beskriver egenskapen kortfattat på svenska och engelska.

Benämning av egenskaper enligt CoClass

Vid val att använda CoClass egenskaper ska de benämnas med dess etikett, det vill säga med en engelsk benämning med Pascalnotation inklusive prefixet cc, exempelvis *ccNameManufacturer* enligt figuren ovan. Det går dock att i egenskapsdefinitionerna ange både egenskapens svenska och engelska benämning eftersom det finns definierat inom CoClass.

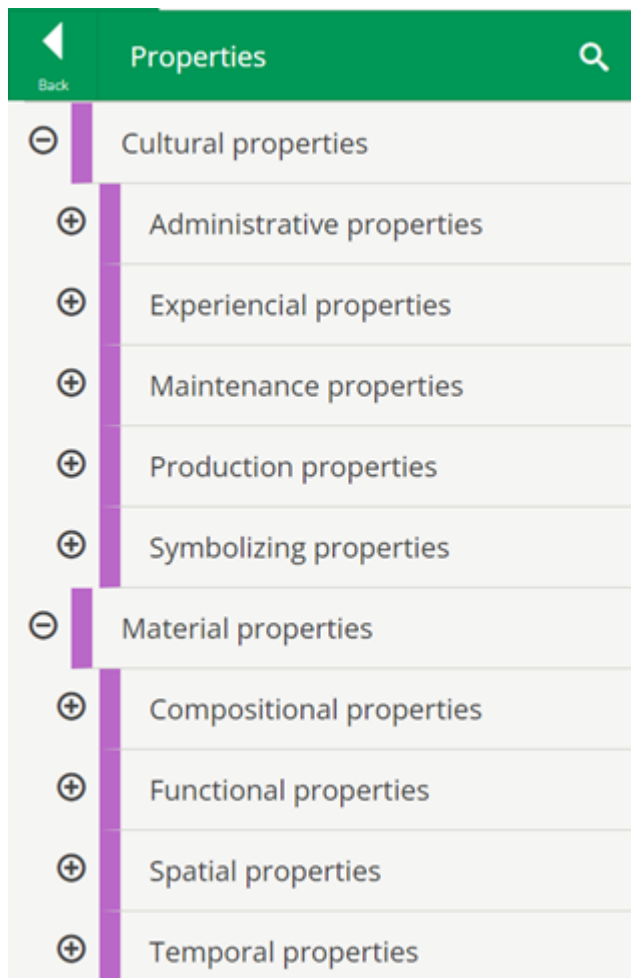
Benämning av egenskapsuppsättningar enligt CoClass

Vid nyttjande av egenskaper enligt CoClass ska dessa egenskaper placeras i en egenskapsuppsättning som anger att ingående egenskaper ingår i CoClass. Där för ska benämningen av egenskapsuppsättningen följa nedan namnkonvention:

[Landskod]-CoClass_[Benämning]

- [Landskod] – Tvåställig kod enligt ISO 3166-1 och -2. Gällande kod är SE eftersom CoClass är ett svenskt klassifikationssystem.
- CoClass – Identifikation av egenskapsuppsättningens ursprung, vilket i detta fall är CoClass.
- [Benämning] – Aktuell rubrik på engelska enligt CoClass egenskapstabell. Denna del är valfri.

Det rekommenderas att placera alla egenskaper i en och samma egenskapsuppsättning men det är naturligtvis möjligt att dela in dessa i flera egenskapsuppsättningar. Vid indelning i flera egenskapsuppsättningar rekommenderas att använda rubrikerna på den andra nivån i egenskapstabellen, se figuren nedan.



Nedan redovisas några exempel på benämningar:

- *SE-CoClass*
- *SE-CoClass_AdministrativeProperties*
- *SE-CoClass_SpatialProperties*

Användardefinierade

Användardefinierade egenskaper bör enbart nyttjas när önskad egenskap varken återfinns inom IFC-standardens eller CoClass, och vidare om önskad information inte kan placeras på något av IFC-standardens attribut (se avsnitt Attribut och egenskaper).

Benämning av användardefinierade egenskaper

Benämning av användardefinierade egenskaper bör följa samma systematik som IFC-standardens och alltså benämnas med Pascalnotation. Egenskaper kan benämnas på svenska eller engelska, men en informationsleverans bör enbart bestå av egenskaper på ett av språken så att inte svenska och engelska egenskaper blandas i en leverans.

Benämning av användardefinierade egenskapsuppsättningar

Benämning av användardefinierade egenskapsuppsättningar bör följa nedan namnkonvention.

[LANDSKOD]-[URSPRUNG]-[PROJEKT]_[Benämning]

- [LANDSKOD] – Tvåställig kod enligt ISO 3166-1 och -2. (SE för Sverige)
- [URSPRUNG] – Identifiering av ursprunget till egenskapsuppsättningen. Detta anges lämpligtvis med en förkortning i versaler, exempelvis *TRV* för Trafikverket, *VGR* för Västra Götalandsregionen, *BIP* för bpkoder.se, eller *BEAst* för Byggbranschens elektroniska affärsstandard. Ursprunget kan även skrivas ut utan förkortning men med Pascalnotation, exempelvis *OrebroBostader*.
- [PROJEKT] – Denna del är valfri och kan användas för att redovisa projektspecifika egenskaper, det vill säga sådana egenskaper som inte är fastställda av ägaren till egenskapsuppsättningen. Detta kan vara ett projektnummer (exempelvis 11537), en förkortning av projektets namn (exempelvis *SGL* för Sahlgrenska Life), eller enbart ett valfritt tecken för att markera att egenskapsuppsättningen avviker från ägarens fastställda egenskaper.
- [Benämning] – Denna del är valfri och kan definiera egenskapsuppsättningen ytterligare. Benämning anges med Pascalnotation.

Nedan redovisas några exempel på benämning av egendefinierade egenskapsuppsättningar:

- *SE-BIP*
- *SE-OrebroBostader*
- *SE-TRV-X*
- *SE-VGR-11573_Utrymmen*

Det rekommenderas att undvika å, ä, ö och specialtecken (? # % & m.fl.) eftersom det finns system som kan ha problem med att läsa dessa tecken.

Undvik överinformation

Oavsett val ovan är det viktigt att enbart kravställd information levereras i IFC-modellen. Återfinns mer egenskaper än de som är kravställda blir det svårt att lita på informationen i modellen. Det blir svårt att veta vilka egenskaper som är kravställda och validerade jämfört med egenskaper som levererats utan att ha kvalitetskontrollerats.

Hantera egenskapers livscykel

Liksom att byggdelar har en livscykel har även egenskaper en livscykel, exempelvis ”som specificerat”, ”som efterfrågat”, ”som offererat” m m, men IFC 4.3 hanterar inte livscykelinformation för egenskaper. Om man vill hantera de olika stegen i livscykeln rekommenderas att ett suffix används. Suffixet tydliggör egenskapens livscykelsteg. För en kravställd egenskap kan exempelvis suffixet *_SPE* läggas till (*SPE* är en förkortning av den engelska termen ”As specified” enligt standarden IEC 62569-1:2017).).

Detta kan exempelvis nyttjas för att ange akustiska och brandtekniska krav på byggdelar. En vägg med ett akustiskt krav kan beskrivas av en akustiker med egenskapen *ccSoundInsulation_SPE* som ska placeras i egenskapsuppsättningen *SE-CoClass*. Vid projektering av samma vägg anger arkitekten **det projekterade värdet i egenskapen *ccSoundInsulation*** i samma egenskapsuppsättning. Det uppmätta värdet under förvaltningen kan ges egenskapen ***ccSoundInsulation_OP*** (där *OP* står för Operated).

I Praxi beskrivs detta i metoden Kvalificerare för Egenskaper.

□ TÄNK PÅ

Rekommendationen är att i första hand, och så långt som möjligt, använda de i IFC skapade egenskapsuppsättningarna.

Metadata

Namespace: swe-nrb

Paket: openbim

Version: 1.0.0-rc.1

Publiceringsdatum: 2025-06-10

Sökväg: 4_tillampning_ifc/egenskaper.partial.html

Genererad:



QR koden innehåller en länk tillbaka till underlagsfilen