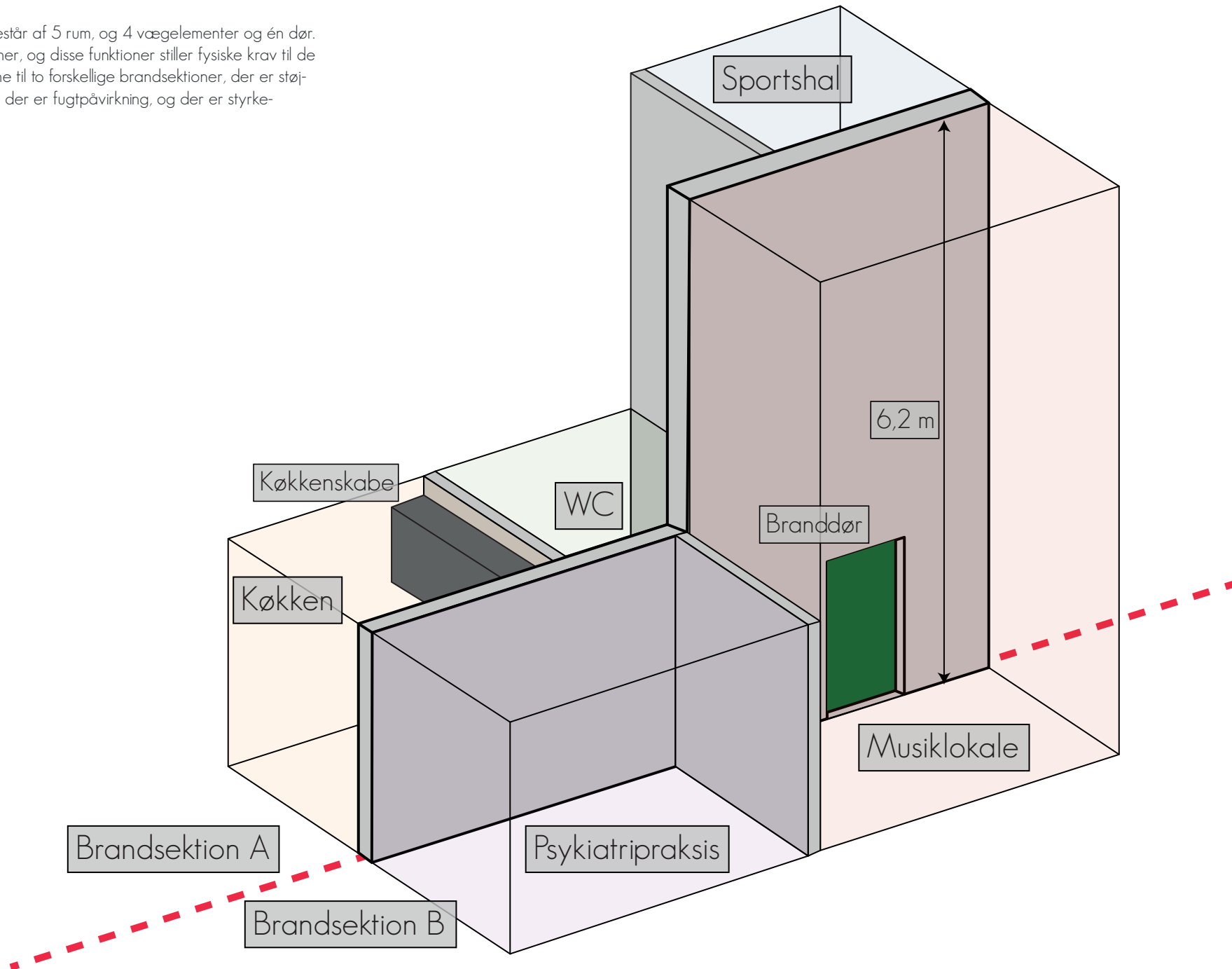


Testmodellen

Testmodellen er et udsnit af en bygning som består af 5 rum, og 4 vægelementer og én dør. Rummene danner ramme for forskellige funktioner, og disse funktioner stiller fysiske krav til de tilstødende vægge. Eksempelvis hører rummene til to forskellige brandsektioner, der er støj-mæssige foranstaltninger som skal overholdes, der er fugtpåvirkning, og der er styrke-mæssige krav.

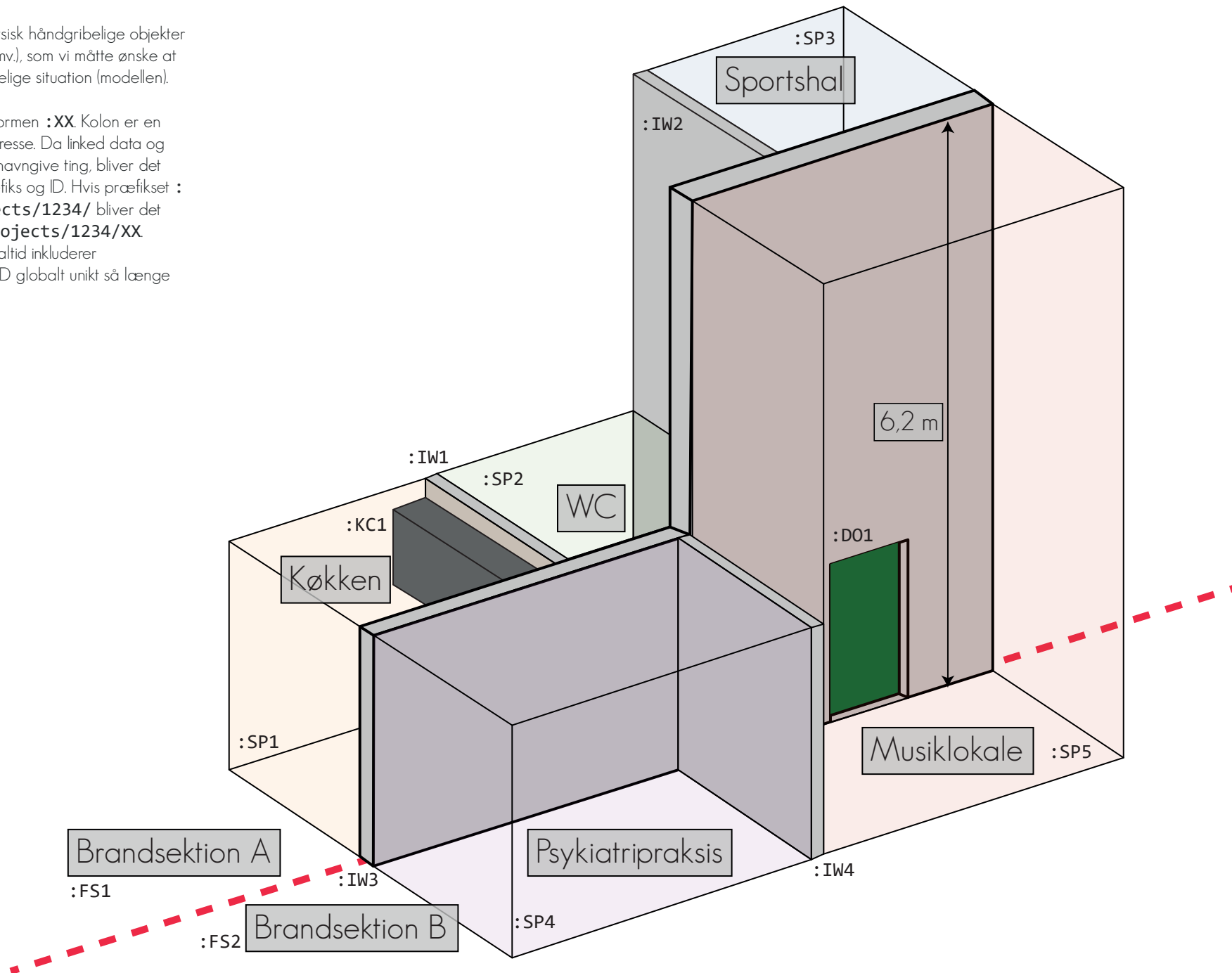


Objekter

I udgangspunktet er der her tale om alle de fysisk håndgribelige objekter (vægge, døre, dæk) og ikke-fysiske ting (rum mv.), som vi måtte ønske at knytte viden til i vores virtualisering af den virkelige situation (modellen).

Objekterne er her identificeret med ID'er på formen **:XX**. Kolon er en forkortelse for et præfiks i form af en URL-adresse. Da linked data og RDF forudsætter brug af webadresser til at navngive ting, bliver det egentlige ID, som er en sammenføjning af præfiks og ID. Hvis præfikset **:** defineres som **https://niras.dk/projects/1234/** bliver det egentlige ID altså **https://niras.dk/projects/1234/XX**. Hvis NIRAS internt bruger en URL-struktur der altid inkluderer projektnummeret (her **1234**), så bliver ethvert ID globalt unikt så længe det blot er unikt i projektet.

Hvis objekter kommer fra en IFC-fil er det typisk praksis at benytte GlobalId.
Ex: **:ØBTBfw6f90Nfh9rP1d1Xr2**.



Klassifikation

I RDF klassificeres objekter ved at knytte et **rdf:type**-prædikat på objektet. Et objekt kan tilhøre flere klasser, så en bil kan eksempelvis både være et køretøj, et transportmiddel og en potensforlænger.

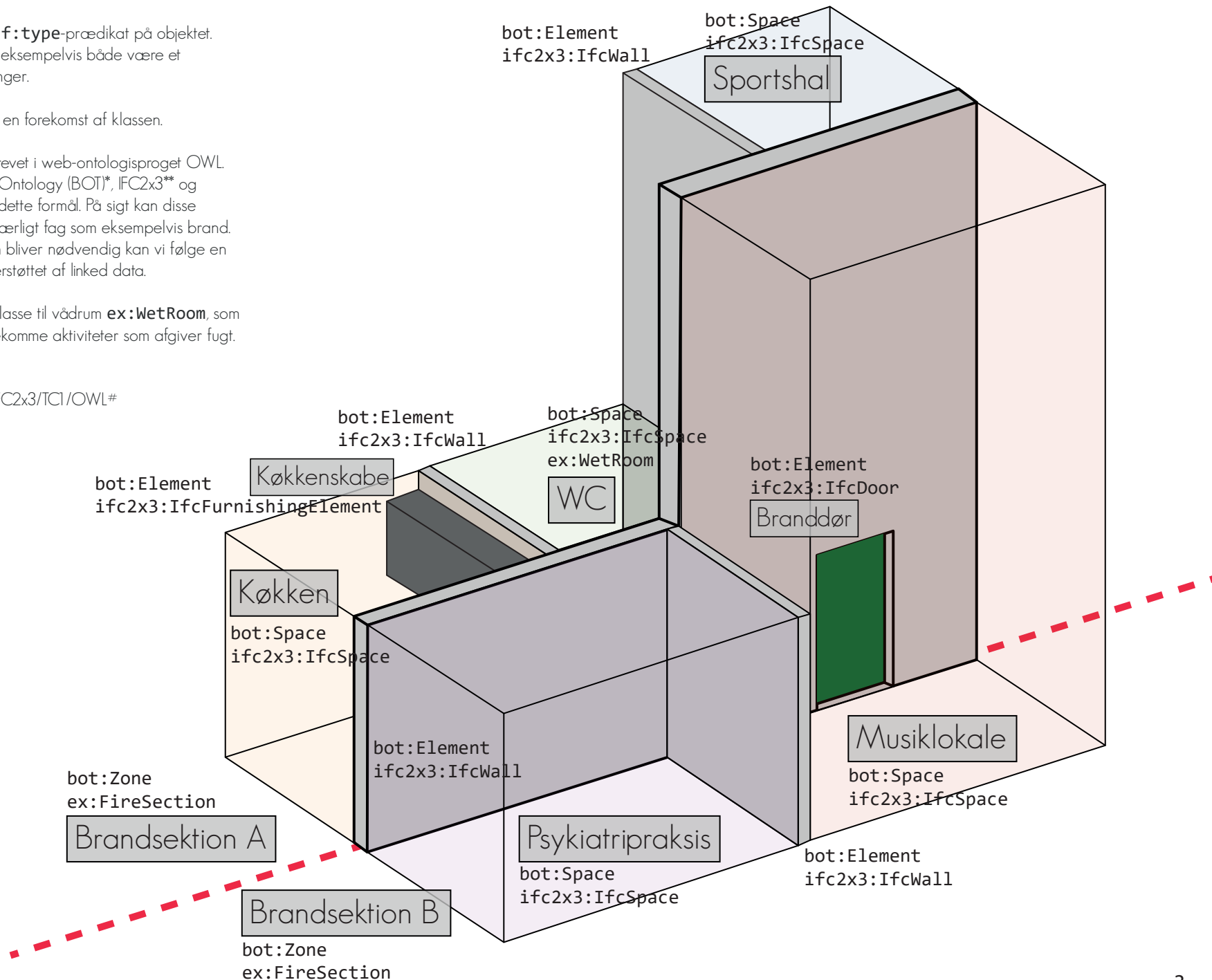
Man siger også at objektet er en instans, eller en forekomst af klassen.

Klasserne defineres i såkaldte ontologier beskrevet i web-ontologisproget OWL. Vi klassificerer her med the Building Topology Ontology (BOT)*, IFC2x3** og vores egne klasser som vi opretter specifikt til dette formål. På sigt kan disse trækkes ud og publiceres i en ontologi for et særligt fag som eksempelvis brand. Da terminologien opstår efterhånden som den bliver nødvendig kan vi følge en bottom up approach i standardiseringen, understøttet af linked data.

Lad os i denne sammenhæng introducere en klasse til vådrum **ex:WetRoom**, som betegner et rum hvor der må forventes at forekomme aktiviteter som afgiver fugt.

*<https://w3id.org/bot#>

**<https://standards.buildingsmart.org/IFC/DEV/IFC2x3/TC1/OWL#>



Simple egenskaber

Det er muligt at knytte simple egenskaber til objekter.

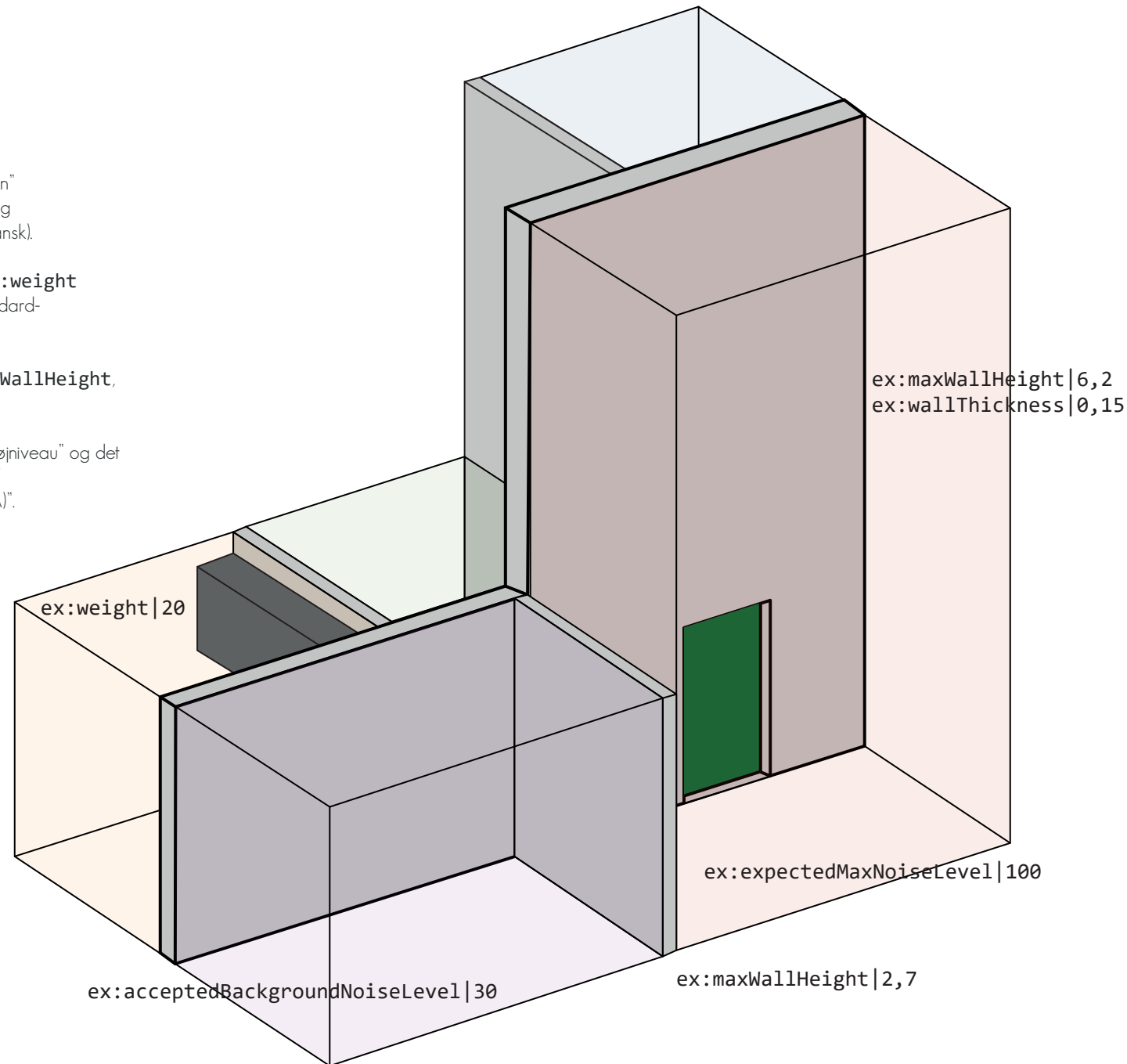
Eksempler på simple såsom "Skab har vægt 50 kg", men de kan også være komplekse som "Skab har denne geometri i IFC" og "Skab har denne geometri i glTF".

Det er muligt at beskrive datatypen som en egenskab såsom "string", "boolean" eller "decimal", og det er endvidere muligt at definere sproget på en tekststreng med dens ISO 639-1 sprogkode som eksempelvis "en" (engelsk) eller "da" (dansk).

For at kunne beskrive krav til væggens styrke, introducerer vi egenskaben **ex:weight** som knyttes til det væghængte skab. For denne egenskab definerer vi at standard-enheden er "kg".

Af fysiske mål på en væg introducerer vi **ex:wallThickness** og **ex:maxWallHeight**, som begge angives i enheden "m".

For at kunne beskrive støjforhold introducerer vi det "forventede maksimale støjniveau" og det "accepterede niveau for baggrundsstøj" (**ex:expectedMaxNoiseLevel** / **ex:acceptedBackgroundNoiseLevel**) som begge har enheden "dB(A)".



Grænseflader

Objekter kan som beskrevet være ikke-fysiske koncepter som rum. I BOT beskrives konceptet **bot:Interface** som et generisk koncept til at kvalificere relationen mellem to eller flere objekter. I testmodellen findes der vægge som grænser op til mere end ét rum, og derfor er det ideelt at bruge dette koncept til at beskrive vægoverfladerne. Dette er også ideelt idet en væg typisk består af et rammeværk som kan være ens for hele væggen, imens overflader tilpasses til rummets brug (ex vådrum).

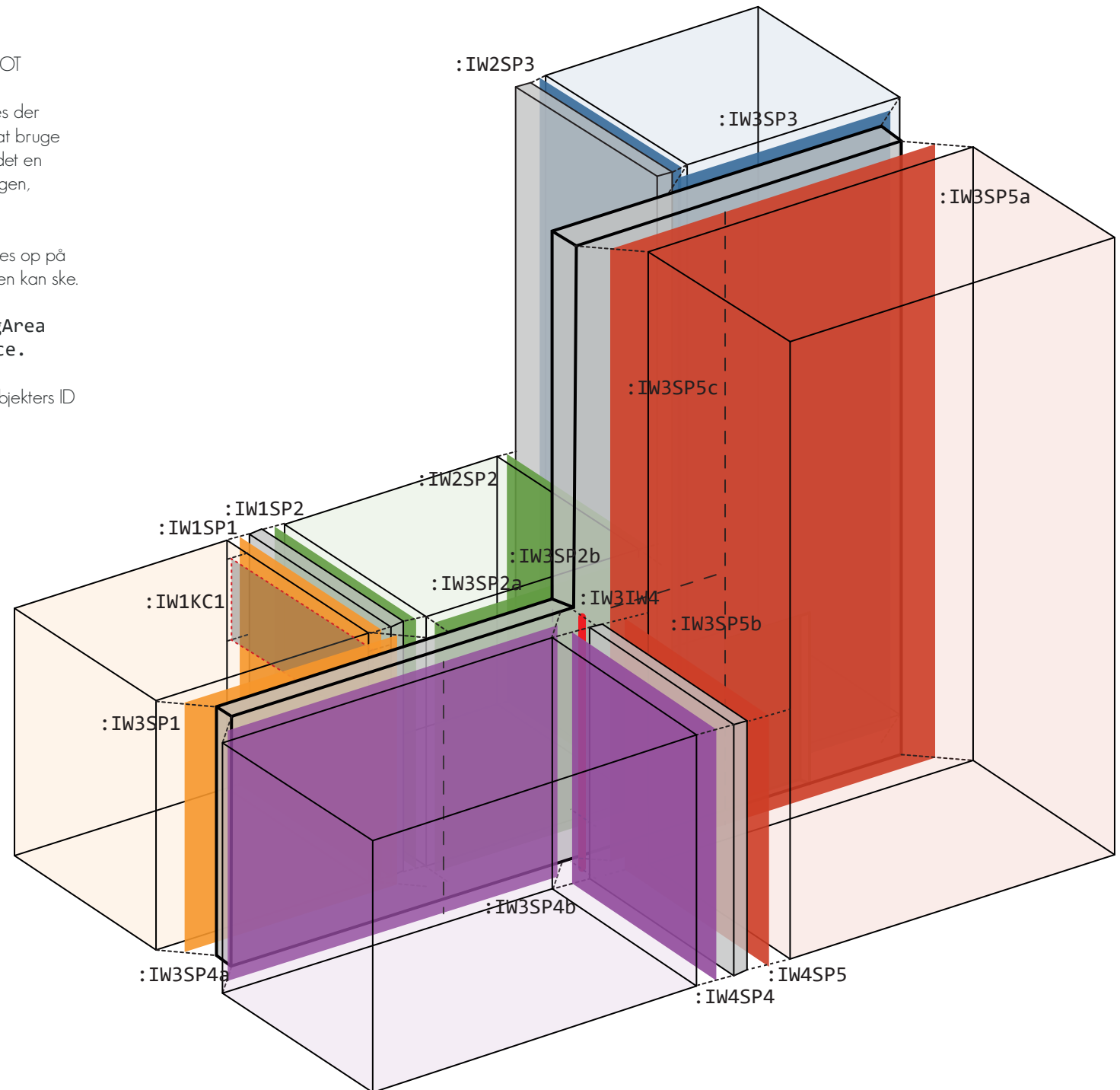
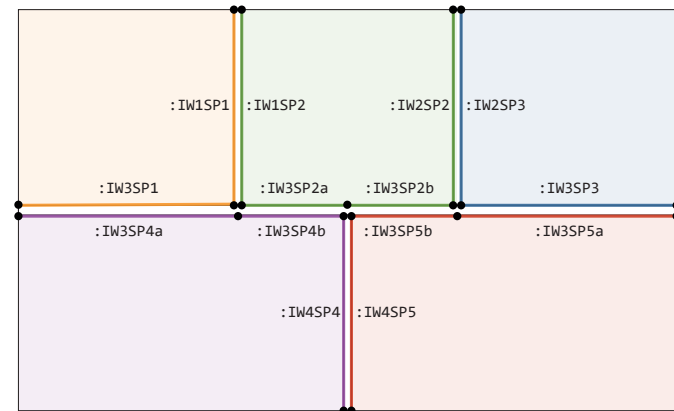
Et andet eksempel hvor et interface er nyttigt er når noget skal hænges op på væggen. Her kan grænsefladen beskrive det område hvor fastgørelsen kan ske.

Vi introducerer derfor klasserne **ex:WallSurface**, **ex:MountingArea** og **ex:WallIntersection** som undeklasser af **bot:Interface**.

ID-koderne er blot genereret ved at sammentrække de tilstødende objekters ID og tilføje et indeks-bogstav om nødvendigt.

Grænseflader mellem rum og vægge (**ex:WallSurface**) kan i plan visualiseres som vist herunder.

Bemærk at grænsefladen skilles hver gang en anden væg grænser op til den pågældende væg, uanset hvilken side af væggen denne situation forekommer. Denne opdeling er nødvendig idet kravene til væggen varierer ift. brugen af de tilstødende rum.



Relationer

I RDF er det muligt at beskrive relationer mellem objekter, hvormed helheden til sammen kommer til at danne en "graf" bestående af knudepunkter og forbindelser mellem dem (kanter).

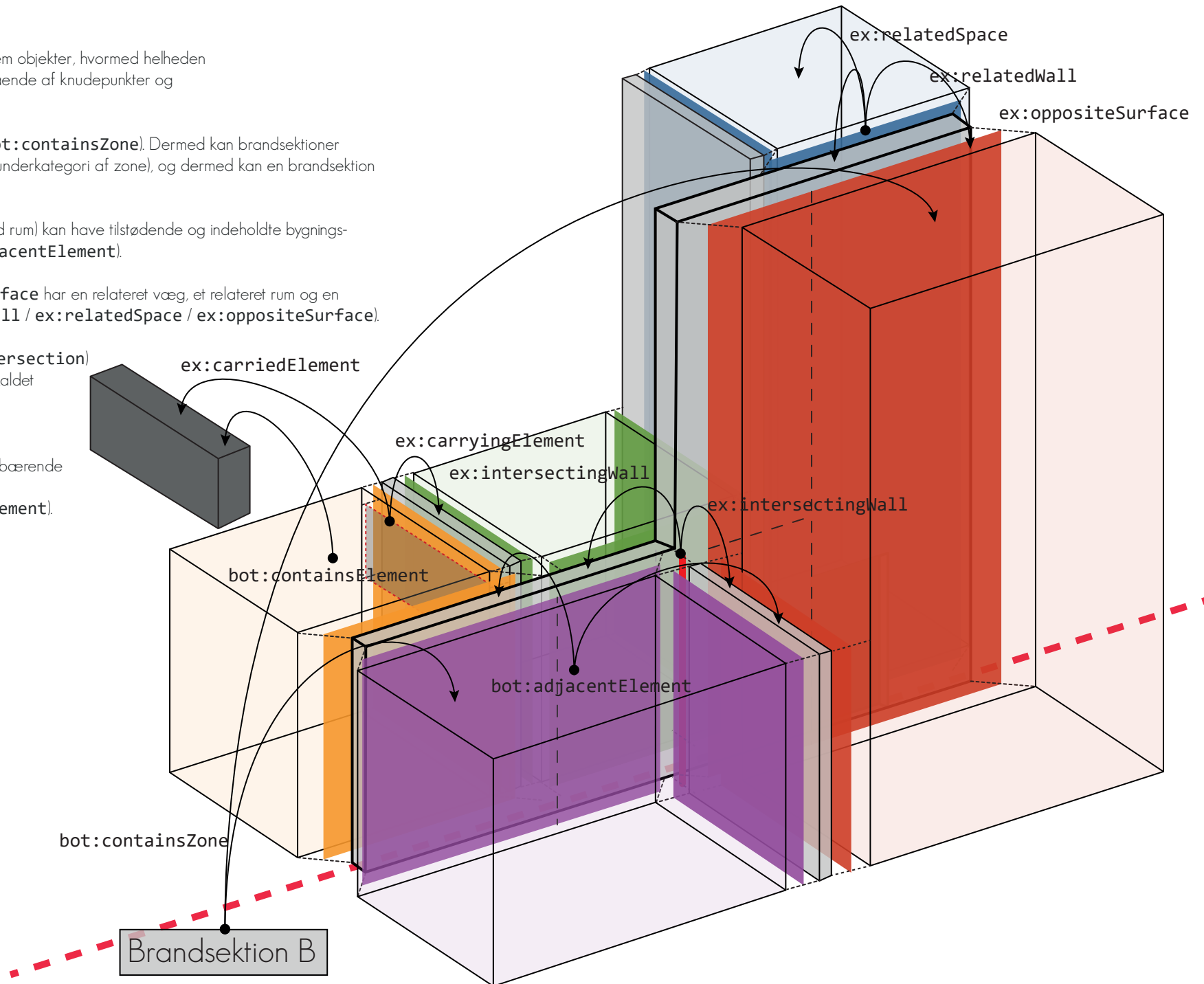
Iht. BOT kan zoner indeholde andre Zoner (**bot:containsZone**). Dermed kan brandsektioner (underkategori af zone) indeholde rum (også underkategori af zone), og dermed kan en brandsektion bruges som en aggregering af rum.

BOT beskriver endvidere at zoner (og dermed rum) kan have tilstødende og indeholdte bygningsdele (**bot:containsElement** / **bot:adjacentElement**).

Vi definerer at en grænseflade **ex:WallSurface** har en relateret væg, et relateret rum og en modstående vægoverflade (**ex:relatedWall** / **ex:relatedSpace** / **ex:oppositeSurface**).

Vi definerer at en vægsamling (**ex:WallIntersection**) har en relation til alle de tilstødende vægge kaldet **ex:intersectingWall**.

Til slut defineres det at et ophængningsfelt (**ex:MountingArea**) har en relation til den bærende bygningsdel og den bærede bygningsdel (**ex:carryingElement** / **ex:carriedElement**).



Prægranskning

Prægranskning af modellen udføres for at sikre at de nødvendige oplysninger er til rådighed inden den reelle granskning udføres. Vi udfører prægranskning i forskellige kategorier sådan at de enkelte granskninger kan vælges til/fra.

Konsistens

- Er der rum i bygningen?
 - Er der vægoverflader i modellen?
 - Er alle vægoverflader relateret til en væg?
 - Er alle vægoverflader relateret til et rum?
 - Er alle vægoverflader relateret til en modstående vægoverflade?
- (SHACL shape Precheck1-Consistency.ttl)

Geometri

- Har alle vægge egenskaben **ex:wallThickness**?
 - Har alle vægge egenskaben **ex:maxWallHeight**?
- (SHACL shape Precheck2-Geometry.ttl)

Støj

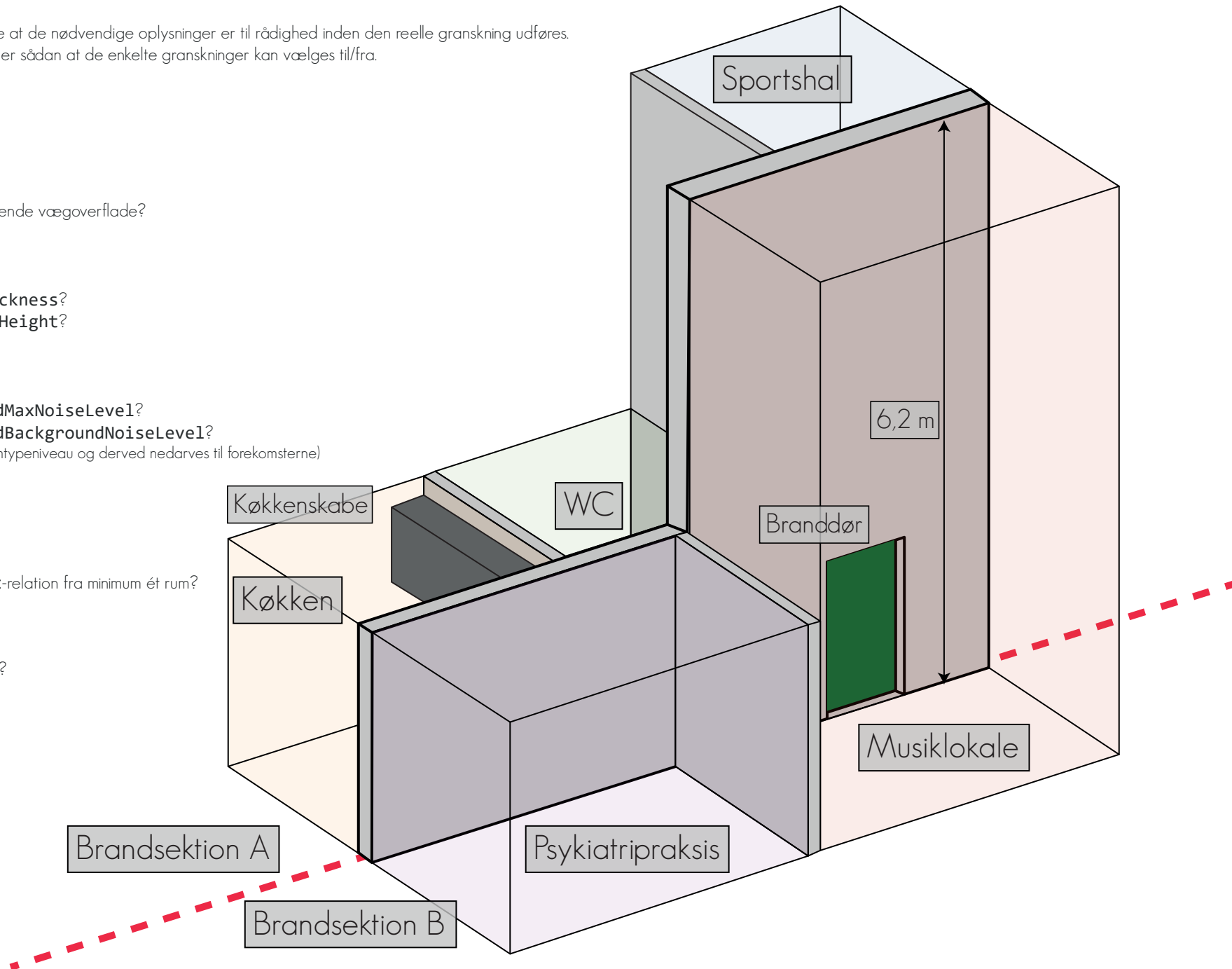
- Er der rum med egenskaben **ex:expectedMaxNoiseLevel**?
 - Er der rum med egenskaben **ex:acceptedBackgroundNoiseLevel**?
- (de to egenskaber kan potentielt beskrives på rumtypeniveau og derved nedarves til forekomsterne)
- (SHACL shape Precheck3-Noise.ttl)

Brand

- Er der brandsektioner i modellen?
 - Er alle rum knyttet til en brandsektion?
 - Har alle døre en **bot:adjacentElement**-relation fra minimum ét rum?
- (SHACL shape Precheck4-Fire.ttl)

Fugt

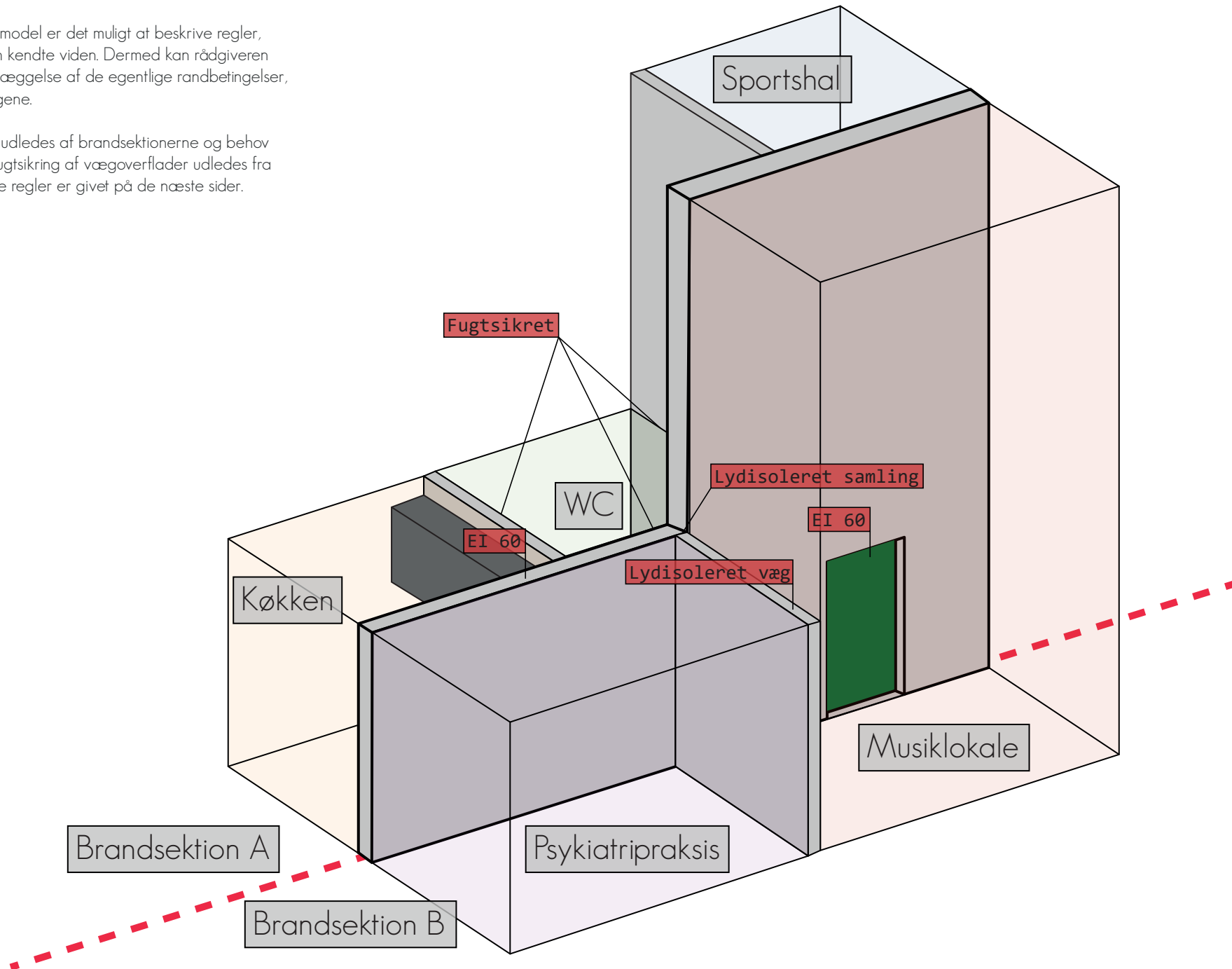
- Er der forekomster af klassen **ex:WetRoom**?
- (SHACL shape Precheck5-Moisture.ttl)



Opkvalificering

Med udgangspunkt i den semantiske bygningsmodel er det muligt at beskrive regler, som medfører at implicit viden udledes fra den kendte viden. Dermed kan rådgiveren fokusere på at tilvejebringe oplysninger til fastlæggelse af de egentlige randbetingelser, snarere end at beskrive egenskaber på væggene.

Eksempelvis kan væggenes brandklassifikation udledes af brandsektionerne og behov for lydisolering af vægge og samlinger samt fugtsikring af vægoverflader udledes fra de tilstødende rums brug. Eksempler på logiske regler er givet på de næste sider.



Opkvalificering ift. brand

Med logiske regler er det muligt at beskrive de to nedenstående regler, der udleder en væggs brandklasse ud fra de tilstødende rums brandsektion.

- 1) Hvis en væg rører et rum som er en del af en brandsektion, så rører væggen denne brandsektion.
- 2) Hvis en væg rører mere end én brandsektion, så skal væggen have brandklassifikationen EI 60

bot:containsZone

bot:adjacentElement

ex:touchesFireSection

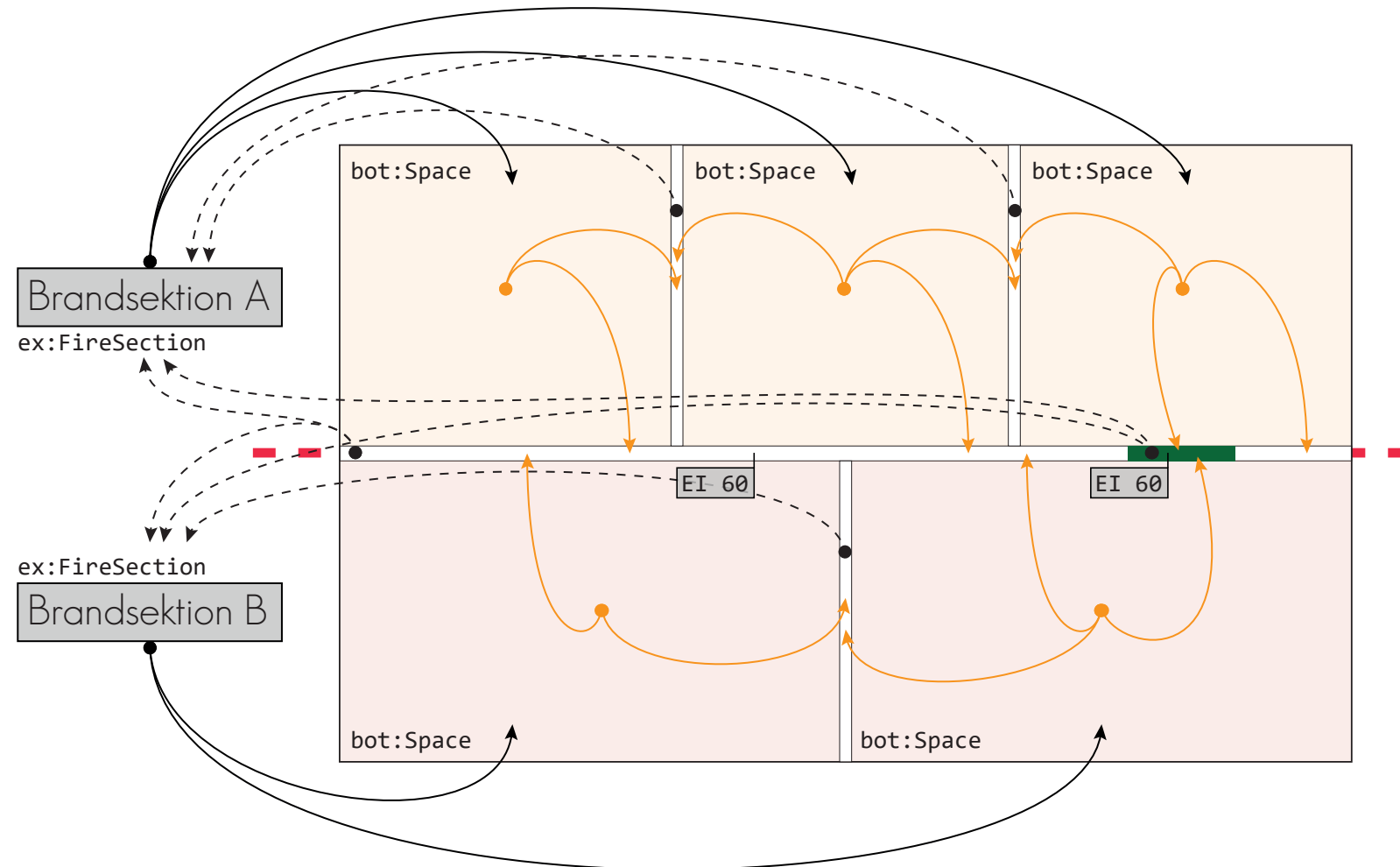


Regel 1)

```
[?el, ex:touchesFireSection, ?fs]
:-
[?fs, a, ex:FireSection],
[?sp, a, bot:Space],
[?fs, bot:containsZone, ?sp],
[?el, a, bot:Element],
[?sp, bot:adjacentElement, ?el].
```

Regel 2)

```
[?el, ex:fireClassification, ex:EI60]
:-
[?el, a, bot:Element],
AGGREGATE(
  [?el, ex:touchesFireSection, ?fs]
  ON ?el
  BIND COUNT(?fs) AS ?count
),
FILTER(?count > 1).
```



Opkvalificering ift. brand

Reglen for brandklasifikation ift. rummenes tilhørsforhold til en brandsektion var okay med den brandsektionering, som indledningsvis blev foreslået. Men hvis :SP1 flyttes til Brandsektion B, som illustreret herunder, opstår der et problem idet det kun er en del af væg :IW3, som skal overholde EI 60-kravet. Derfor giver det bedre mening at beskrive brandklassifikationen for de enkelte vægoverflader.

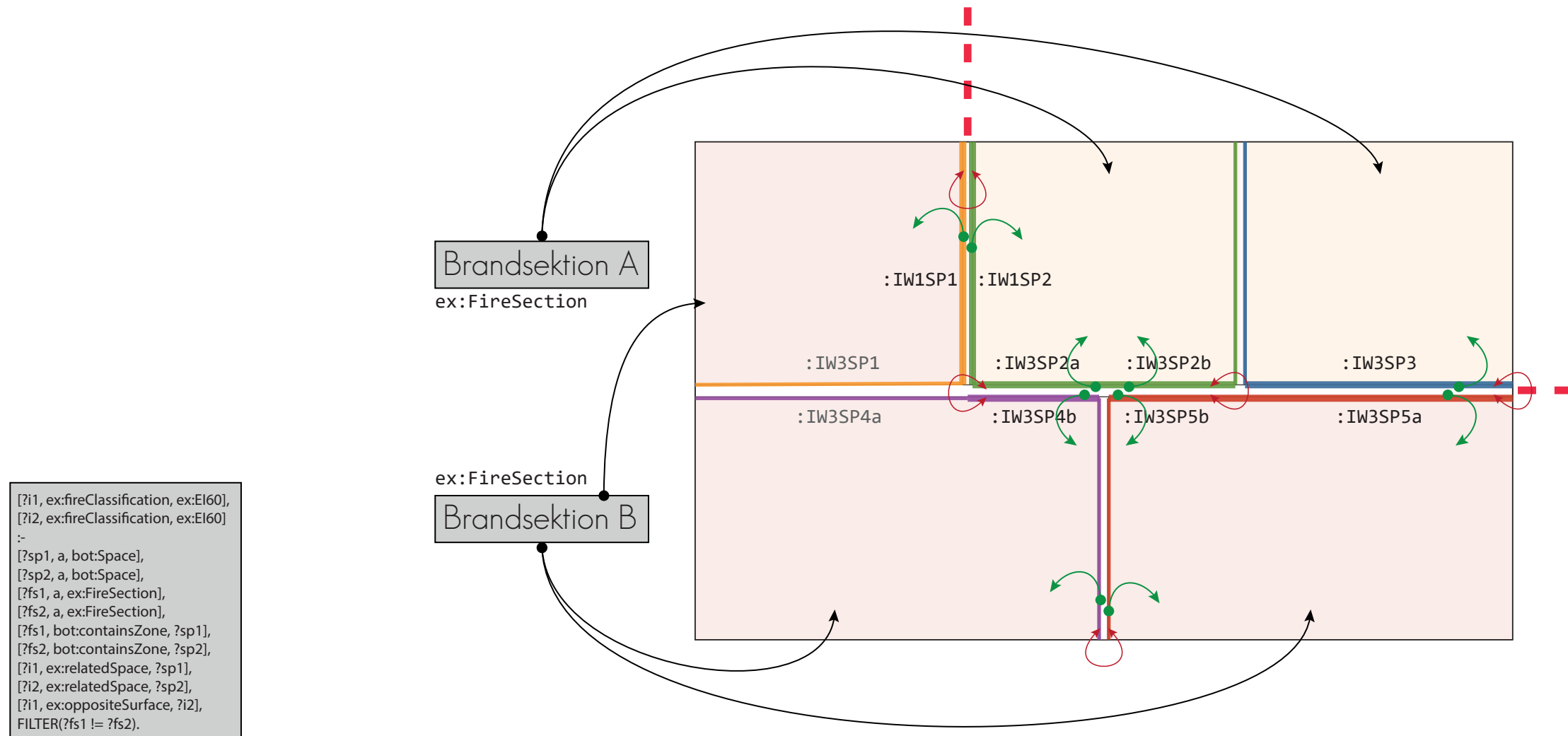
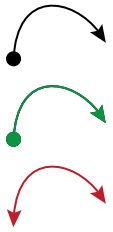
"Hvis to modstående vægoverflader rører rum som tilhører forskellige brandsektioner, så er væggen en EI 60."

Der er opbygget et test-script som dokumenterer at denne opkvalificering virker på tesmodellen: "fire-resistance.script" (se testresultat på side 12).

bot:containsZone

ex:relatedSpace

ex:oppositeSurface



Opkvalificering ift. støj

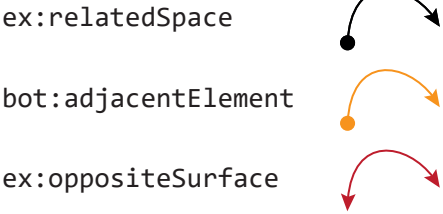
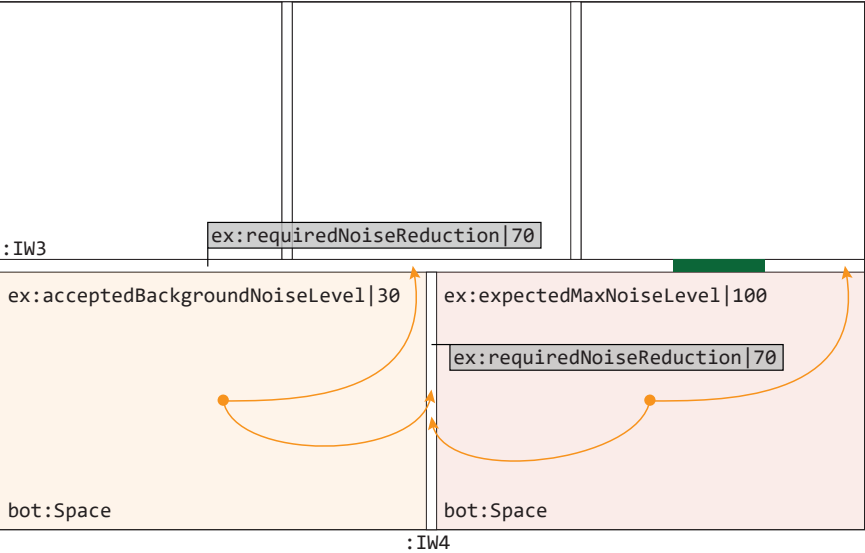
En væg der adskiller to rum skal have en støjreduktionsevne som svarer til forskellen mellem det forventede maksimale støjniveau i rum A og det accepterede niveau for baggrundsstøj i rum B.

Hvis denne regel blot beskrives på baggrund af tilstødende vægge opstår der imidlertid et problem, som er illustreret i figuren til venstre herunder. Begge rum er tilstødende til indervæggene **:IW3** og **:IW4**, men det er kun **:IW4** (og samlingen mellem **:IW3** og **:IW4**) som skal lydisoleres. Derfor giver det bedre mening at applikere reglen ift. vægoverfladerne som det er illustreret i figuren til højre. Dermed lyder reglen:

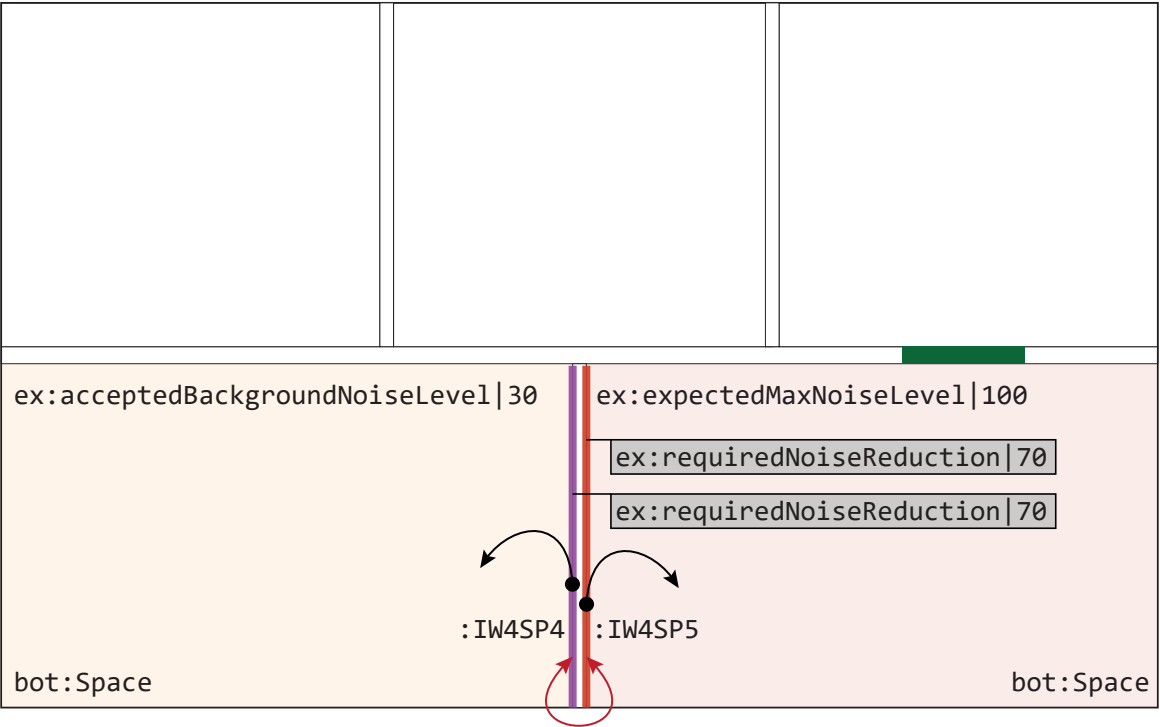
“Hvis en vægoverflade vender mod et rum, der har et maksimalt accepteret niveau for baggrundsstøj, og dens modstående vægoverflade vender mod et rum, der har en forventet støjbelastning, da gælder det at væggen, i området ved disse vægoverflader, skal lydisoleres.”

Der er opbygget et test-script som dokumenterer at denne opkvalificering virker på tesmodellen: “noise-reduction.script” (se testresultat på side 12).

```
[?el, ex:requiredNoiseReduction, ?nr]
:-
[?sp1, a, bot:Space],
[?sp2, a, bot:Space],
[?sp1, bot:adjacentElement, ?el],
[?sp2, bot:adjacentElement, ?el],
[?sp1, ex:acceptedBackgroundNoiseLevel, ?max],
[?sp2, ex:expectedMaxNoiseLevel, ?nl],
BIND(?nl-?max AS ?nr).
```



```
[?i1, ex:requiredNoiseReduction, ?nr],
[?i2, ex:requiredNoiseReduction, ?nr]
:-
[?sp1, a, bot:Space],
[?sp2, a, bot:Space],
[?i1, ex:relatedSpace, ?sp1],
[?i2, ex:relatedSpace, ?sp2],
[?i1, ex:oppositeSurface, ?i2],
[?sp1, ex:acceptedBackgroundNoiseLevel, ?max],
[?sp2, ex:expectedMaxNoiseLevel, ?nl],
BIND(?nl-?max AS ?nr).
```



Test-setup

Test af brandmodstandskrav.

```
### LIST SPACE + FIRE SECTION
Answering query from file: ./QUERIES/Space-fire-section.sparql

inst:FS1 "Brandsektion A"@da inst:SP1 "Køkken"@da .
inst:FS1 "Brandsektion A"@da inst:SP3 "Sportshal"@da .
inst:FS1 "Brandsektion A"@da inst:SP2 "WC"@da .
inst:FS2 "Brandsektion B"@da inst:SP4 "Psykiatripraksis"@da .
inst:FS2 "Brandsektion B"@da inst:SP5 "Musiklokale"@da .

### GET FIRE CLASSIFICATIONS
Answering query from file: ./QUERIES/Surface-fire-classification.sparql

inst:IW3SP1 ex:EI60 .
inst:IW3SP4b ex:EI60 .
inst:IW3SP2a ex:EI60 .
inst:IW3SP5b ex:EI60 .
inst:IW3SP2b ex:EI60 .
inst:IW3SP5a ex:EI60 .
inst:IW3SP3 ex:EI60 .
inst:IW3SP4a ex:EI60 .

### MOVE SPACE 1 TO FIRE SECTION B

### LIST SPACE + FIRE SECTION AGAIN
Answering query from file: ./QUERIES/Space-fire-section.sparql

inst:FS1 "Brandsektion A"@da inst:SP3 "Sportshal"@da .
inst:FS1 "Brandsektion A"@da inst:SP2 "WC"@da .
inst:FS2 "Brandsektion B"@da inst:SP4 "Psykiatripraksis"@da .
inst:FS2 "Brandsektion B"@da inst:SP1 "Køkken"@da .
inst:FS2 "Brandsektion B"@da inst:SP5 "Musiklokale"@da .

### GET FIRE CLASSIFICATIONS AGAIN
Answering query from file: ./QUERIES/Surface-fire-classification.sparql

inst:IW1SP2 ex:EI60 .
inst:IW1SP1 ex:EI60 .
inst:IW3SP4b ex:EI60 .
inst:IW3SP2a ex:EI60 .
inst:IW3SP5b ex:EI60 .
inst:IW3SP2b ex:EI60 .
inst:IW3SP5a ex:EI60 .
inst:IW3SP3 ex:EI60 .
```

✓ Korrekt iht. figur på side 11

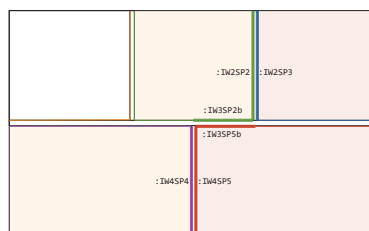
Ændret til 40 dB(A)

Korrekt iht. figur på side 10 ✓

✓ Opdateret som forventet

Flyttet fra A til B

Ændret til 30 dB(A)



✓ Korrekt iht. ovenstående figur

Korrekt iht. figur på side 10 ✓

```
### LIST SPACE | EXPECTED NOISE LEVEL | ACCEPTED BACKGROUND NOISE LEVEL
Answering query from file: ./QUERIES/Space-noise-req.sparql

inst:SP1 UNDEF UNDEF .
inst:SP2 UNDEF UNDEF .
inst:SP5 100 UNDEF .
inst:SP4 UNDEF 30 .
inst:SP3 100 UNDEF .

### LIST REQUIRED NOISE REDUCTION OF WALL SURFACES
Answering query from file: ./QUERIES/Space-noise-red-req.sparql

inst:IW4SP4 70 .
inst:IW4SP5 70 .

### CHANGE REQUIREMENT IN SP4 TO 40 dB(A)
Answering query from file: ./QUERIES/Space-noise-req.sparql

inst:SP1 UNDEF UNDEF .
inst:SP2 UNDEF UNDEF .
inst:SP5 100 UNDEF .
inst:SP4 UNDEF 40 .
inst:SP3 100 UNDEF .

### LIST REQUIRED NOISE REDUCTION OF WALL SURFACES
(SHOULD NOW BE 60 dB(A))
Answering query from file: ./QUERIES/Space-noise-red-req.sparql

inst:IW4SP4 60 .
inst:IW4SP5 60 .

### SET REQUIREMENT IN SP2 TO 30 dB(A)
Answering query from file: ./QUERIES/Space-noise-req.sparql

inst:SP1 UNDEF UNDEF .
inst:SP2 UNDEF 30 .
inst:SP5 100 UNDEF .
inst:SP4 UNDEF 40 .
inst:SP3 100 UNDEF .

### LIST REQUIRED NOISE REDUCTION OF WALL SURFACES
(SHOULD NOW BE 60 dB(A))
Answering query from file: ./QUERIES/Space-noise-red-req.sparql

inst:IW3SP2b 70 .
inst:IW2SP3 70 .
inst:IW2SP2 70 .
inst:IW3SP5b 70 .
inst:IW4SP4 60 .
inst:IW4SP5 60 .
```

Regeltjek

Når prægranskning er gennemført og modellen er blevet opkvalificeret (opkvalificering kan eventuelt ske før og efter prægranskning, og potentielt kan prægranskning også gøres i to etaper) er den klar til det egentlige regeltjek.

De fire regler der ønskes undersøgt er vist i skemaet herunder. Reglerne er nøje udvalgt så de favner de fire kategorier "Geometri", "Overflader", "Egenskaber" og "Fysisk egenskab".

	Geometri	Overflader	Egenskaber	Fysisk egenskab
Regel 1	En væg må ikke være for høj			
Funktion	Højde i relation til tykkelsen			
Krav til regel 1	Der skal være en geometri/placeholder der repræsenterer en væg			
Regel 2		Hvis der er vægophængte elementer så skal væggen have en forstærkende overflade		
Krav til regel 2		Køkkenelementer skal være modelleret		
Regel 3			Hvis rum x støder op til rum z og der er forskel i de akustiske krav, så skal væggen have akustisk isolering	
Krav til regel 3			Rummenes funktion skal være defineret	
Regel 4				En væg skal være stærkere hvis der er en døråbning
Krav til regel 4				Døre og vinduer skal være modelleret

Regel 1: Geometri

Overholdelse af første regel kan tjekkes udelukkende på baggrund af geometri.

Indervæggen kan være selvbærende så længe visse krav til forholdet mellem højde og tykkelse overholdes. En sådan forespørgsel kan formuleres således:

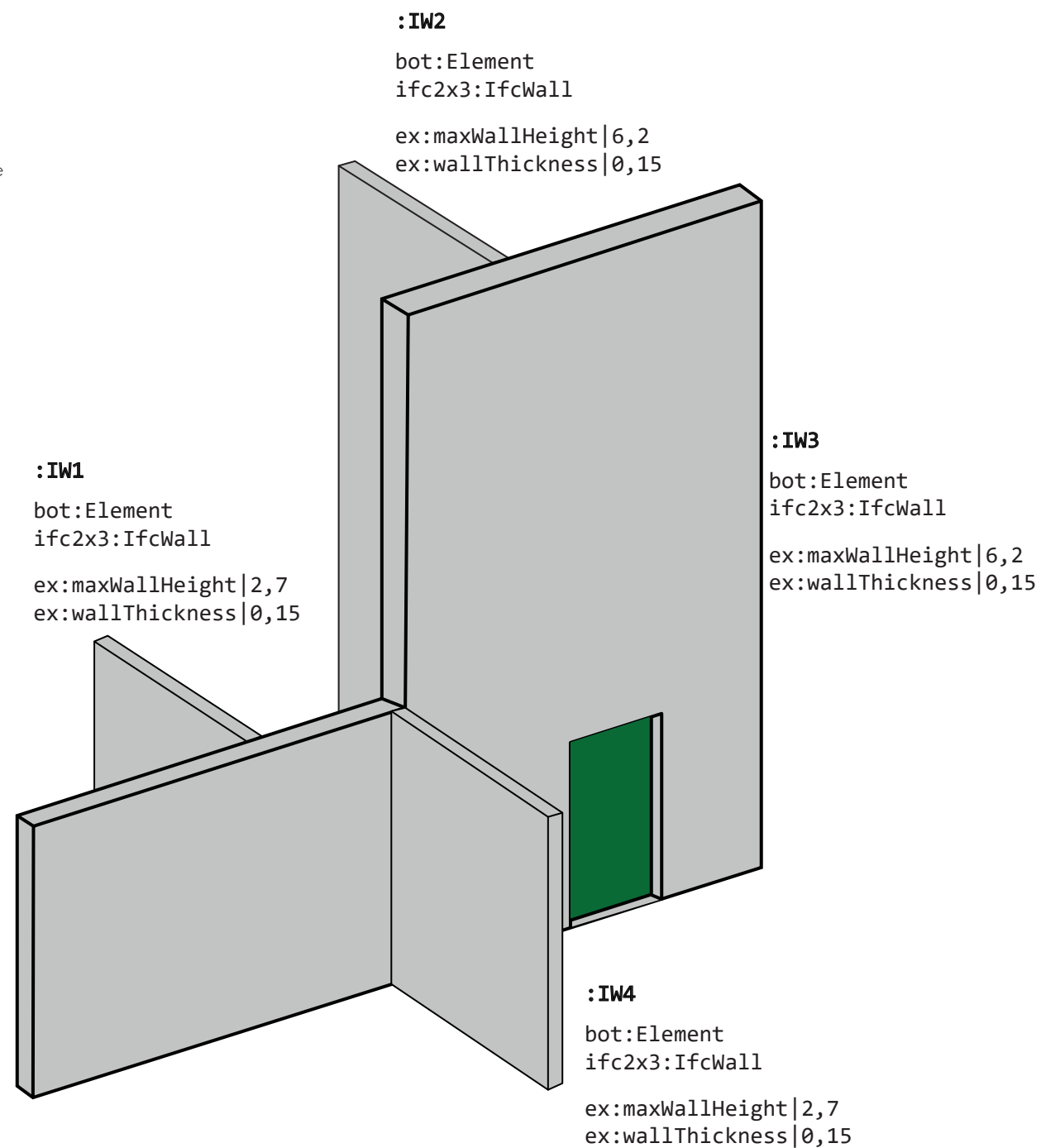
Dataforespørgsel

Returner alle vægge inkl:

- væggenes højde (**ex:maxWallHeight**)
- væggenes tykkelse (**ex:wallThickness**)

Dataforberedelse

- Vægelementer skal være klassificeret som sådan (**IfcWall**)
- De geometrisk afledte egenskaber skal være tilgængelige. Disse kan komme fra BIM-værktøjet, eller kan efterfølgende afledes af geometrien.



Regel 2: Overflader

Overholdelse af anden regel kræver at vi ved noget om de bygningsdele der grænser op imod væggene. Specifikt er der brug for at kende til elementer som bæres af væggen. Her er vi afhængige af den forbehandling af modellen som udtrækker **ex:MountingArea**-grænseflader.

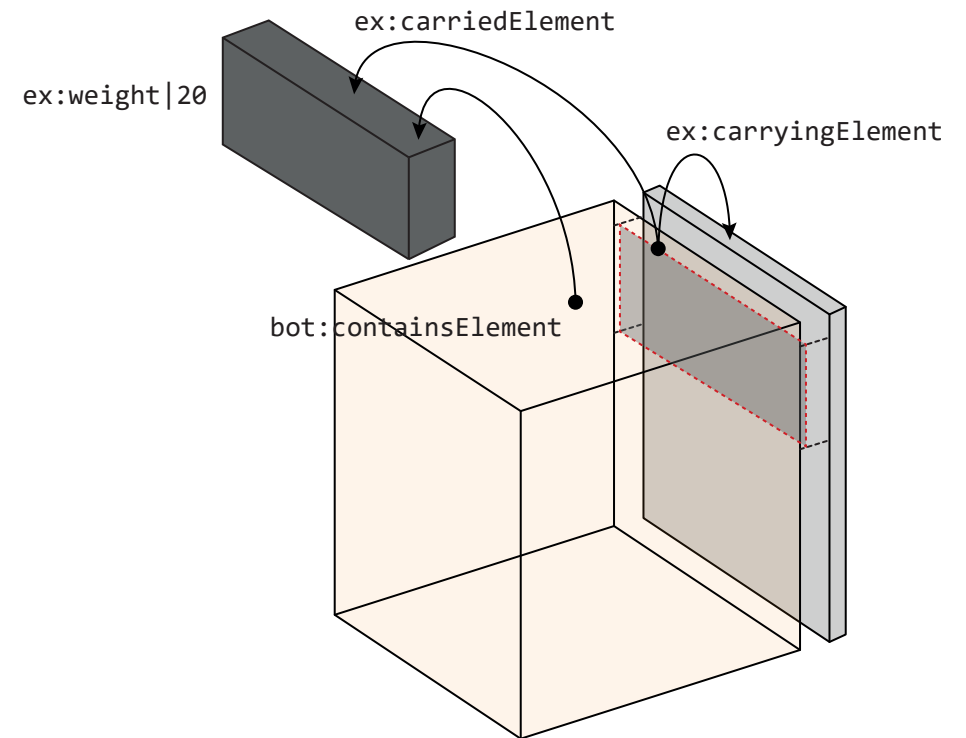
Dataforespørgsel

Returner alle vægmonteringsområder (**ex:MountingArea**) inkl:

- Arealet af området (hvor stort et område har vi til at ophænge elementet?)
- Vægten af det bærede element

Dataforberedelse

- Vægmonteringsområder skal skabes af en algoritme ud fra følgende:
 - Identificer vægmonterede bygningsmodele (hosted elements)
 - Beregn den fælles overflade hvor væg og bygningsdel mødes
 - Beregn arealet af overfladen
- Vægten af den bærede bygningsdel skal være angivet



Regel 3: Egenskaber

Overholdelse af tredje regel kræver at vi ved noget om de rum der grænser op imod væggene. Specifikt er der brug for at kende til de forventede forekommende støjniveauer og de maksimalt accepterede niveauer af støj overført fra tilstødende rum.

Som beskrevet på side 11, er det muligt at udlede krav til støjsolering ud fra forhold i de enkelte rum. Derfor består selve regeltjekket blot i at finde vægoverflader (**ex:WallSurface**) og samlinger (**ex:WallIntersection**) med krav om støjdæmpning.

Dataforespørgsel

Returner alle vægoverflader (**ex:WallSurface**) inkl:

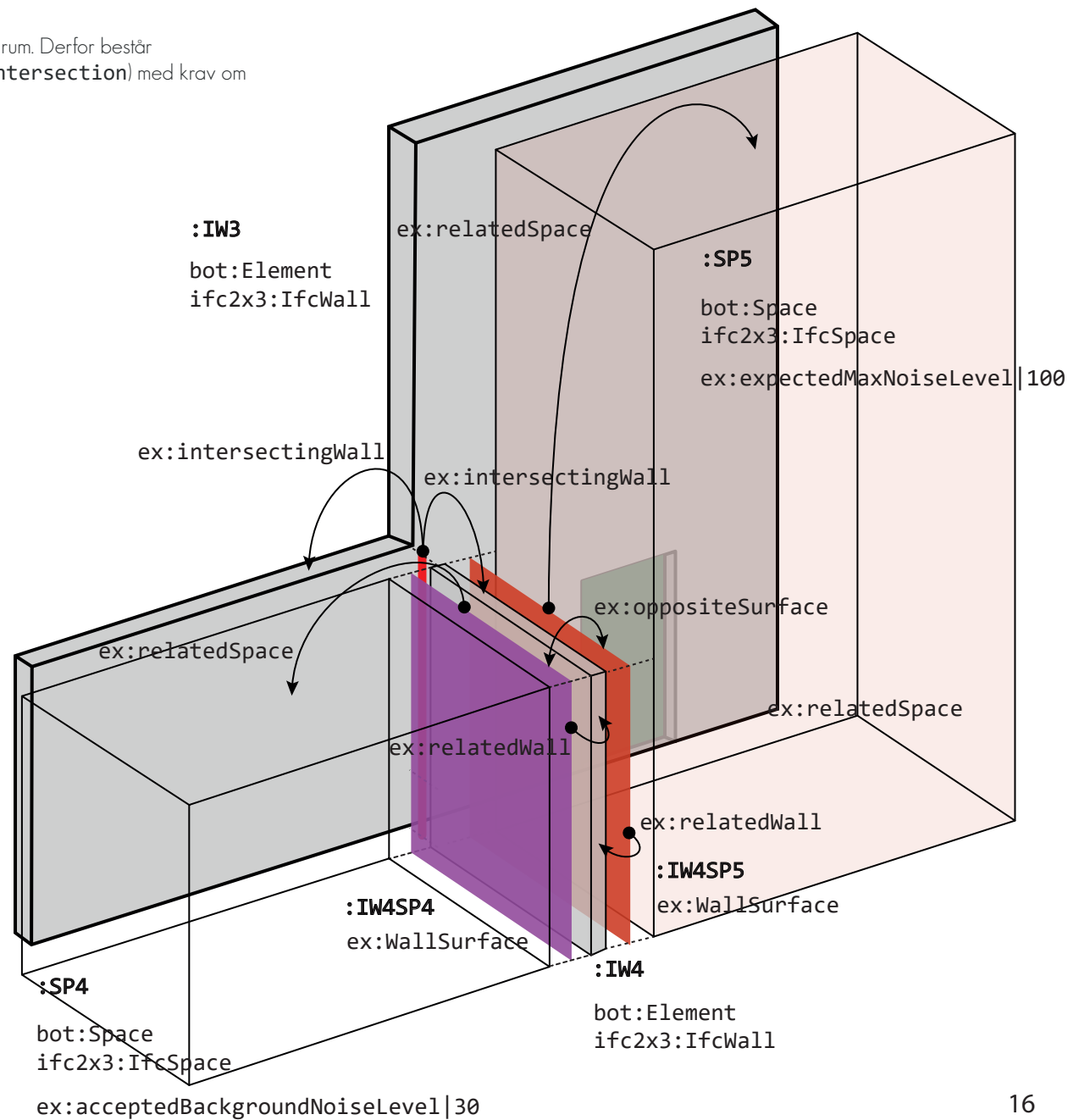
- Krav til støjdæmpning

Returner alle vægsamlinger (**ex:WallIntersection**) inkl:

- Krav til støjdæmpning

Dataforberedelse

- Vægsamlinger skal skabes af en algoritme som finder tilstødende vægge
- Vægoverflader og deres modstående part skal skabes af en algoritme
 - IFC Space boundaries er et muligt udgangspunkt



Regel 4: Fysisk Egenskab

Overholdelse af fjerde regel kræver at vi ved noget om døren og den væg den er monteret i.

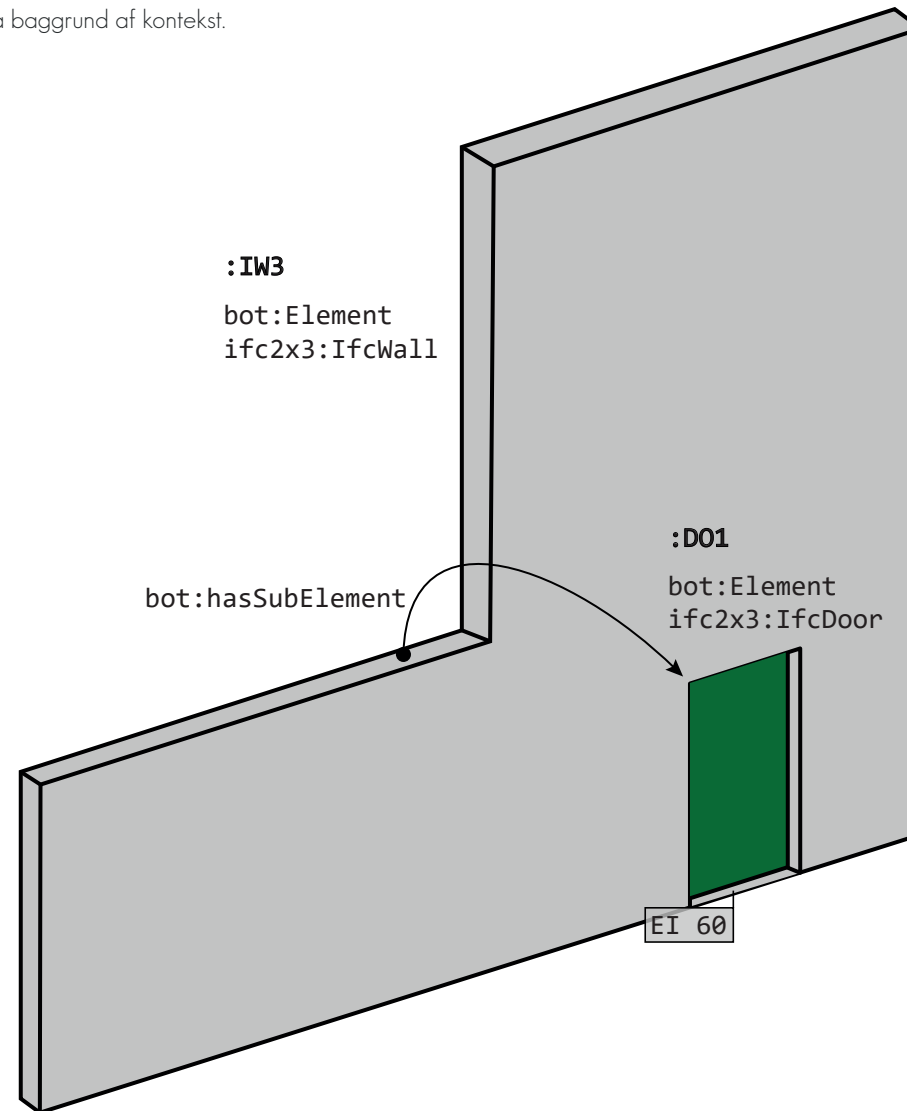
Som beskrevet på side 9-10, er det muligt at udlede krav til brandresistans ud fra de tilstødende rums brandsektioner.
En branddør er tungere end en almindelig dør, og denne information kan derfor tilvejebringes på baggrund af kontekst.

Dataforespørgsel

Returner alle vægge som hoster en dør inkl:
- Dørens vægt (er det en branddør?)

Dataforberedelse

- Brandklassifikation af døre og vægge kendes ud fra brandsektionering
- Væg/dør- relation kendes gennem IFC-fil



Roadmap

Med dette arbejde har vi defineret en RDF-baseret datamodel som udbygger BOT og IFC med specifik terminologi til at beskrive de forhold som er nødvendige for at kunne beskrive randbetingelser specifikt for gipsvægge i bygninger. Denne terminologi er beskrevet i en OWL-ontologi som kan gøres tilgængelig for branchen - udestående spørgsmål er blot hvilken instans der skal udgive den.

Følgende mangler stadig at blive undersøgt:

- Ejerforhold og adgang

- Datamodellen er linked data-baseret, og kan derfor distribueres så data ligger hos den aktør der har skabt dem. Med SOLID* er dette muligt, og forholdene omkring dette kunne undersøges nærmere

- Arkitektur

- Opkvalificering og samling af datasæt kunne bygges som en web-service. Klienter (Revit, Archicad, Excel) kan berige projektmodellen, og resultat af prægranskning leveres tilbage
- Når modellen har de fornødne informationer kan denne sendes til diverse producenter som melder retur med produkter/assemblies som kan opfylde kravene
- Er der forhold ved de tilgængelige produkter/assemblies som afviger fra grundgeometrien (ex. vægtykkelse) der afviger fra det modellerede, skal klienten informeres
- Findes der ingen produkter/assemblies som kan opfylde kravene skal klienten informeres

- Dataudtræk

- Der findes offentligt tilgængelige værktøjer som kan udtrække grundlæggende informationer fra en IFC-fil til RDF, men der skal udvikles særlige scripts til udtræk af de i dette projekt definerede grænseflader

- Ejerskab

- Interaktion

* <https://solidproject.org/>

