



TEKIS-LV – Begrepps- och tillämpningsmodell

Dokumentnr. 1-02
Beteckning. [MOD]
Rev. 2.2
Utgivare Tekis AB
Ändrat 2007-05-21

Ändringsförteckning

Rev	Datum	Kommentar
1.1	2002-12-20	Första version av dokumentet.
1.2	2003-01-22	Reviderad med avseende på de remissvar som inkom beträffande version 1. Anpassad till version 1.0.87 av databasmodellen.
1.3	2003-03-07	Teckenförklaring till ER-modell
1.4	2003-09-16	Ny indelning med begrepps-, tillämpnings- och lagringsmodell. Begreppsmodell beskriven med STANLI-notation och tillämpningsmodellen beskriven med UML. Databasdiagram och specifikationen utbrutet till bilaga 1. Tillagt index.
1.5	2003-09-23	Kompletterad med fullständiga tillämpnings- och begreppsmodeller.
1.6	2003-10-24	Uppdaterad tillämpningsmodell
1.7	2004-07-02	Uppdaterad inför leverans av ”Externa företeelser” samt genomgången inför leverans. Uppdaterat avsnitt 4.9, lagt till avsnitten 4.11 och 5.8
1.8	2004-09-23	Uppdaterad inför slutleverans. Tagit bort alla beskrivningar av ej färdig funktionalitet. Uppdaterad beskrivning av användare och behörighet samt verksamhet och LIVO.
1.9	2004-10-14	Kompletteringar efter synpunkter på slutleverans
2.2	2007-05-21	Uppdaterad inför release av version 2.2 av TEKIS-LV.

Innehållsförteckning

ÄNDRINGSFÖRTECKNING.....	2
INNEHÅLLSFÖRTECKNING.....	3
1. INLEDNING.....	5
2. TERMER OCH DEFINITIONER	6
2.1. FÖRKORTNINGAR	6
3. BEGREPPSMODELL.....	8
3.1. BEGREPPSMODELL, JÄMFÖRANDE TABELL	8
3.2. OBJEKT.....	10
3.2.1. <i>Versioner</i>	11
3.3. OPERATIONER OCH FÖRÄNDRINGAR.....	13
3.4. JOBB.....	14
3.4.1. <i>Lås/konflikthantering</i>	15
3.5. AKTÖR	16
3.5.1. <i>Användare och behörigheter</i>	16
3.5.2. <i>Funktion och applikation</i>	17
3.5.3. <i>Intressenter</i>	17
3.6. TRAFIKNÄT	17
3.6.1. <i>Verksamhetsnät och LIVO</i>	18
3.7. NÄTANKNUTEN FÖRETEELSE.....	20
3.8. VERKSAMHET.....	21
3.9. DATAKATALOG	22
3.10. DETALJERINGSNIVÅER	23
4. TILLÄMPNINGSMODELL.....	26
4.1. TILLÄMPNINGSMODELL, JÄMFÖRANDE TABELL.....	27
4.2. FÖRÄNDRINGSBART OBJEKT.....	28
4.2.1. <i>Objekt</i>	29
4.2.2. <i>Redigeringsversioner</i>	29
4.2.3. <i>Utförd förändring</i>	30
4.3. JOBBHANTERING	30
4.3.1. <i>Jobb</i>	31
4.4. AKTÖR	32
4.4.1. <i>Session</i>	32
4.4.2. <i>Applikation</i>	33
4.5. VERKSAMHET.....	33
4.6. TOPOLOGI.....	33
4.6.1. <i>Nätelement</i>	34
4.6.2. <i>Port</i>	36
4.6.3. <i>Referenslänksport</i>	37
4.6.4. <i>Referensnodsport</i>	37
4.6.5. <i>Referenslänk</i>	37
4.6.6. <i>Referenslänksdel</i>	38
4.6.7. <i>Referensnod</i>	40
4.7. NÄTANKNUTEN FÖRETEELSE.....	41
4.7.1. <i>Företeelse</i>	42
4.8. DATAKATALOG	44
4.8.1. <i>Företeelsetyp</i>	44
4.8.2. <i>Attributtyp</i>	48
4.8.3. <i>Attribut</i>	49

4.9.	EXTERNA FÖRETEELSETYPER	50
4.9.1.	Externa företeelser - företeelsetyp	50
4.9.2.	Externa företeelser – attributtyp	51
4.9.3.	Externa företeelser – knytning till extern företeelse	51
4.10.	UTBREDNINGAR	52
4.10.1.	Punktutbredning	55
4.10.2.	Linjeutbredning	56
4.10.3.	Vägutbredning	56
4.10.4.	Nodutbredning	57
4.10.5.	Svängutbredning	57
4.11.	DETALJERINGSNIVÅER	58
	BILAGA A, FULLSTÄNDIG TILLÄMPNINGSMODELL	59
	REFERENSER	60

1. Inledning

Det här dokumentet beskriver de begrepp som TEKIS-LV berörs av och hanterar samt hur de tillämpas i systemet. Begreppen är i de flesta fall desamma som de som återfinns i de svenska standarderna SS 637004-1 [1], SS 637006 [2] och SS 637007 [3].

Där begreppen avviker från svensk standard ges tydlig information om avsteget och en utförlig beskrivning av hur begreppet definieras i TEKIS-LV presenteras.

I beskrivningen av begreppen används, så kallad, STANLI-notation för att grafiskt presentera begreppen. Analogt används notationen som föreskrivs i UML-standarderna för tillämpningsmodellen.

2. Termer och definitioner

I det här dokumentet används nedanstående termer med tillhörande definitioner. Definitionerna är främst hämtade från handboken *Samverkande GIS med ISO 19100 – en handbok om tekniskt ramverk för geografisk information* [4] och ordboken *GIS-ordboken* [5]. Den förstnämnda är samma källa som de svenska standarderna SS 637004-1 [1], SS 637006 [2] och SS 637007 [3] har för sina begreppsdefinitioner. Till varje definition finns det en referens som återger källan till definitionen.

begreppsmodell

modell som förklarar begrepp och anger deras inbördes relationer [4]

länk

endimensionellt topologiskt grundelement med definierad riktning och som begränsas av en eller två noder [5].

länk/nod-baserad topologi

en modell där topologi beskrivs med hjälp av länkar och noder och där länkarna utgör förbindelser eller kopplingar mellan noder.

Anm. Inom matematiken benämns topologimodellen grafteori.

metadata

uppgifter om en datamängd [4]

nod

nolldimensionellt topologiskt grundelement [5]

tillämpningsmodell

formell beskrivning av datastruktur, regler och innehåll för information inom ett visst tillämpningsområde [4]

topologi

beskrivning av hur geografiska objekt ansluter eller angränsar till varandra [5]

2.1. Förkortningar

Förkortning	Beskrivning
GML	Geography Markup Language - XML-baserat sätt att representera geografiska geometriska data. [1]
ISO	The International Organization for Standardization
NVDB	Nationell Vägdatabas – Officiell svensk rikstäckande grunddata-bas omfattande statliga, kommunala och privata vägar. [1]
OGC	Open GIS Consortium. Internationellt standardiseringsorgan för geografisk data.

SIS	Swedish Standards Institute. Svenskt organ för officiell standardisering
Stanli	Projektområde inom SIS som ägnar sig åt geografisk information. [1]
UML	Unified Modeling Language – objektorienterat analys- och designspråk framtaget av Object Management Group (OMG). [1]
XML	Extensible Markup Language – förenklad variant av SGML anpassad för användning på Internet. SGML är en standard för strukturering av dokument. [1]

3. Begreppsmodell

Begreppsmodellen förklarar innebörden för de centrala begreppen i TEKIS-LV. Strukturen i det här avsnittet återfinns sedan även i avsnittet *Tillämpningsmodell*. Detta för att läsaren enklare ska kunna se hur begreppen är tillämpade i TEKIS-LV.

Inledningsvis jämförs de begrepp som nyttjas i TEKIS-LV med begrepp i svenska standarder eller liknande källor. Därefter ges en mer ingående beskrivning av begreppen under respektive rubrik.

3.1. Begreppsmodell, jämförande tabell

I tabellen nedan presenteras samtliga begrepp som används i TEKIS-LV. I de fall som begreppet finns definierat i någon tillförlitlig källa, såsom de svenska standarderna SS 637004-1 [1], SS 637006 [2] och SS 637007 [3], hänvisas källan till. I övriga fall ges en kortfattad beskrivning av begreppet. Utförligare definitioner av de begrepp som nyttjas i TEKIS-LV återfinns senare i avsnittet.

Begrepp	Källa	Beskrivning
aktör	[saknas]	En extern betraktare eller påverkare av ett system.
attribut	SS 637006 [2]	Se hänvisad källa.
attributtyp	SS 637006 [2]	Se hänvisad källa.
detaljeringsnivå	SS 637004-1 [1]	Se hänvisad källa.
egen företeelsetyp	[saknas]	En företeelsetyp som ägs av den aktuella verksamheten.
extern företeelsetyp	[saknas]	En företeelsetyp för vilken värdena för minst ett attribut lagras externt, det vill säga utanför datakatalogstrukturen i TEKIS-LV.
främmande företeelsetyp	[saknas]	En publik företeelsetyp som ägs av någon annan verksamhet än den aktuella.
företeelse	SS 637006 [2]	Se hänvisad källa.
företeelsetyp	SS 637006 [2]	Se hänvisad källa. Företeelsetyp motsvaras i standarden av begreppet objekttyp.
förändring	SS 637007 [3]	Se hänvisad källa.
gatulänk	SS 63 70 04, Nätförbindelse	Se hänvisad källa.

gatunod	SS 63 70 04, Nätknut	Se hänvisad källa.
grundgatunät	[saknas]	Den topologiska och geografiska grundnivå som används för nätredigering. En verksamhet kan välja att göra en egen verksamhetsindelning av grundgatunätet.
huvudattribut	[saknas]	Det "väsentligaste" attributet hos en företeelsetyp och som bör användas i presentationer där enbart ett attribut ingår.
intressent	[saknas]	En aktör som interagerar med systemet utan att utföra några operationer i det.
intern företeelsetyp	[saknas]	Subtyp av nätanknuten förteelse som lagras helt i LV.
jobb	[saknas]	En transaktion av förändringar vilka syftar till att skapa en mer korrekt beskrivning av verkligheten.
linjärt verksamhetsobjekt (LIVO)	[saknas]	Motsvarar gatulänk i en verksamhetsindelning av grundgatunätet.
nätkomponent	SS 637004-1 [1]	Se hänvisad källa.
objekt	SS 637006 [2]	En entitet vars existens inte beror av någon annan entitets existens.
operation	[saknas]	En åtkomst eller redigering av ett objekt.
populärnamn	[saknas]	Det utåt identifierbara namnet på ett LIVO
privat företeelsetyp	[saknas]	En företeelsetyp som enbart är tillgänglig för den verksamhet som äger den.
publik företeelsetyp	[saknas]	En företeelsetyp som är tillgänglig för samtliga verksamheter.
redigeringsversion	[saknas]	En beskrivning av den upplevda verkligheten under en viss tid.
referenssystem	[saknas]	Indelning av grundgatunätet som är tillämplig för en verksamhet.
tidsversion	SS 637004-1 [1]	Se hänvisad källa.

trafiknät	SS 637004-1 [1]	Se hänvisad källa.
transaktion	SS 637007 [3]	Se hänvisad källa.
verksamhet	[saknas]	En verksamhet är en organisation eller delorganisation som är ägare av företeelsetyper samt har möjlighet att för sin räkning äga en verksamhetsindelning av gatunätet. Verksamhet är liksom betraktelsesdatum en systeminställning i en LV-applikation.
verksamhetsnät	[saknas]	Indelning av grundgatunätet som är tillämplig för en verksamhet.
värdeomän	SS 637006 [2]	Se hänvisad källa.

3.2. Objekt

I TEKIS-LV hanteras en mängd olika typer av information. Där det är möjligt hanteras dessa olika typer av information på ett gemensamt sätt. Med andra ord finns det en abstraktion eller generalisering av informationen som hanteras i systemet och som tillämpas när så är brukligt.

Nyttan av abstraktionerna i TEKIS-LV är att funktionalitet kan tas fram som fungerar på ett liknande sätt för olika typer av objekt. Det går alltså att återanvända idéer och lösningar som är framtagna för en viss informationstyp när andra informationstyper ska hanteras. I TEKIS-LV hanteras alla objekt på ett gemensamt sätt vad det gäller:

- historik,
- redigeringskonflikter,
- långa transaktioner,
- behörighet,
- loggning,
- prenumerationer samt
- synkroniseringar.

Funktionaliteten ovan utgör sammantaget den förändringshantering som finns och är en central del av systemet.

Begreppet objekt har givits följande definition i TEKIS-LV:

Definition 1: Ett *objekt* är en entitet vars existens ej beror av någon annan entitets existens.

Begreppet har en motsvarighet i den svenska standarden SS 647006 [2] och benämns där FI_FeatureInstance.

Det finns tre olika grundtyper av förändringsbara objekt i TEKIS-LV vilka beskrivs i fig 4.1.

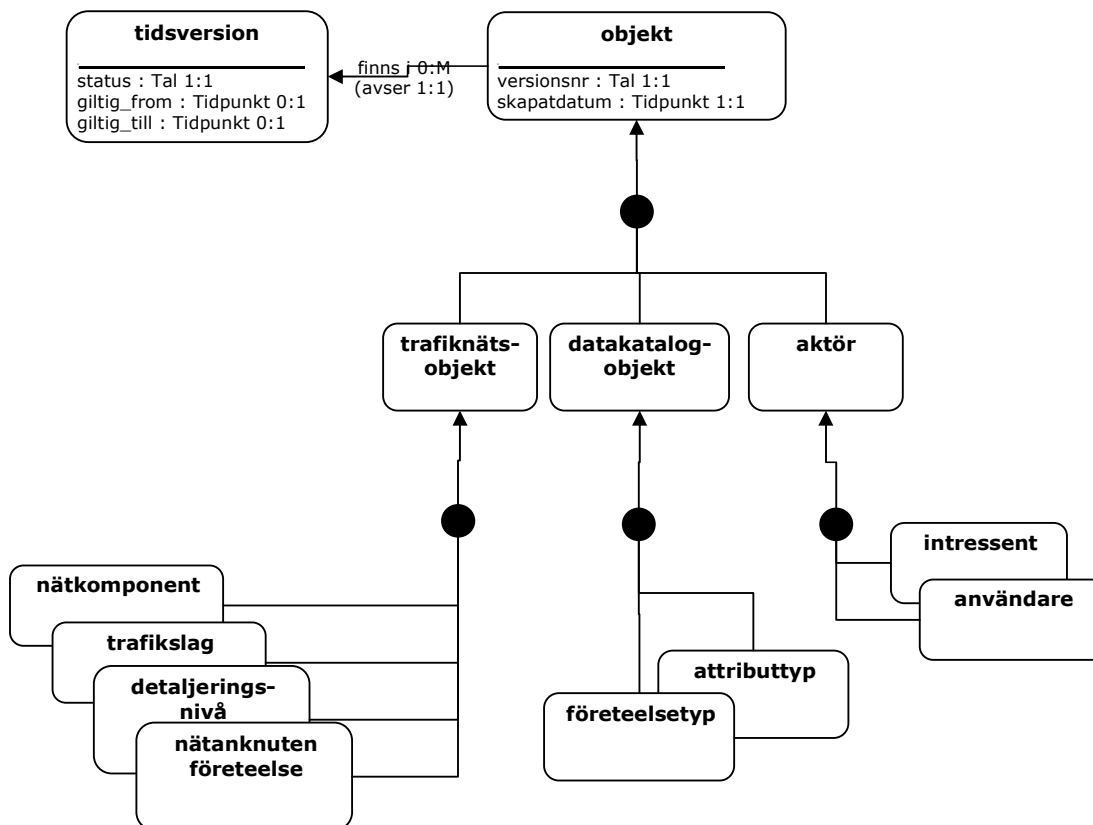


Fig 3.1: Begreppsgraf - förändringsbart gatuobjekt.

3.2.1. Versioner

TEKIS-LV har en avancerad versionshantering som innefattar dels en förändringshistorik och dels en verklighetshistorik. Förändringshistoriken beskriver hur uppfattningen av verkligheten har förändrats över tiden medan verklighetshistoriken beskriver hur verkligheten har förändrats över tiden. I fig 4.2 nedan återfinns ett exempel som belyser skillnaden mellan de båda historikerna.

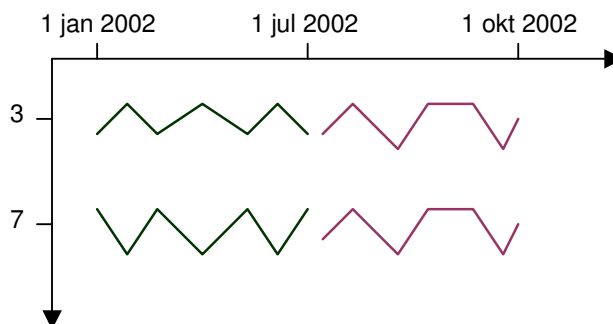


Fig 3.2: En företeelses förändrings- och verklighetshistorik.

Figuren innehåller en beskrivning av ett viltstängsel dels verklighetshistoriskt och dels förändringshistoriskt. Horisontellt går det att utläsa hur viltstängslets utseende har förändrats över tiden och vertikalt går det att utläsa hur uppfattningen av viltstängslets utseende har förändrats över tiden. Med andra ord har viltstängslet aldrig haft det utseende som återfinns i version 3 mellan den första januari 2002 och den första juli 2002. Däremot uppfattades viltstängslets utseende under tiden mellan version 3 och 7 som sådant.

I TEKIS-LV lagras historik både ur ett verklighetsperspektiv och ur ett förändringsperspektiv för alla typer av objekt. Verklighetshistoriken hanteras genom giltighetstider hos de olika typerna av objekt medan förändringshistoriken hanteras genom versioner av objekten. Begreppet version definieras som:

Definition 2: En *version* är en beskrivning av den upplevda verkligheten under en viss tid.

En objektversion har två datumattribut vilka beskriver datumet då objektversionen skapades respektive datumet då objektversionen togs bort. Därutöver har en objektversion även en objektstatus. De status som är möjliga är historisk, gällande, under bearbetning och föreslagen och en beskrivning av dessa ges i tabell 4.2.

Status	Beskrivning
Historisk	En historisk objektversion är en version av ett objekt vilken har upphört att gälla genom att den ersatts av en ny gällande version.
Gällande	En gällande objektversion är en gällande version av ett objekt och har således godkänts som en mer korrekt beskrivning av objektet och som ännu inte ersatts av någon ny version.
Under bearbetning	En objektversion som är under bearbetning är en kopia av en gällande version av objektet vilken går att förändra och sedan genom att exempelvis checka in jobbet som ansvarar för den nya versionen går att föreslå som en ny gällande

	version. (Se avsnittet <i>Jobb</i> för mer information om jobb.)
Föreslagen	En föreslagen objektversion är en version av ett objekt vilken är föreslagen att ersätta en för närvarande gällande objektversion.

Tabell 3.1: Beskrivning av de status en objektversion kan ha i TEKIS-LV.

3.3. Operationer och förändringar

Allt nyttjande av informationen i TEKIS-LV sker i form av operationer, där begreppet operation definieras som:

Definition 3: En *operation* är en åtkomst eller förändring av ett objekt.

Begreppen åtkomst och förändring definieras i sin tur som:

Definition 4: En *åtkomst* är en operation vilken läser information om objektet som opereras på.

Definition 5: En *förändring* är en operation vilken modifierar objektet som opereras på.

En övergripande bild av sambandet mellan de olika begreppen och deras beståndsdelar återfinns i fig 4.3 nedan.

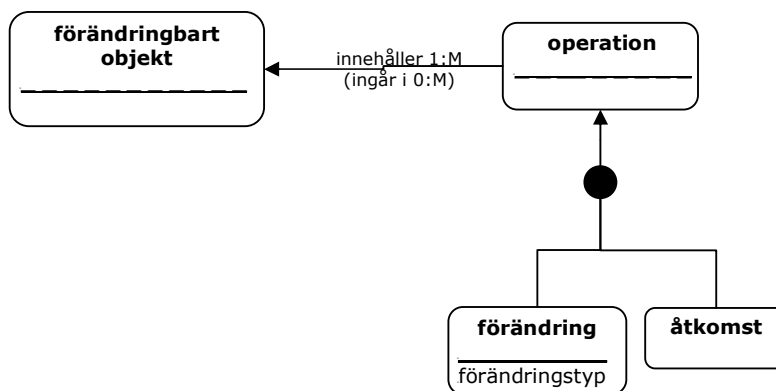


Fig 3.3: Begreppsgraf – operation.

Åtkomst används i TEKIS-LV för beskrivning av behörighet för aktörer.

Förändringar implementeras genom användning av *jobb*, se avsnitt 4.4.

3.4. *Jobb*

Förändringar hanteras i TEKIS-LV av jobb genom att till varje jobb finns det en associerad version vilken jobbet ansvarar för. Begreppet jobb definieras som:

Definition 6: Ett *jobb* är en transaktion av förändringar vilka syftar till att skapa en mer korrekt beskrivning av verkligheten där förändringarna har en inbördes korrelation.

Med andra ord kan ett jobb exempelvis vara projekteringen av en ny del i vägnätet, en ombyggnation av ett vägvagnsnitt eller en överföring av information från ett annat system. Alla förändringar av objekt med version i TEKIS-LV sker som en del av ett jobb. Dessa förändringar kan, precis som databastransaktioner, när som helst avbrytas eller genomföras genom att jobbet makuleras respektive exekveras.

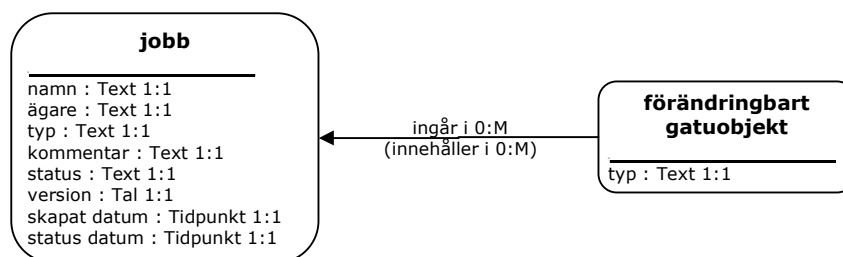


Fig 3.4: Begreppsgraf - *jobb*.

Fördelarna med att låta förändringar jobbanteras är bland annat följande:

- varje förändring kan avbrytas eller återkallas när som helst,
- förändringars validitet kan kontrolleras på ett säkert sätt vid incheckning av ett jobb utan att krav ställs på att alla förändringar redan från början är giltiga,
- externa applikationer kan tillåtas redigera informationen i systemet direkt i databasen genom att den kan valideras samlat vid exekveringen av ett jobb,
- överföring av data till Vägverket kan ske på ett strukturerat sätt där varje jobb motsvarar en transaktion mot NVDB och
- låsning och transaktionshantering kan skötas på ett strukturerat sätt.

Ett jobb har ett antal administrativa attribut, som kan användas för att knyta jobbet till ett ärendesystem, samt datumstämplar och status.

Den status som jobbet har beskriver var i ärendekedjan, se fig 4.5, som jobbet befinner sig.

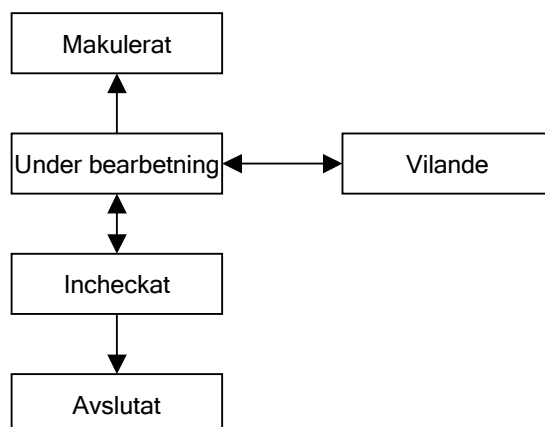


Fig 3.5: Status för jobb och deras inbördes förhållande.

Pilarna i figuren anger vilka möjliga övergångar det finns mellan de olika status ett jobb kan ha. I tabell 4.3 nedan ges sedan en kortfattad beskrivning av dem.

Status	Beskrivning
Under bearbetning	Ett jobb under bearbetning är öppet för modifiering.
Vilande	Ett vilande jobb kan inte modifieras men kan aktiveras, det vill säga sättas till under bearbetning, igen.
Makulerat	När ett jobb makuleras släpps alla de lås jobbet utverkar samt de förändringar som är gjorda i jobbet utan påverkan på informationen i databasen. Ett makulerat jobb kan inte aktiveras igen.
Incheckat	Ett incheckat jobb väntar på att bli avslutat. De förändringar som är gjorda har ännu inte definierats som aktuella utan är satta till föreslagna. Ett incheckat jobb kan sättas till under bearbetning igen om ingen intressent ¹ har tagit del av informationen och måste sättas om ifall någon intressent som tagit del av informationen har underkänt den.
Avslutat	Ett jobb blir avslutat när samtliga intressenter som måste godkänna förändringarna har godkänt dem. När ett jobb är avslutat kan det inte återgå till att vara under bearbetning.

Tabell 3.2: Beskrivning av de status ett jobb kan ha i TEKIS-LV.

3.4.1. Lås/konflikthantering

I ett flerskiktat system där data hanteras bortkopplat från databasen är konflikthantering kritiskt för systemets stabilitet och korrekthet. Problemet med konflikter kan hanteras endera optimistiskt eller pessimistiskt. Hanteras det optimistiskt utgår systemet från att

¹. Begreppet intressent beskrivs i avsnittet *Aktörer*.

sannolikheten att flera användare modifierar samma information är så liten att om det skulle ske så vägras den användare som sist försöker ändra informationen att göra så. Om problemet istället hanteras pessimistiskt så låser systemet all information som lästs upp av en användare för andra användare. På så sätt uppkommer aldrig problemet med konflikter.

I TEKIS-LV används både optimistisk och pessimistisk konflikthantering. Standardfallet är att en redigering sker med en optimistisk konflikthantering. Vid behov kan även en pessimistisk låsning av förändringsbara objekt göras. En sådan förändring sker alltid inom ramen för ett jobb. En redigering av gatunätet sker alltid genom ett jobb och är därmed alltid pessimistiskt konflikthanterad.

3.5. Aktör

Alla kopplingar till TEKIS-LV sker i form av aktörer vilket är ett begrepp som har följande definition:

Definition 7: En *aktör* är en extern betraktare eller påverkare av ett system.

I TEKIS-LV finns det två typer av aktörer, nämligen användare och intressenter. De båda typerna beskrivs under respektive rubrik nedan.

3.5.1. Användare och behörigheter

Alla förändringar i TEKIS-LV utförs av en användare. På grund av att systemets främsta uppgift är att fungera som kärna till andra system så får begreppet användare en något vidare innebörd i TEKIS-LV än i många andra system, nämligen:

Definition 8: En *användare* av ett system är en aktör som nyttjar och förändrar informationen i systemet.

Vad en användare tillåts förändra i TEKIS-LV definieras av vilka behörigheter användaren är tilldelad. En behörighet har följande definition:

Definition 9: En *behörighet* är ett avtal mellan ett system och en användare där användaren ges tillåtelse att tillgå eller förändra information av en typ som avtalet specificerar.

Behörighet utgörs i TEKIS-LV av två följande mekanismer:

- behörighet för en användare vid åtkomst av information
- behörighet för en användare vid åtkomst av en *funktion* i en *applikation*. Dessa båda beskrivs i avsnitt 3.5.2.

Typen av information som en behörighet ger en användare tillåtelse att tillgå eller förändra definieras av vilket objekt det gäller och om det handlar om läsningar eller förändringar. Om det rör sig om förändringar så är det typen av förändring som definierar behörigheten. Möjliga förändringstyper är tillägg, uppdatering och borttag.

3.5.2. Funktion och applikation

Behörighet kan begränsas för en användare vid åtkomst av en funktion i en applikation. Dessa definieras som:

Definition 10: En *funktion med behörighet* är ett användningsfall eller del av ett användningsfall som kan initieras från en applikation och som man vill kunna behörighetsstyra.

Definition 11: En *applikation* är en samling användningsfall som kan paketeras till en och samma programkomponent.

I TEKIS-LV finns funktionalitet för att registrera funktioner till en TEKIS-LV-applikation, kunna behörighetsadministrera dessa för varje användare samt kontrollera funktionsbehörigheten vid framtagningen av användarfall i applikationen. Se vidare beskrivning i *TEKIS-LV – Administratören – Handbok* [6].

3.5.3. Intressenter

Ett av TEKIS-LV:s främsta uppgifter är att fungera som en kärna för system med vägnätsanknuten information. Det innebär att förändringar påförda i TEKIS-LV av ett system kan vara av stort intresse för ett annat system. Begreppen intressent har följande definitioner i TEKIS-LV:

Definition 12: En *intressent* till ett system är en aktör som har ett intresse av att meddelas alla förändringar av vissa fördefinierade typer i systemet.

En viktig intressent till TEKIS-LV är den nationella vägdatabasen, NVDB.

3.6. Trafiknät

TEKIS-LV beskriver trafiknätets uppbyggnad på identiskt samma sätt som i SS 637004-1 [1]. Trafiknätet byggs upp av de två olika topologiska entiteterna, *gatunoder* och *gatulänkar* (motsvarande termer i SS 637004-1 är *nätförbindelse* och *nätknut*).

I nedanstående begreppsgraf, **fig 3.6**, definieras följande begrepp:

En **nätkomponent**, som är superklassen till gatunod och gatulänk, kan ha *en* geometrisk representation, denna term definieras i SS 637004-1 [1].

En gatulänk förbinder normalt två gatunoder (vändslinga är ett specialfall där gatulänken börjar och slutar i samma gatunod).

En **gatunod** knyter ihop två eller flera gatulänkar, eller utgör ändpunkt för en gatulänk.

Data kan knytas till en nätkomponent genom *nätanknuten företeelse*, se 3.7. En eller flera nätkomponenter kan representeras av en eller flera andra nätkomponenter i en annan detaljeringsnivå, se 3.10.

För en djupare diskussion om trafiknät hänvisas till SS 637004-1 [1].

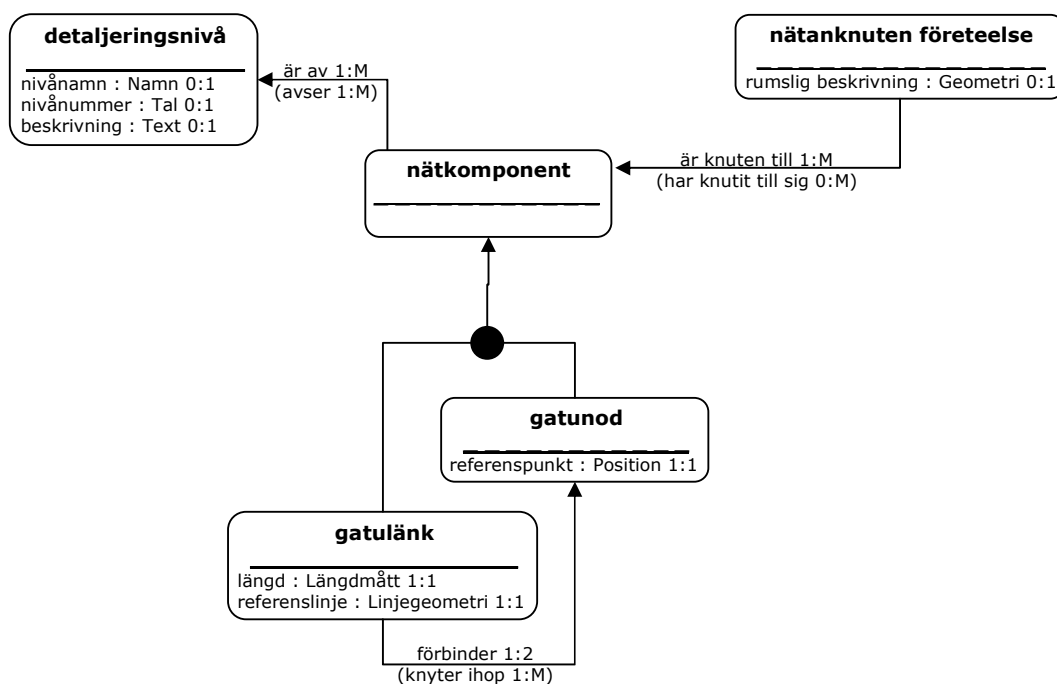


Fig 3.6: Begreppsgraf, trafiknät

3.6.1. Verksamhetsnät och LIVO

I TEKIS-LV finns två olika typer av tillämpningar på trafiknät, *grundgatunätet* och *verksamhetsnät*. Dessa definieras här under.

Definition 13: *Grundgatunätet* är den minsta indelning av trafiknät som en organisation vill kunna referera till.

I TEKIS-LV är det grundgatunätet som redigeras geometriskt och topologiskt med ett nätredigeringsverktyg, se *TEKIS-LV – Nätredigering – Användningsfall* [7].

Definition 14: Ett *verksamhetsnät* är en möjlig indelning av *grundgatunätet* som är gjord för en verksamhets behov.

Ett verksamhetsnät kan förekomma för varje *verksamhet* som vill kunna referera till gatunätet. Ett verksamhetsnät följer exakt grundgatunätets geometri. Topologin för ett verksamhetsnät kan dock skilja från grundgatunätets. Se **fig 3.7**. Ett verksamhetsnät är uppbyggt av speciella *gatulänkar*. Dessa benämns linjära verksamhetsobjekt (LIVO).

Definition 15: Ett *linjärt verksamhetsobjekt (LIVO)* är en namngiven linjärt sammanhängande mängd gatulänkar ur grundgatunätet.

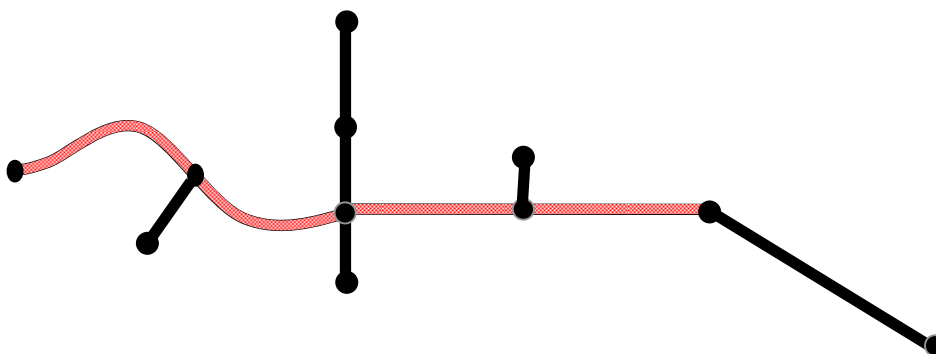


Fig 3.7: Rödmarkering i figuren representerar **ett** linjärt verksamhetsobjekt (LIVO) som sammanfogats av flera underliggande gatulänkar ur grundgatunätet.

För att identifiera sammanhängande delar av gatunätet oberoende av underliggande indelning av gatulänkar, används en *verksamhetsindelning*, där varje LIVO har (kan ha) ett populärnamn.

Definition 16: *Populärnamn* är det utåt identifierbara namnet på ett LIVO.

LIVO administreras av varje enskild verksamhet som vill använda en annan indelning av gatunätet än grundgatunätet. Ett LIVO kan i alla avseenden betraktas som en gatulänk, utbredningar på LIVO:n presenteras och redigeras längs dess egna linjära referenssystem. Utbredningar i en verksamhetsindelning lagras dock i det underliggande grundgatunätets referenssystem för att kunna presenteras och användas i andra verksamheter med andra verksamhetsindelningar.

LIVO är infört för att ge användare i en viss verksamhet ett begrepp att använda och identifiera, samt att referera utbredningar till. Ett exempel på detta kan vara cykelnätet, där man vill hantera alla "cyklingsbara" gatulänkar samt de noder som finns i detta nät för vägval. Man kan även utnyttja LIVO för att i ett visst verksamhetsnät "gömma ointressant geometri eller topologi" samt anslutningar till dessa. Ovanstående är exempel där verksamhetsnätet kan abstraheras till ett traditionellt nod-länk-nät och kan användas i GIS-system med full funktionalitet. Ytterligare andra exempel som inte ger nod-länk-nät är ett verksamhetsnät med "hela gator" (alla underliggande gatulänkar som har utbredningar av samma gatunamnsföreteelse), ett körstråk (sträckan Norrtull till Sergels Torg) eller en busslinje.

En vanlig indelning av LIVO är med avseende på *gatudelar*. Normalt finns i olika verksamheter inom en organisation en indelning av *gator* som baseras på en regel, t ex ”bilgata mellan korsning a och b”, där man med *korsning* menar korsning med andra *bilgator*. Med hjälp av LIVO kan alla gatulänkar som uppfyller denna regel (och som kanske gäller endast för denna verksamhet) sättas samman till ett, för användarna, externt identifierbart begrepp.

Ett LIVO motsvarar precis en gatulänk i det nät som är associerat till en viss verksamhet.

3.7. Nätanknuten företeelse

TEKIS-LV beskriver knytningen av information till trafikinätet på identiskt samma sätt som i SS 637004-1 [1] och i SS 637006 [2]. Informationsknytningen görs genom *nätanknutna företeelser* som är en subclassning av *företeelse* (begreppet finns i SS 637006 [2]).

I nedanstående begreppsgraf, **fig 3.8**, definieras följande begrepp:

En **nätanknuten företeelse** kan ha *en* geometrisk representation som beskriver läget för den företeelsen relativt trafikinätet. Begreppet finns beskrivet i SS 637004-1 [1].

En **företeelse**, som är superklassen för *nätanknuten företeelse* kan innehålla ett eller flera attribut. Begreppet finns beskrivet i SS 637006 [2].

Ett **attribut** har ett namn, ett värde och en datatyp. Giltiga värden för attributet hämtas från attributets domän.

En **domän** beskriver ett attributs giltiga värden. En domän kan vara en lista av diskreta värden eller en eller flera regler som beskriver giltiga värden (t ex >10 , <0 och >5).

För en vidare diskussion om nätanknutna företeelser hänvisas till SS 637004-1 [1] och SS 637006 [2].

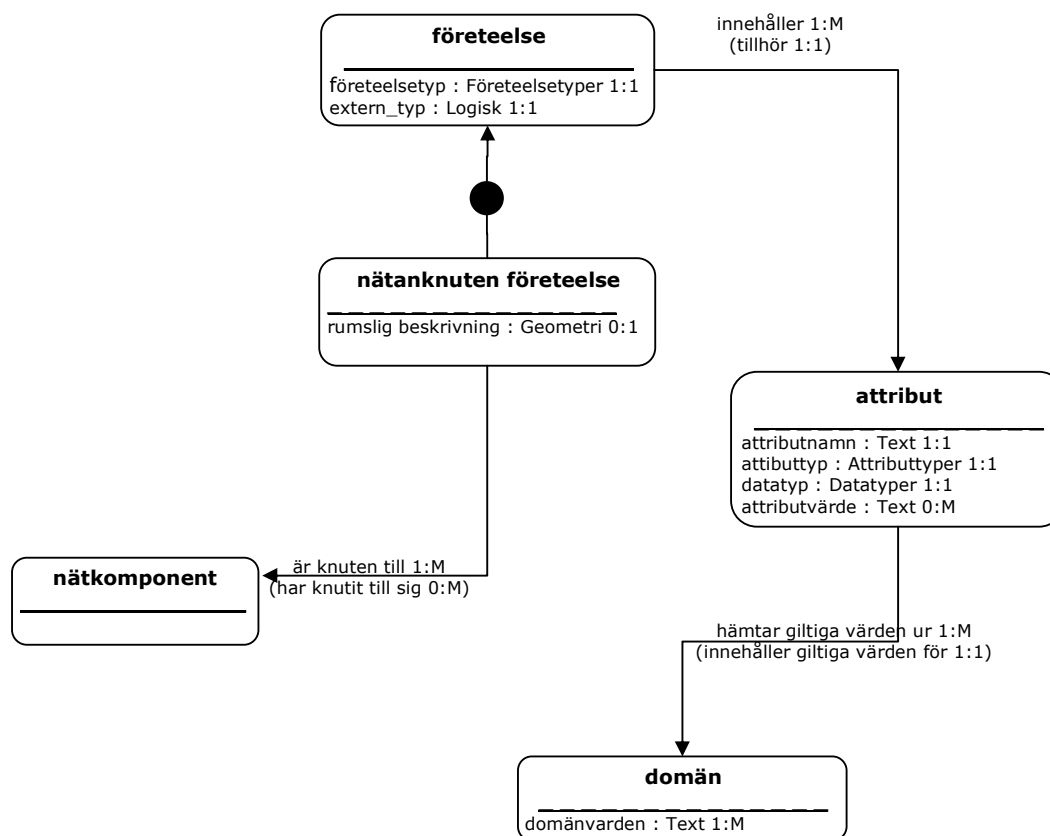


Fig 3.8: Begreppsgraf, nätanknuten företeelse

För en företeelsetyp kan man definiera ett av attributen i mängden av definierade attribut som ett *huvudattribut*. Huvudattributet är det attribut som betraktas som ”lite viktigare” än alla andra attribut och det finns i TEKIS-LV teknik för att presentera huvudattributet i ett GIS-verktyg eller på ett enkelt sätt, i rapportform.

3.8. Verksamhet

Definition 17: En *verksamhet* är en organisation eller delorganisation som har samma eller likartade behov vad gäller information samt användning av gatunätets referenssystem.

Typiskt representeras en verksamhet i TEKIS-LV av dess *verksamhetsapplikation* (som till exempel TEKIS-GATA eller TEKIS-LTF) men en verksamhet kan även användas för flera verksamhetsapplikationer inom samma verksamhet.

I TEKIS-LV kan en verksamhet äga ett antal nätanknutna företeelser och omvänt ägs en nätanknuten företeelse av en och endast en verksamhet. En verksamhet kan vidare äga en egen verksamhetsindelning av gatunätet. Verksamhet är liksom betraktelsesdatum en systeminställning för en TEKIS-LV-applikation.

När en verksamhet betraktar gatunätet och den information som finns knuten därtill, betraktas de företeelser som ägs av verksamheten själv som *egna företeelsetyper* och företeelser som är ägda av någon annan verksamhet som *främmande företeelsetyper*. En verksamhet kan för varje egen företeelsetyp välja att publicera företeelsetypen. Den kallas då för en *publik företeelsetyp*.

Till en *verksamhet* finns en eller flera *verksamhetsapplikationer*. Till en *verksamhetsapplikation* kan man tilldela behörighet för en *användare*.

3.9. Datakatalog

Datakatalogen beskriver den funktionalitet som finns i TEKIS-LV för att beskriva generella företeelser som ska kunna nätanknytas genom TEKIS-LV. I datakatalogen beskrivs metadata för företeelsetyper som TEKIS-LV hanterar med dess ingående attribut samt de regler som ska tillämpas vid nätanknytning av företeelsen. Datakatalogen *kan* men behöver ej användas för att lagra data för en företeelsetyp.

Definition 18: En *intern företeelsetyp* är en företeelsetyp vars företeelser lagras information i TEKIS-LV:s datakatalog.

Definition 19: En *extern företeelsetyp* är en företeelsetyp vars företeelser lagras information utanför TEKIS-LV:s datakatalog.

Det externa systemet kan vara ett befintligt databssystem. För externa företeelser lagras endast den geometriska representationen samt den främmande nyckeln till den externa företeelsen i det externa systemet. Användning av externa företeelsetyper beskrivs noggrannare i *TEKIS-LV – Systembeskrivning* [8].

För interna företeelsetyper kan även enskilda attribut lagras externt. Sådana attribut kallas externa attribut.

Definition 20: Ett *externt attribut* är ett attribut för vilket värden lagras utanför TEKIS-LV:s datakatalog.

Observera att företeelser av intern och extern företeelsetyp knyter sig till den gemensamma beskrivningen av vägnätet på samma sätt.

Observera vidare att lagringen av interna företeelsetyper alltid görs i respektive verksamhets verksamhetsschema. Detta beskrivs uttömmande i kapitel *Isolering* i dokumentet *TEKIS-LV – Systembeskrivning* [8].

En företeelsetyps attribut kan vara *publikt* eller *privat*. En företeelsetyp som beskrivs som privat kan inte visas i någon annan verksamhet än i den som äger den. Omvänt kan företeelsetyper som är publika visas i andra verksamheter än den som äger företeelsetypen. En

företeelsetyp kan vara publik eller privat oberoende av om den är av intern eller extern typ.

Definition 21: En *publik företeelsetyp* är en företeelsetyp som den ägande verksamheten publicerat och är åtkomlig för andra verksamheter.

Definition 22: En *privat företeelsetyp* är en företeelsetyp som enbart är åtkomlig för den den verksamhet som äger den.

Även enskilda attribut för en företeelsetyp kan vara publika eller privata. För en publik företeelsetyp är således **alla** attribut publika och för en privat företeelsetyp är **alla** attribut är privata.

Datakatalogen beskrivs nedan i **fig 3.9**.

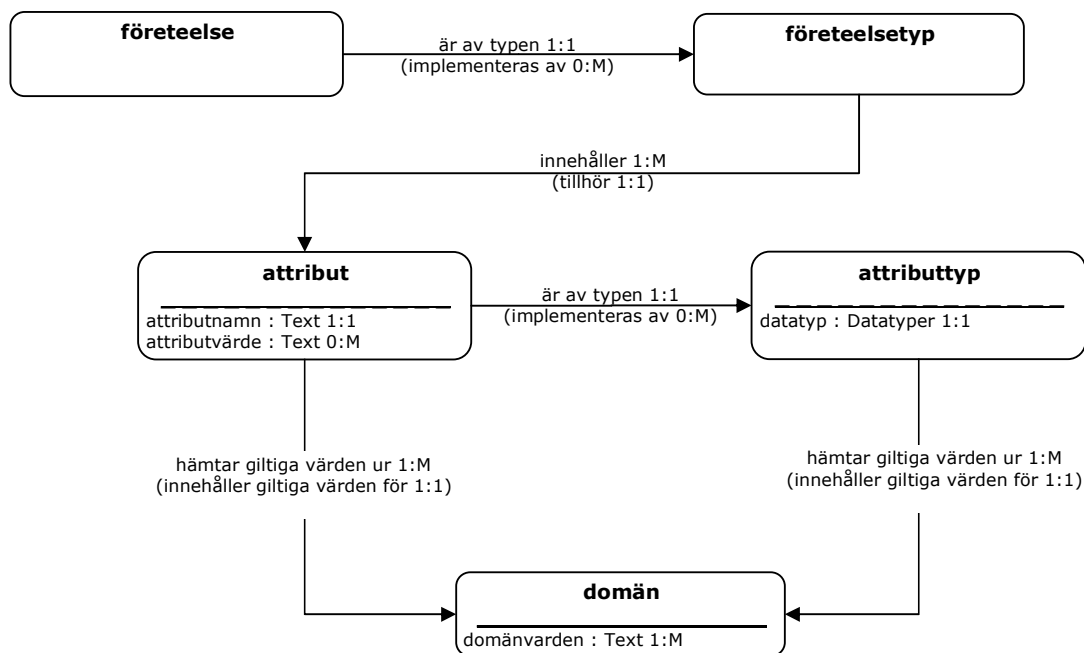


Fig 3.9: Begreppsgraf, datakatalog

3.10. Detaljeringsnivåer

Detaljeringsnivåer är ett begrepp som syftar till att möjliggöra olika grader av detaljrikedom vid presentationer och analyser av data. Enklast åskådliggörs dess nytta genom ett exempel. För en speditiionsfirma som ska transportera en last mellan Malmö och Umeå och för en taxichaufför som har en körning mellan två adresser i Stockholm så är behovet

av detaljer väldigt olika. Medan det för speditorsfirman är i stort sett tillräckligt att veta hur Europavägarna utbreder sig i Sverige behöver förmodligen taxichauffören information om även de minsta gatorna.

Detaljeringsnivåer ingår som en del i den informationsmodell som är framtagen för den nationella vägdatabasen NVDB (se *NVDB – Översiktlig informationsmodell* [9]) även om endast detaljeringsnivån *vägbana* för närvarande används i NVDB. Sättet på vilket de är modellerade i informationsmodellen påminner mycket om begreppen Thevenin- och Nortonkretsar i elkretsläran där en del av nätverket ersätts av en mindre komplex beskrivning av nätverksdelen men vilken har samma gränssnitt utåt till resterande nätverket. Denna abstraktion möjliggörs genom det användande av portar som kopplingselement mellan topologiska primitiv beskrivs i avsnittet 0, *Topologi*, tillsammans med en trädstrukturerad relationsbeskrivning mellan topologiska primitiv i olika detaljeringsnivåer.

Det finns en hel del användningsområden för detaljeringsnivåer och några av dessa listas nedan:

- Presentation av vägnätet kan göras mer tydlig genom att man vid olika inzoomningsgrader kan presentera adekvat detaljerad information.
- Detaljeringsnivåerna möjliggör en beskrivning av en väg från en högre topologisk nivå ner till en mer detaljrik körbane- och kantstensnivå.
- Med hjälp av detaljeringsnivåer går det att optimera nätverksanalyseringsalgoritmer, såsom kortaste vägen och handelsresandeproblemet.

Det enskilt största problemet med detaljeringsnivåer är underhållet av kopplingarna mellan nätelement i en högre nivå med dito i en lägre nivå. Det finns ingen funktionalitet för det i de geografiska informationssystem som finns på marknaden. För Vägverkets klientapplikation för redigering av vägnät och företeelser, Slussen, pågår ett projekt vilket syftar till att ta fram funktionalitet för hantering av detaljeringsnivåer men det återstår en del innan det kan bli klart. Bland annat har man helt bortsett från företeelser i detta angreppssätt. Det innebär även problem att på ett tydligt sätt åskådliggöra förhållandet mellan nätelement i olika nivåer. En ytterligare svårighet uppkommer i samband med geometriredigeringar. Då mittlinjen för en väg redigeras är det sannolikt önskvärt att även geometrin för lägre detaljeringsnivåer, såsom körfält och kantstenar, uppdateras. Det är dock inte helt trivialt hur förhållandet mellan geometrier i olika nivåer ser ut.

Trots att detaljeringsnivåer inte används för närvarande i NVDB finns det ändå en mening i att använda sig av den tänkta modell som togs fram för detaljeringsnivåer i NVDB:s informationsmodell. Dels så underlättar det vid ett möjligt framtida införande av detaljeringsnivåer i NVDB och dels så ligger det en hel del tankearbete bakom modellen vilket man bör dra nytta av.

Informationsmodellens modell av detaljeringsnivåer är relativ enkel. Den består i att man för varje nätelement kan definiera en koppling upp till ett nätelement på en högre nivå och ner till en mängd nätelement på en lägre nivå. Likadant fungerar det med kopplingselementen portar. Varje nätelement har sedan ett attribut vilket specificerar detaljeringsnivån det tillhör. Portarnas detaljeringsnivå, å andra sidan, beror på vilket topologiskt primitiv de är kopplade till.



I TEKIS-LV finns ingen funktionalitet implementerad för att hantera detaljeringsnivåer men ett framtida eller externt införande är möjliggjort genom att databasmodellen förbättrats för det.

4. Tillämpningsmodell

Detta avsnitt beskriver tillämpningsmodellen för TEKIS-LV. Tillämpningsmodellen är uttryckt i UML och är oberoende av teknisk implementation och lagringsmodell. En fullständig tillämpningsmodell inklusive attribut och arv finns även i bilaga A.

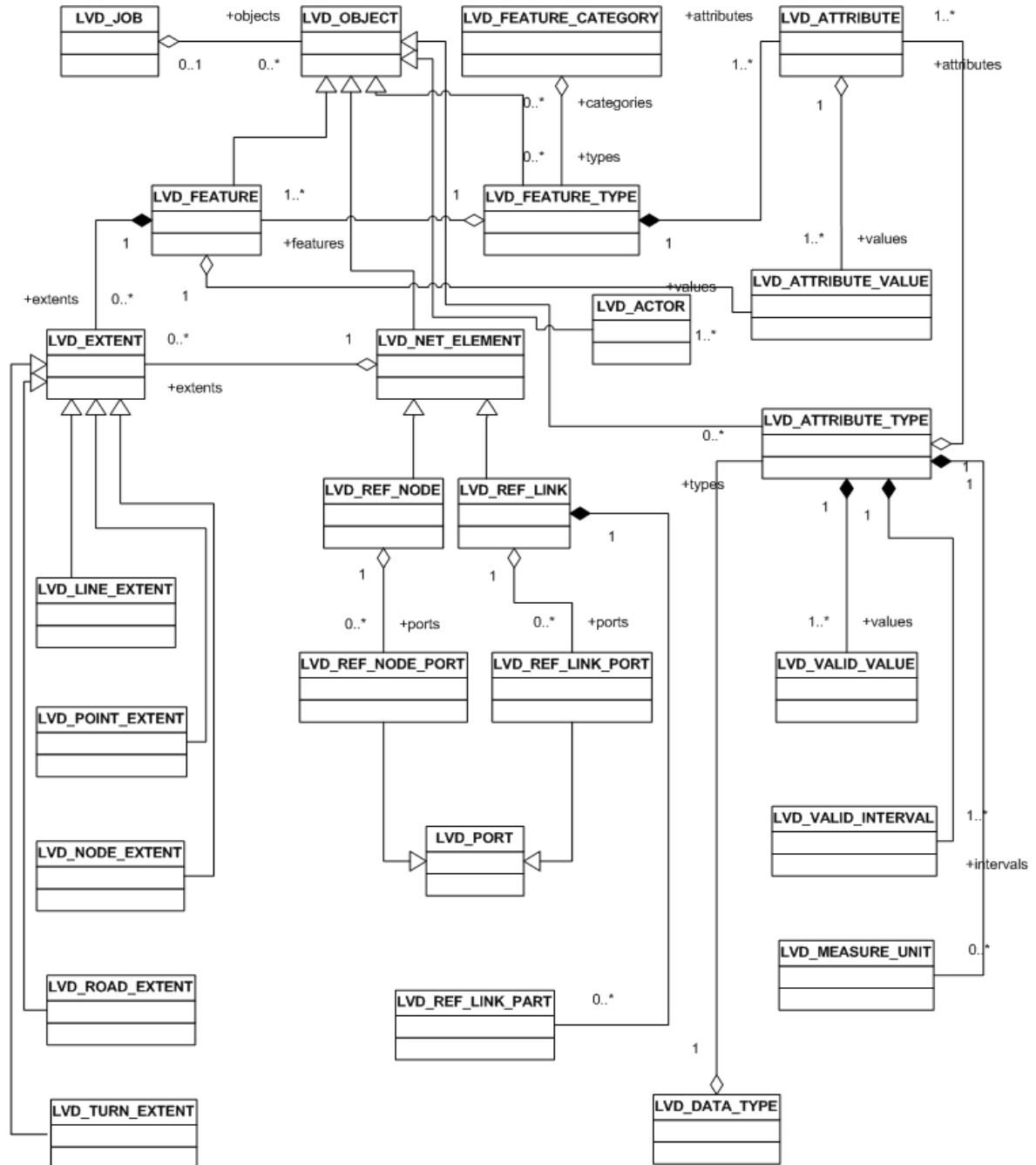


Fig 4.1: TEKIS-LV tillämpningsmodell, fullständig graf, utan attribut och arv

Syftet med tillämpningsmodellen är att beskriva TEKIS-LV:s tekniska uppbyggnad utan att ta speciella hänsyn till teknikval. Detta gäller speciellt teknikval för databashanterare, teknik för GIS-integration eller utvecklingsarkitektur och -miljö. Ambitionen med TEKIS-LV har under hela projektet varit att ta fram en modell som är oberoende av dessa och liknande teknikval.

4.1. Tillämpningsmodell, jämförande tabell

Tabell 5.1 beskriver alla tillämpningsklasser som finns i TEKIS-LV samt, i förekommande fall, en referens till klasser från standarddokumenten SS 637004-1 [1], SS 637006 [2] och SS 637007 [3].

Klass	Ref	Ärver från SIS-klass	Motsvarar begrepp i begreppsmodellen
lvd_actor	4.4	Finns ej	Aktör
lvd_attribute		SS 637006: GFI_AttributeInstance	Attribut
lvd_attribute_type		SS 637006: GFC_AttributeType	Attributet typ på attribut
lvd_attribute_value		SS 637006: GFI_AttributeValue	Attributet värde på attribut
lvd_data_type		ISO/TS 19103: TypeName	Attributet datatyp på attribut
lvd_extent	4.9	SS 637004-1: NW_Extent	Attributet utbredning på nätanknuten företeelse
lvd_feature	5.7.1	SS 637006: GFI_FeatureInstance	Nätanknuten företeelse
lvd_feature_type		SS 637006: GFC_FeatureType	Attributet typ på nätanknuten företeelse
lvd_line_extent		SS 637004-1: NW_LineExtent	[saknas]
lvd_job	4.3	SS 637007: GUM_ChangeTransaction	Jobb
lvd_measure_unit		ISO/TS 19103: UnitOfMeasure	[saknas]
lvd_net_element	-	SS 637004-1: NW_NetElement	[saknas]
lvd_node_extent		SS 637004-1: NW_NodeExtent	[saknas]

lvd_object	4.2	SS 637007: GUM_HistoryObject	Förändringsbart objekt
lvd_point_extent		SS 637004-1: NW_PointExtent	[saknas]
lvd_port		SS 637004-1: NW_Port	[saknas]
lvd_ref_node		SS 637004-1: NW_RefNode	[saknas]
lvd_ref_node_port	5.5.4	SS 637004-1: NW_RefNodePort	[saknas]
lvd_ref_link		SS 637004-1: NW_RefLink	[saknas]
lvd_ref_link_part		SS 637004-1: NW_RefLinkPart	[saknas]
lvd_ref_link_port		SS 637004-1: NW_RefLinkPort	[saknas]
lvd_road_extent		SS 637004-1: NW_RoadExtent	[saknas]
lvd_turn_extent		SS 637004-1: NW_TurnExtent	[saknas]
lvd_valid_interval		SS 637006: GFC_ValidRange	[saknas]
lvd_valid_value		SS 637006: GFC_ValidValue	[saknas]

Tabell 4.2: Tillämpningstabell

4.2. Förändringsbart objekt

Förändringsbart objekt motsvaras i tillämpningsmodellen av *lvd_object*. Klassen *lvd_object* representerar ett nätelement (*lvd_net_element*), nätanknoten företeelse (*lvd_feature*), företeelsetyp (*lvd_feature_type*), attributtyp (*lvd_attribute_type*) eller aktör (*lvd_actor*). Förändringsbara objekt har den gemensamma egenskapen att kunna versionshanteras (tids- och redigeringshistoriskt), behörighetshanteras och låsas för att ingå i jobb, (se avsnitt 3.2 för en diskussion om *förändringsbara objekt*).

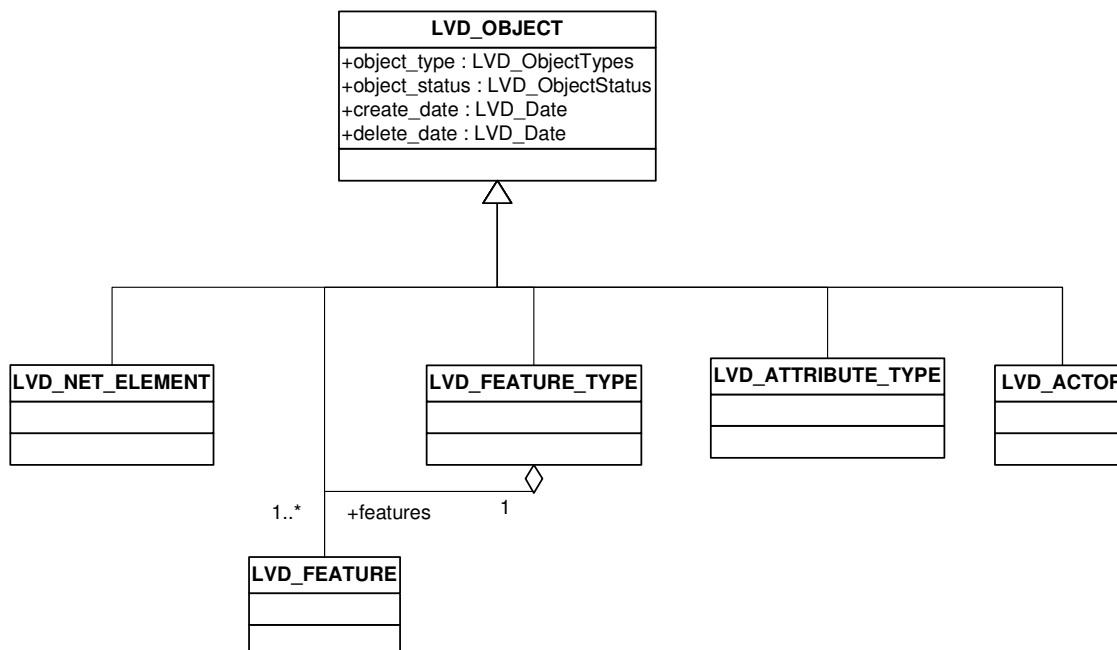


Fig 4.3: Klassen *lvd_object* med subklasser.

4.2.1. Objekt

object_type (attribut)

Typ av subclassat objekt.

datatyp: *lvd_object_type* {*NetElement, Feature, FeatureType, AttributeType, Actor*}

object_status (attribut)

Objektstatus för ett förändringsbart objekt.

datatyp: *lvd_object_status* {*Historisk, Gällande, Under bearbetning, Föreslagen*}

create_date (attribut)

Skapat datum.

datatyp: *lvd_time_stamp*

delete_date (attribut)

Datum för borttag. Ett objekt tas aldrig explicit bort, det ”deletemarkeras” endast.

datatyp: *lvd_time_stamp*

4.2.2. Redigeringsversioner

En version av ett objekt benämns i TEKIS-LV som en objektversion. Motsvarande benämning för ett objekts giltighetstid är tidsversion. En objektversion motsvarar entiteten *GUM_HistoryObject* i SS 637007 [3]. I **fig 4.4** nedan presenteras entiteten objektversion tillsammans med dess samband med objekt och version.

Som beskrivets tidigare lagras förändringshistorik för alla typer av förändringsbara objekt i TEKIS-LV vilket innebär att det finns en motsvarande objektversionsentitet för varje objektentitet. I begreppsmodellen och i tillämpningsmodellen göms denna funktionalitet, se vidare dokumenten *TEKIS-LV – Lagringsmodell – Diagram* [10] och *TEKIS-LV – Lagringsmodell – Referens* [11] för en beskrivning av hur objektversioner och objektentiteter modelleras i databasen.

4.2.3. Utförd förändring

En förändring kan utföras och benämns när den är genomförd för en *utförd förändring*. En utförd förändring motsvarar begreppet *GUM_Change* i SS 637007 [3] och nedan presenteras begreppet i fig 5.3.

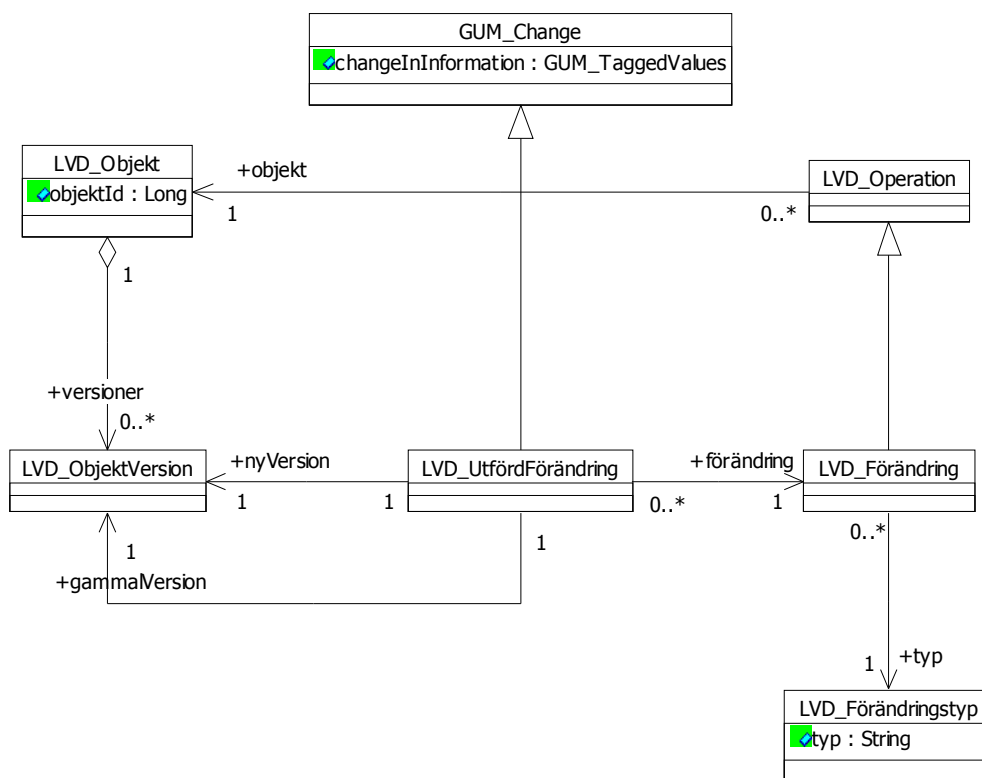


Fig 4.4: Modell av begreppet utförd förändring i TEKIS-LV.

4.3. Jobbhantering

För att knyta en serie förändringar till en förändringsprocess som behandlas likformigt införs begreppet *jobb*, se avsnitt 3.4. En sådan förändringsprocess kan t ex vara ett ärende.

I fig 4.4 framgår det att jobbet har en identitet vilken unikt identifierar jobbet. Dessutom har ett jobb fyra administrativa attribut nämligen namn, diarienummer, ägare och kommentar. Utöver det finns det även två datumattribut. Ett som anger datumet då jobbet skapades och ett som anger datumet då jobbet senast ändrade status.

De domänklasser, *lvd_job_type* och *lvd_job_status*, som finns med i figuren kopplar en numerisk representation av ett jobs typ respektive status med en textuell beskrivning av dito.

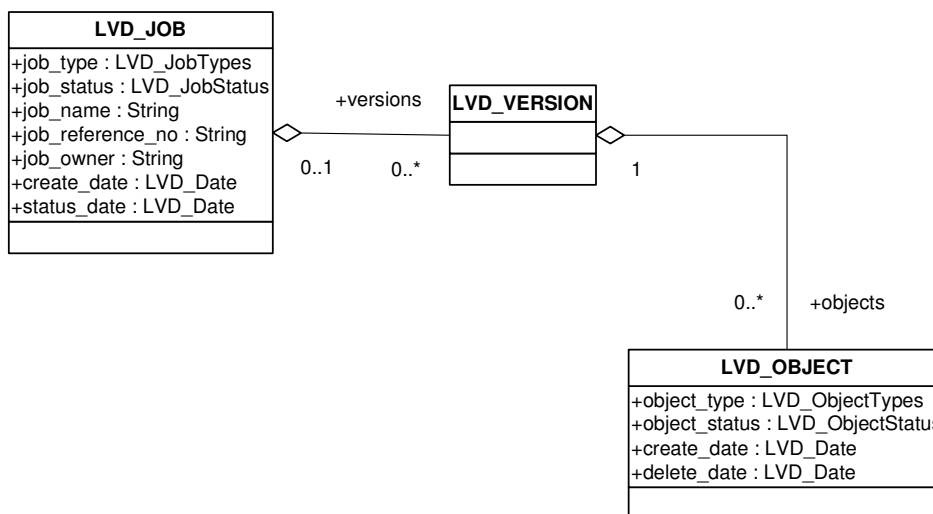


Fig 4.5: Tillämpningsmodell, jobb

4.3.1. Jobb

Ett jobb kan innehålla en eller flera redigeringsversioner. Jobbet fungerar då som ett sammanhållande begrepp för alla förändringar inom dessa redigeringsversioner. Redigeringsversionerna kan antas eller förkastas i sin helhet.

job_type (attribut)

Typ av jobb. Typ kan användas för att beskriva vilket system som föranlett jobbet. Exempel på jobbtyper skulle då kunna vara NVDB, Galant, LTF, et cetera. Attributet skulle även kunna användas för att beskriva typer såsom projektering eller planering.

datatyp: *lvd_job_type*

job_status (attribut)

Jobbstatus. Den status som jobbet har beskriver var i händelsekedjan som jobbet befinner sig.

datatyp: *lvd_job_status* { *Under bearbetning, Vilande, Makulerat, Incheckat, Avslutat* }

job_name (attribut)

Extern identifierare på jobbet.

datatyp: *String*

job_reference_no [0..1] (attribut)

Kan användas för att knyta ett jobb till ett befintligt ärendenummer.

datatyp: *String*

job_owner [0..1] (attribut)

Ägare av ett jobb.

datatyp: *String*

create_date (attribut)

Skapatdatum

datatyp: *lvd_time_stamp*

status_date (attribut)

Datomet då jobbet senast ändrade status

datatyp: *lvd_time_stamp*

4.4. Aktör

4.4.1. Session

Vid en autentisering av en användare i en verksamhetsapplikation skapas en *lvd_session*. Denna kommer sedan fortlöpande att skickas med för autentisering vid varje serveranrop. Detta objekt kan även en användargränssnittsprogrammerare kontrollera behörighet emot.

user_name (attribut)

Inloggad användare.

datatyp: *String*

create_date (attribut)

Datum och tid då sessionen skapades

datatyp: *Integer*

expire_date (attribut)

Datum och tid då sessionen förfaller

datatyp: *Integer*

scope_id (attribut)

Identitet på den verksamhet som används i sessionen.

datatyp: *Integer*

function_id (attribut)

Identitet på det referenssystem som används i sessionen.

datatyp: *Integer*

view_date(attribut)

Betraktelsedatum

datatyp: *Integer*

job_id (attribut)

Identitet på det jobb som är valt för sessionen.

datatyp: *Integer*

4.4.2. Applikation

Till en verksamhet kan det finnas en eller flera *verksamhetsapplikationer*.

4.5. Verksamhet

Aktuell verksamhet kan sättas för en session, se **Error! Reference source not found.**

4.6. Topologi

Det här avsnittet beskriver TEKIS-LV:s topologimodell. Ett av målen för modellen är, som tidigare nämnt, att vara kompatibel med de svenska standarder som finns på området. Därför har det fallit sig naturligt att nyttja topologimodellen i SS 637004-1 [1] vilken använder sig av ett så kallat portkoncept för att beskriva topologin i vägnätet. Portkonceptet skiljer sig en hel del från de nod-länk-system som är vanliga hos andra vägnätssystem (och topologiska system generellt) genom att det inte tar avstamp i grafteorin och dess definitioner för noder och länkar. Istället beskrivs topologin av nätelement och portar vilka saknar motsvarighet i grafteorin. Se **fig 4.6** för en schematisk modell av TEKIS-LV:s och portkonceptets topologimodell.

Anledningen till att portkonceptet nyttjas i vägnätsstandarderna är följande:

- Information som är knuten till ett vägnätssystem får inte förloras vid redigeringar av vägnätets topologi och stabiliteten hos de informationsbärande referenslänkarna i portkonceptet förhindrar på ett smidigt sätt förluster av information genom att de inte förändras vid topologiska förändringar av nätet. I ett nod-länkbaserat nät uppstår svårigheter vid sammanslagning och delning av länkar kring överförandet av den vägnätsanknutna informationen mellan de ursprungliga och de nyskapade länkarna. Överförandet måste hanteras genom logik vilken beräknar informationens nya knytning till vägnätet vilket är fullt möjligt vid intern information men kan vara näst intill omöjligt för extern information vilken systemet inte har någon kontroll över. Mer information om referenslänkar och stabiliteten hos dem återfinns i avsnitt 4.6.5.
- Vägnätet ska kunna beskrivas i olika detaljeringsnivåer, det vill säga, med olika nivåer av detaljrikedom och de topologiska kopplingselementen. Portarna, i portkonceptet förenklar denna hantering genom att nätelementen kan användas i flera detaljeringsnivåer och vara kopplade, genom en uppsättning portar, till olika nätelement i de olika nivåerna.

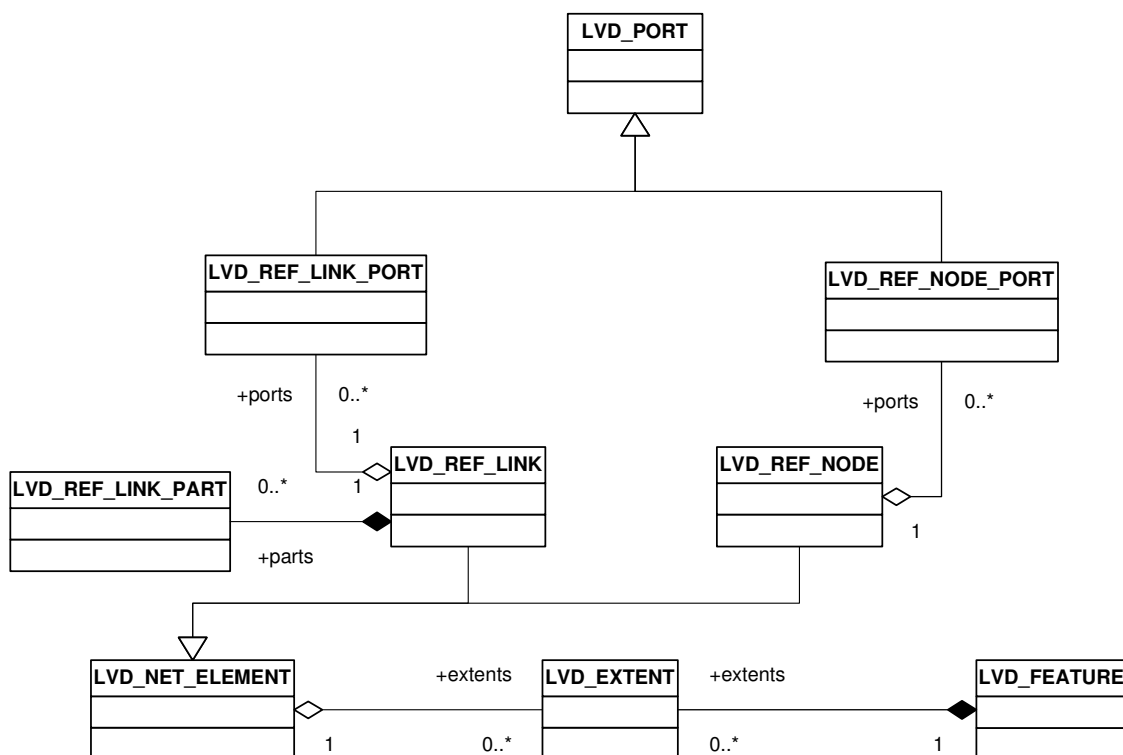


Fig 4.6: Schematisk modell av nättopologin i TEKIS-LV. Referenslänkar (*lvd_ref_link*) och referensnoder (*lvd_ref_node*) knyts samman av portpar (*lvd_ref_link_port* och *lvd_ref_node_port*). En vägnätsanknuten företeelse (*lvd_feature*) kan sedan knyts till vägnätet på ett generellt genom en utbredning (*lvd_extent*).

4.6.1. Nätelement

Nätelement kan ses som portkonceptets motsvarighet till topologiska primitiver. Som nämnts tidigare så beskriver portkonceptet dock inte topologin i ett nät på samma sätt som ett nod-länk-system vilket medför att nätelement och topologiska primitiv inte är helt och hållet synonyma begrepp. Den främsta skillnaden är att begreppet nätelement fungerar som en superklass till begreppet referenslänk vilket inte är en topologisk entitet medan det topologiska primitivet är superklass till alla topologiska entiteter i ett nod-länk-system. Mer information om referenslänkar återfinns i avsnitt 4.6.5.

Klassen *lvd_net_element* ärver från begreppet *NW_NetElement* i SS 637004-1 [1] och beskrivs i **fig 4.7** nedan.

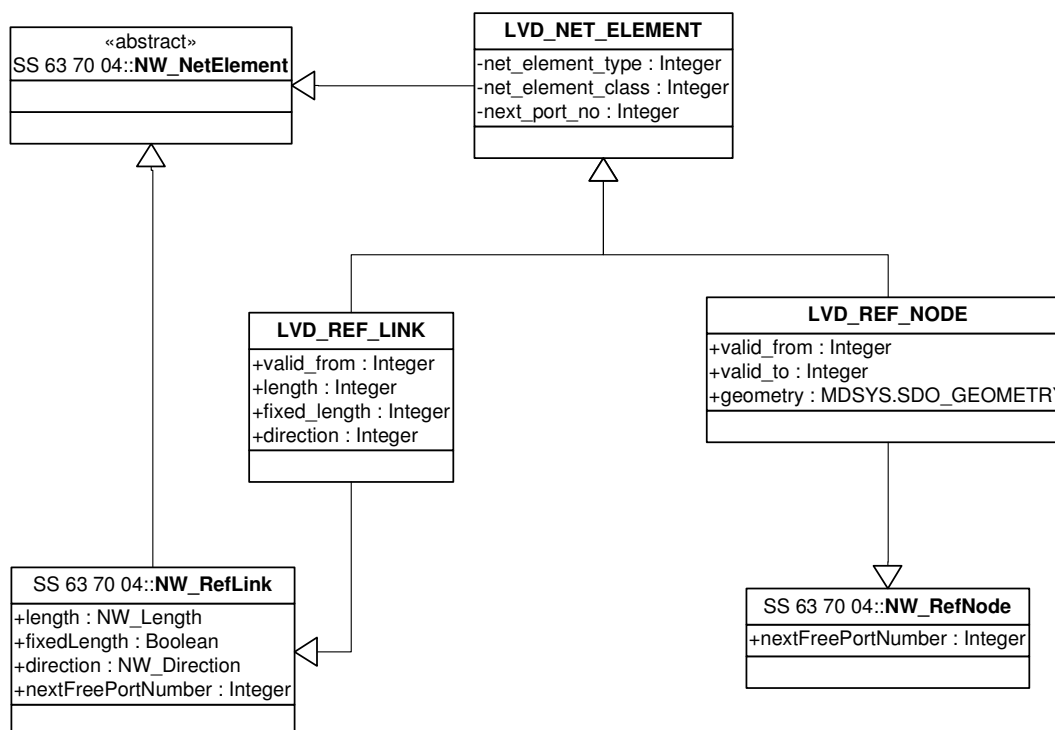


Fig 4.7: Modell av nätelement (*lvd_net_element*, *lvd_ref_link* och *lvd_ref_node*) i TEKIS-LV.

Nätelement ärver av *förändringsbart objekt* (*lvd_object*, se avsnitt 4.2) vilket medför att de kan läsas, behörighetshantera, och versions- och förändringshanteras.

net_element_type

Typ av subklassat nätelement. De typer av nätelement som finns är referenslänkar (se avsnitt 4.6.5) och referensnoder (se avsnitt 4.6.7).

datatyp: *lvd_net_element_type* {referenslänk, referensnod, färjeled}

net_element_class

Typ av subklassat nätelement.

datatyp: *lvd_net_element_class* {referenslänk, referensnod, färjeled}

next_port_no

Attributet nästa ports löpnummer är en räknare som håller reda på vilket löpnummer nästa port som knyts till nätelementet ska tilldelas.

datatyp: *Integer*

4.6.2. Port

Kopplingar mellan nätelement, det vill säga referensnoder och referenslänkar, sker i portkonceptet inte som i nod-länk-system explicit utan definieras implicit med hjälp av så kallade portar. Till ett nätelement finns en mängd portar knutna. Portarna, i sin tur, är knutna till andra portar parvis och det är dessa portpar som utgör den topologiska kopplingen i portkonceptet.

Begreppet port i TEKIS-LV ärver från begreppet *NW_Port* i SS 637004-1 [1] och beskrivs i **fig 4.8** nedan.

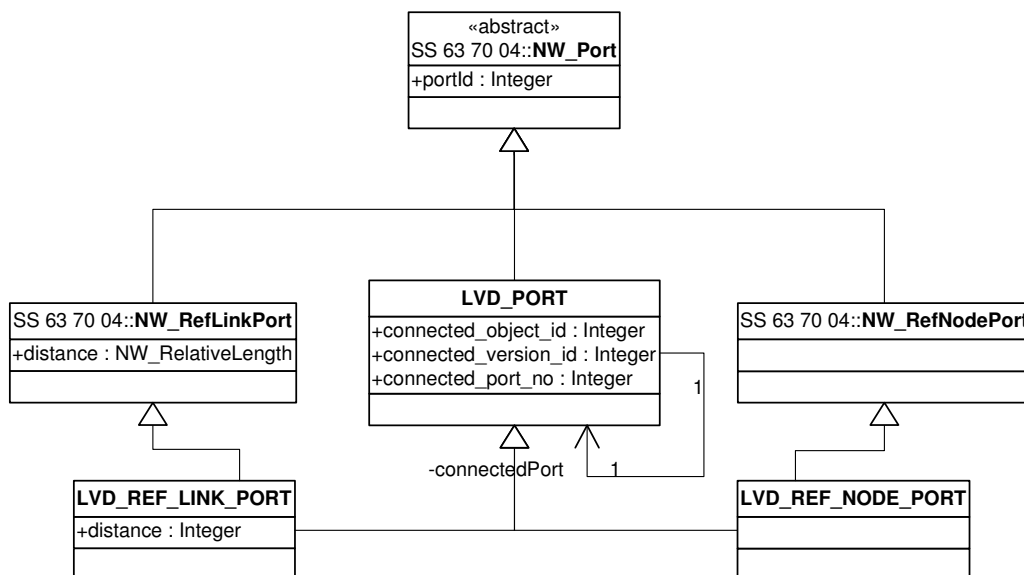


Fig 4.8: Modell av begreppet port i TEKIS-LV.

connected_object_id (associationsroll)

Associerar en port med en annan port.

connected_version_id (associationsroll)

Associerar en port med en annan port.

connected_port_no (associationsroll)

Associerar en port med en annan port.

Det finns två typer av portar: länkportar och nodportar. Länkportar är portar som är kopplade till referenslänkar. Länkportar skiljer sig från nodportar, vilka är portar som är kopplade till referensnoder, genom att de har ett mått definierat som anger portens position längs dess tillhörande referenslänk. Begreppen länkport och nodport i TEKIS-LV motsvarar begreppen *NW_RefLinkPort* respektive *NW_RefNodePort* i SS 637004-1 [1].

Teoretiskt sett tillåter portkonceptet kopplingar mellan två godtyckliga nätelement men reglerna begränsar kopplingstyperna till koppling mellan nätnod och referenslänk.

Kommentar: I NVDB är även kopplingar mellan två referenslänkar tillåtna t ex i specialfallet *kurvrätning*. TEKIS-LV använder inte denna funktionalitet eftersom man eftersträvar en enkel mappning mellan portkonceptets topologimodell och nod-länk-topologi.

4.6.3. Referenslänksport

Referenslänksport. Får kopplas i par med referensnodsportar.

distance (attribut)

Anger portens lägen räknat i avstånd från referenslänkens startpunkt.

datatyp: *Double*

4.6.4. Referensnodsport

Referensnodport. Får kopplas i par med referenslänksportar.

4.6.5. Referenslänk

De mest centrala nätelementen i TEKIS-LV är referenslänkarna vilka i teorin kan ses som lokala koordinatsystem i vilka informationers utsträckning längs vägnätet kan beskrivas och vilka i praktiken representerar vägar eller vägavsnitt. Den viktigaste egenskapen hos en referenslänk är att den är stabil, vilket innebär att den dels aldrig kan försvinna eller tas bort och dels att den aldrig förändras. Det medför att det finns ett stabilt element till vilken man kan knyta information på ett säkert sätt. Med säkert menas här att det inte finns någon risk att den information man knutit till referenslänken försvinner oavsett vilken typ av förändring som utförs på topologin.

Begreppet referenslänk implementeras av *lvd_ref_link* och ärver från begreppet *NW_Ref-Link* i SS 637004-1 [1]. Referenslänkar beskrivs i **fig 4.9**.

valid_from (attribut)

anger det datum från och med vilket referenslänken är giltig. Notera att referenslänken inte har något attribut som anger det datum till vilket referenslänken är giltig beroende på det stabilitetskrav som finns för referenslänken.

datatyp: *Integer*

length (attribut)

anger den uppmätta eller överenskomna längden för länken

datatyp: *Double*

fixed_length (attribut)

anger huruvida det längdvärdet ska ligga fast även om den geometriska representationen av referenslänken uppdateras

datatyp: *Integer*

direction (attribut)

anger huruvida referenslänkens riktning följer längs med geometrins riktning eller om den går emot geometrins riktning

datatyp: *Integer*

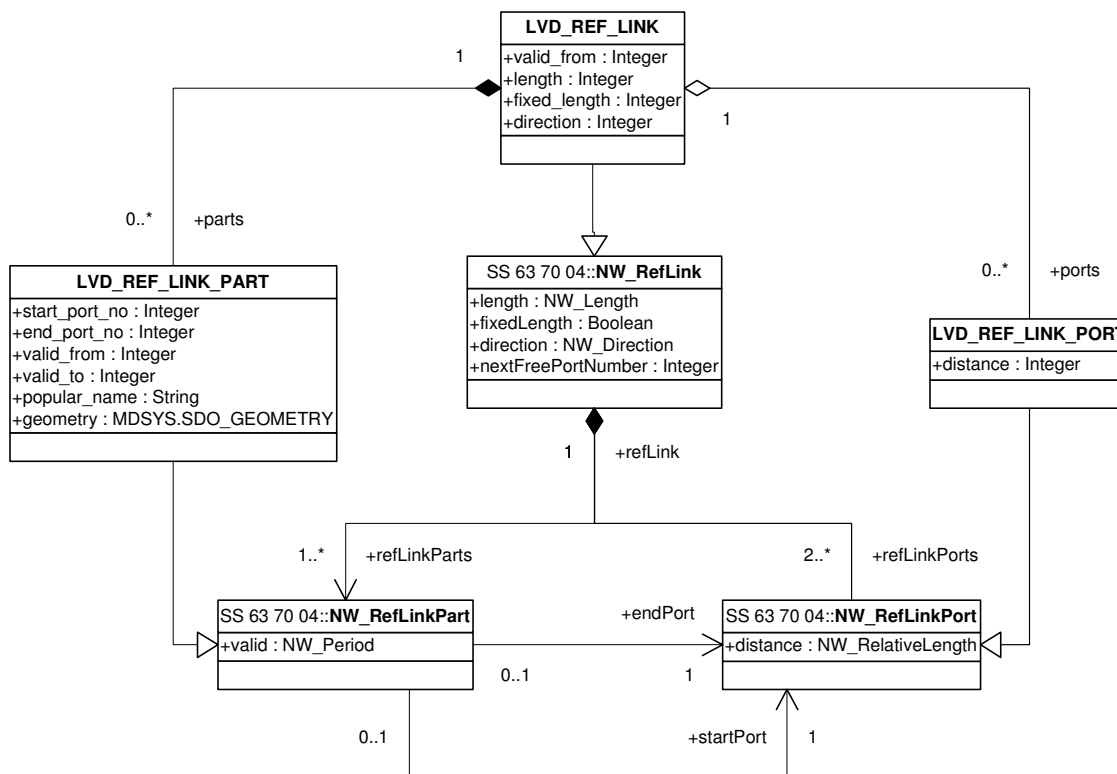


Fig 4.9: Modell som beskriver referenslänkar och referenslänksdelar i TEKIS-LV.

4.6.6. Referenslänksdel

För att möjliggöra stabiliteten hos referenslänkarna finns det till varje referenslänk en mängd delar, benämnda referenslänksdelar, vilka helt saknar stabilitet men som tillsammans alltid utgör hela referenslänken. Delarna definieras av den referenslänk till vilken de tillhör, ett löpnummer för att unikt identifiera dem på referenslänken och ett start- och slutavstånd för att beskriva deras läge längs det koordinatsystem vilket referenslänken ”spänner” upp. Begreppet referenslänksdel implementeras av *lvd_ref_link_part* och ärver från *NW_RefLinkPart* i SS 637004-1 [1] och beskrivs i **fig 4.9**.

start_port_no (associationsroll)

anger löpnummer för den port som referenslänksdelen startar i. Referenslänksdelar börjar och slutar i länkportar.

end_port_no (associationsroll)

anger löpnummer för den port som referenslänksdelen slutar i. Referenslänksdelar börjar och slutar i länkportar.

valid_from (attribut)

anger giltighetsstart på referenslänksdelen

datatyp: *Integer*

valid_to (attribut)

anger giltighetsslut på referenslänksdelen

datatyp: *Integer*

geometry (attribut)

geometri på referenslänksdelen. Geometrier för samtliga ingående referenslänksdelar utgör geometrin för hela referenslänken.

datatyp: *Line*

I fig 5.8 nedan så ges ett exempel på en referenslänk, RL1, och dess ingående delar, RLD1 och RLD2.

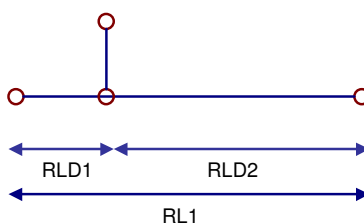


Fig 4.10: Beskrivning av en referenslänk med tillhörande referenslänksdelar samt en anslutande referenslänk.

Kommentar: Att det är referenslänksdelen och inte referenslänken som har en geometri är ett avsteg från standarden SS-63 70 04 [SIS-04]. Anledningen till att TEKIS-LV skiljer sig på den punkten är för att underlätta mappningen mellan portkonceptets topologi och nod-länk-topologi. Denna eftergift underlättar integrationen med geografiska informationssystem. Alla kommersiella geografiska informationssystem betraktar nät som geometriska nod-länk-system och delar kravet på att få arbeta direkt mot databasen. Det innebär att det måste gå att erhålla ett nod-länk-system direkt ur TEKIS-LV:s databas. Då varje referenslänksdel börjar och slutar i en port och inte innehåller några andra portar går det att betrakta dem som en-dimensionella topologiska primitiv, det vill säga länkar. Väljer man dessutom att låta varje koppling mellan två referenslänkar ske i referensnoder går det att betrakta referensnoderna som noll-dimensionella topologiska primitiv, det vill säga noder. Således kan man låta geografiska informationssystem betrakta klasserna *lvd_ref_node* och *lvd_ref_link_part* som nodklass respektive länkklass i ett nod-länk-system. Det medför dock att geometrin måste tillhöra referenslänksdelen och inte referenslänken och ett avsteg från standarden måste göras. En fördel med avsteget är förutom att det möjliggör integration med geografiska informationssystem att det går betydligt fortare att sätta samman linjegeometrier än att dela upp dem i spatials verktyg. Nackdelen med avsteget från standarden är förutom den uppenbara nackdelen med att frånga en standard även att det möjliggör felaktigheter i databasen genom att det går att lagra linjegeometrier i en referenslänks delar som inte tillsammans bildar en linjegeometri. Det är dock fullt möjligt att förhindra sådana felaktigheter genom serverlogik eller databastriggers. För TEKIS-LV har fördelarna ansetts väga tyngre än nackdelarna och därför ligger geometrin på referenslänksdelen.

För att underlätta generering av geometriska segment för utbredningarna (se avsnitt 4.7.1) finns det behov att även lagra geometrier för referenslänkarna. Dubbellagringen innebär en denomalisering av modellen och därför ska referenslänksgeometrierna betraktas som slavgeometrier till de sammantagna geometrierna av referenslänksdelarna.

4.6.7. Referensnod

Referensnoder i standarden SS 637004-1 [1] används för att beskriva topologiska val och för att markera slutet på en väg som inte fortsätter i en annan väg. Det vill säga, referensnoder används i punkter till vilka två eller flera referenslänkar är knutna och i punkter där en referenslänk slutar utan att fortsätta i någon annan referenslänk.

I NVDB har motsvarande begrepp, nätnoden, en annan definition där kopplingar mellan referenslänkar som inte innebär ett topologisk val inte sker via en nätnod utan direkt mellan referenslänkarna [9]. I **fig 4.11** nedan ges tre exempel på vägkopplingar vilka belyser skillnaden mellan standarden SS 637004-1 [1] och NVDB.

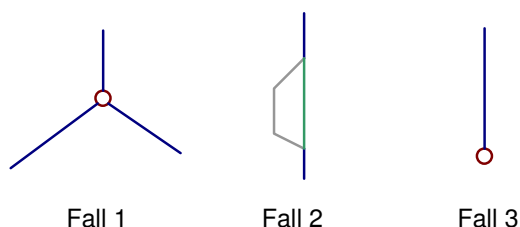


Fig 4.11: Tre fall av vägkopplingar.

Det första fallet beskriver en vägkoppling där fler än två vägar ingår. I kopplingen finns då en referensnod enligt standarden SS 637004-1 [1] och en nätnod enligt NVDB. Det andra fallet är det så kallade kurvrättningsfallet där de blå och de grå linjerna tillsammans utgör en referenslänk och där den gråfärgade delen har ersatts av en annan vägsträcka, nämligen referenslänken som är färgad grön i figuren. I kopplingarna mellan de två referenslänkarna finns det då ingen nätnod i NVDB i och med att kopplingarna inte innebär något topologiskt val. I standarden SS 637004-1 [1] finns det dock referensnoder i kopplingarna. Det tredje fallet beskriver en vägs slut och där placeras för standarden SS 637004-1 [1] och i NVDB en referensnod respektive en nätnod. I TEKIS-LV har man valt att ansluta sig till SS 637004-1 [1] och inte använda några fall av kopplingar mellan referenslänksdelar utan motsvarande referensnod. För kurvrättningsfallet använder TEKIS-LV referensnoder som anslutning mellan referenslänkarna. Detta angreppssätt medger att vägnätet enklare kan abstraheras till en traditionellt nod-länk-system, där nod motsvarar nätnod och länk motsvarar giltig referenslänksdel.

Referensnoder implementeras i TEKIS-LV av klassen *lvd_ref_node* som ärver från *NW_RefNode* i SS 637004-1 [1] och beskrivs i **fig 4.12**.

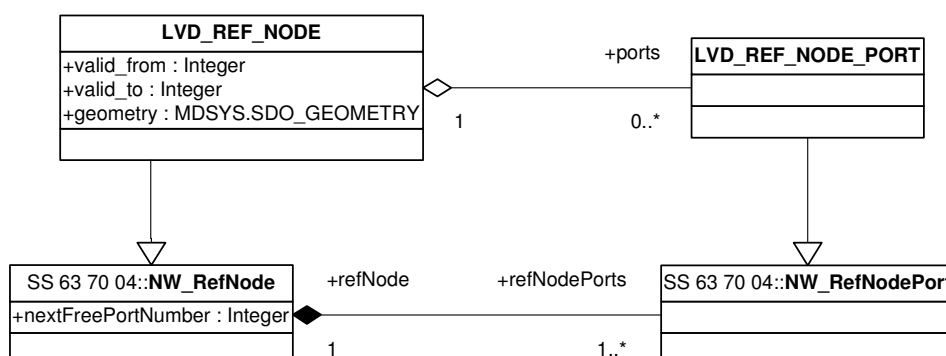


Fig 4.12: Modell av referensnod i TEKIS-LV.

valid_from (attribut)

anger giltighetsstart för referensnoden

datatyp: *Integer*

valid_to (attribut)

anger giltighetslut för referensnoden

datatyp: *Integer*

geometry (attribut)

geometri på referennoden.

datatyp: *Point*

4.7. Nätanknuten företeelse

Företeelser och utbredningar används för att beskriva information knuten till vägnätet. En företeelse motsvarar själva informationen medan utbredningen definierar hur informationen är geometriskt och topologiskt knuten till vägnätet. Till exempel beskrivs en lyktstolpe som står längs en väg med en företeelse av typen *Belysning* med ett antal attribut, varav ett är *Typ* vars värde är *Lyktstolpe*. Lyktstolpens läge utmed vägen definieras av en länkutbredning vilken kopplar lyktstolpen till vägen med hjälp av ett avstånd utmed vägen och ett sidoavstånd ut från vägen.

I TEKIS-LV beskrivs nästan all vägnätsinformation med hjälp av företeelser och utbredningar. En *nätanknuten företeelse* är en specialisering av en *företeelse* där man använder *utbredningar* som associationsklass till vägnätets topologi. Se **fig 4.13** för en schematisk beskrivning av hur en nätanknuten företeelse ansluter till vägnätet. De viktigaste undantagen är geometri och topologi vilka beskrivs av de topologiska entiteter som presenteras i avsnitt 4.6.

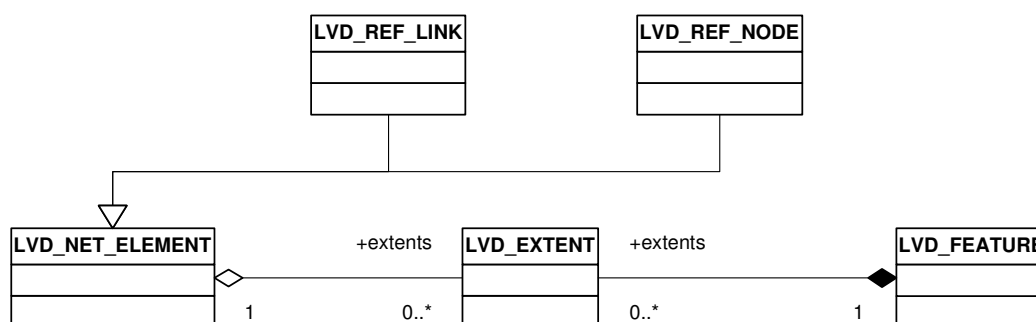


Fig 4.13: Schematisk modell av nätanknuten företeelse. En nätanknuten företeelse, *lvd_feature*, ansluter till nättopologin på ett generellt sätt genom associationsklassen *lvd_extent*.

Precis som för topologin i TEKIS-LV så bygger modellen av företeelser, inklusive modellerna för datakatalog och utbredningar, på svenska standarder. Företeelser och datakatalog finns standardiserade i standarden SS 637006 [2] medan utbredningar standardiserats i SS 637004-1 [1]. En del avsteg har gjorts från standarden SS 637006 [2] vilket kommer att belysas i de följande avsnitten.

4.7.1. Företeelse

Begreppet *företeelse* implementeras i TEKIS-LV av *lvd_feature* som ärver från begreppet *GFI_FeatureInstance* i standarden SS 637006 [2]. Företeelser används för att beskriva samtliga egenskaper på vägnätet. En beskrivning av modellen för *lvd_feature* finns i **fig 4.14**.

Företeelser ärver av förändringsbart objekt, *lvd_object*, vilket medför att de kan läsas, behörighetshanteras samt versions- och förändringshanteras.

feature_type_object_id (associationsroll)

anger företeelsetyp för företeelsen

valid_from (attribut)

anger giltighetsstart för företeelsen.

datatyp: *Integer*

valid_to (attribut)

anger giltighetsslut för företeelsen.

datatyp: *Integer*

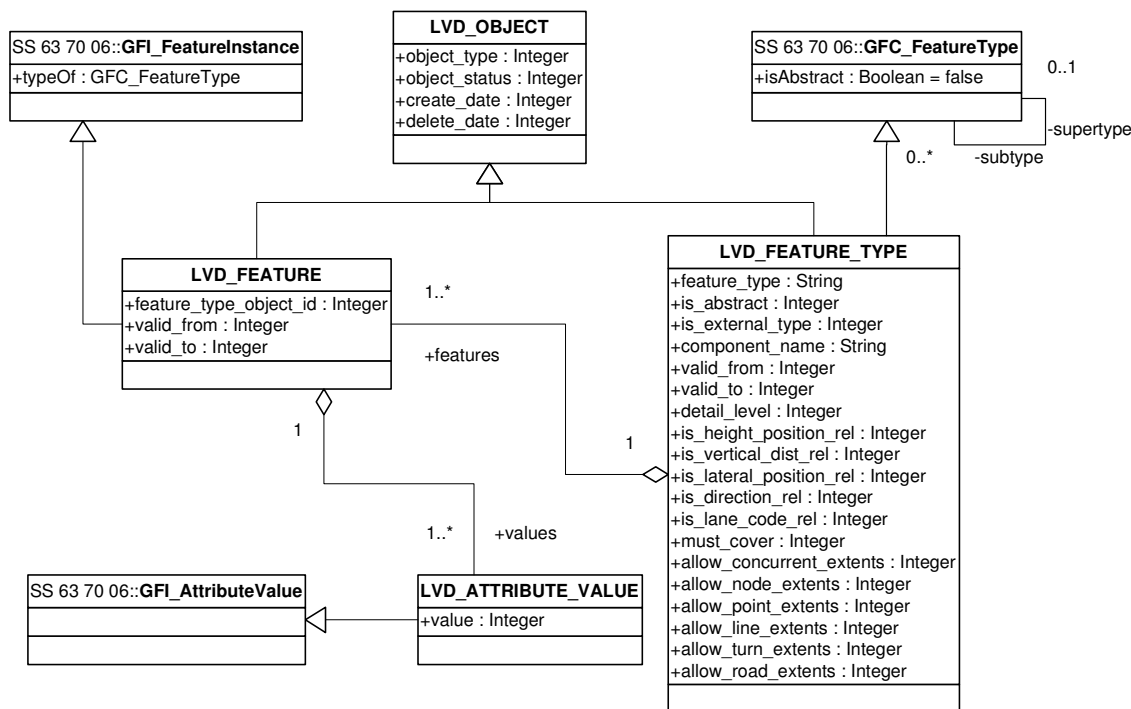


Fig 4.14: Modell av begreppet företeelse i TEKIS-LV.

Företeelsen är av en typ som finns registrerad i *lvd_feature_type*. En företeelsetyp kan antingen vara intern, det vill säga lagras helt på TEKIS-LV:s datakatalogformat, eller extern, det vill säga lagras i en extern datastruktur. För en djupare teknisk beskrivning av användningen av externa företeelsetyper hänvisas till dokumentet *TEKIS-LV – Systembeskrivning* [8].

En företeelses attributvärden innehåller värden för de attribut som hör till företeelsens typ och motsvarar entiteterna *GFI_AttributeInstance* i SS 637006 [2].

I standarden SS 637006 [2] beskrivs två typer av företeelser, nämligen företeelser med historik samt företeelse utan historik, där begreppet historik motsvarar begreppet verklighetshistorik i det här dokumentet. Förändringshistorik finns som tidigare nämnt standardiserat i SS 637007 [3] och för att hantera även den typen av historik införs begreppen *GFI_FeatureWithHistoryAndHistoryObject* samt *GFI_FeatureWithoutHistoryAndHistoryObject* i SS 637006 [2]. I TEKIS-LV är kopplingen mellan tids- och förändringsversioner tätare och företeelsers tidsversioner hanteras med hjälp av förändringsversioner av företeelserna, vilket medför att det kan finnas flera versioner av en företeelse som är gällande. Dock är det ett krav att giltighetstiderna för versionerna inte överlappar varandra vilket innebär att en företeelse kan ha flera gällande tidsversioner, men givet ett betraktelsesdatum så finns det en och endast en gällande version av en företeelse.

Motsvarigheten till begreppet *GFI_FeatureWithoutHistoryAndHistoryObject* är i TEKIS-LV en företeelse vilken inte kan ha fler än en gällande version.

Huruvida en företeelse har en verklighetshistorik beror på dess företeelsetyp. För till exempel trafikföreskriftsföreteelser kan det vara så att varje förändring av företeelsen (attributmässigt eller utbredningsmässigt) bör ge ett helt nytt företeelseobjekt. För andra företeelser kan det räcka med att skapa en ny tidsversion av företeelsen. En inventering behöver göras för alla företeelsetyper för att fastslå vilka företeelsetyper som kräver eller tillåter verklighetshistorik.

4.8. Datakatalog

Datakatalogen i TEKIS-LV möjliggör ett dynamiskt regelverk för informationen hos företeelserna i databasen. Modellen för datakatalogen bygger på den standardiserade datakatalogen i SS 637006 [2]. Dock har ett antal avsteg gjorts från standarden där den ansetts vara för generell eller för omfattande.

Det centrala begreppet i datakatalogen är företeelsetypen. En företeelsetyp specificerar dels ett antal regler för företeelser som har typen, bland annat hur de kan utbredas och om det kan existera två företeelser av samma typ för samma del av vägnätet, och dels vilka attribut som tillhör företeelsetypen. **Fig 4.15** visar en schematisk modell av datakatalogens entiteter.

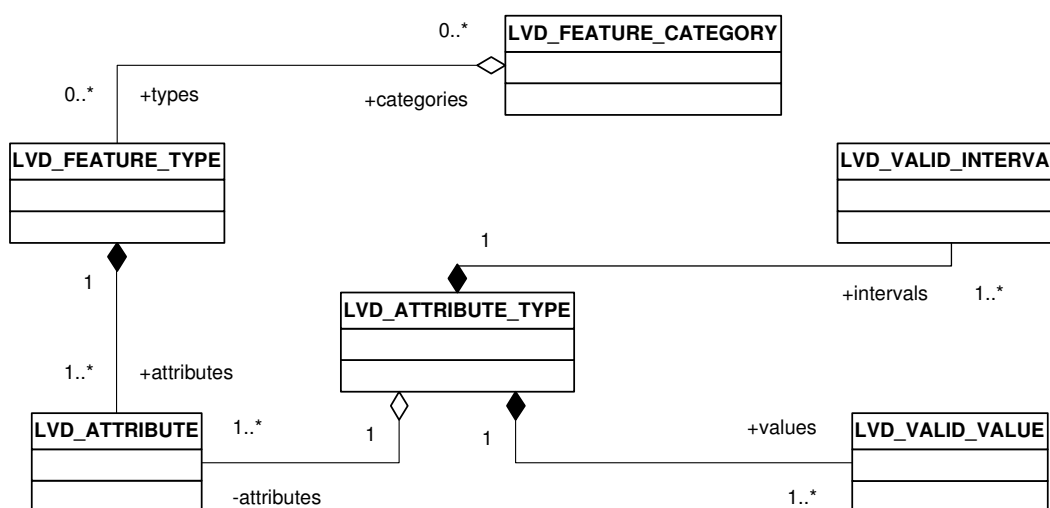


Fig 4.15: Schematisk modell av datakatalogens klasser.

Företeelsetyper ärver av objekt vilket medför att de kan låsas, behörighetshanteras samt versions- och förändringshanteras.

4.8.1. Företeelsetyp

Företeelsetyp implementeras i TEKIS-LV av *lvd_feature_type* som ärver av begreppet *GFC_FeatureType* i SS 637006 [2].

Fig 4.16 visar den kompletta modellen av *lvd_feature_type* i TEKIS-LV.

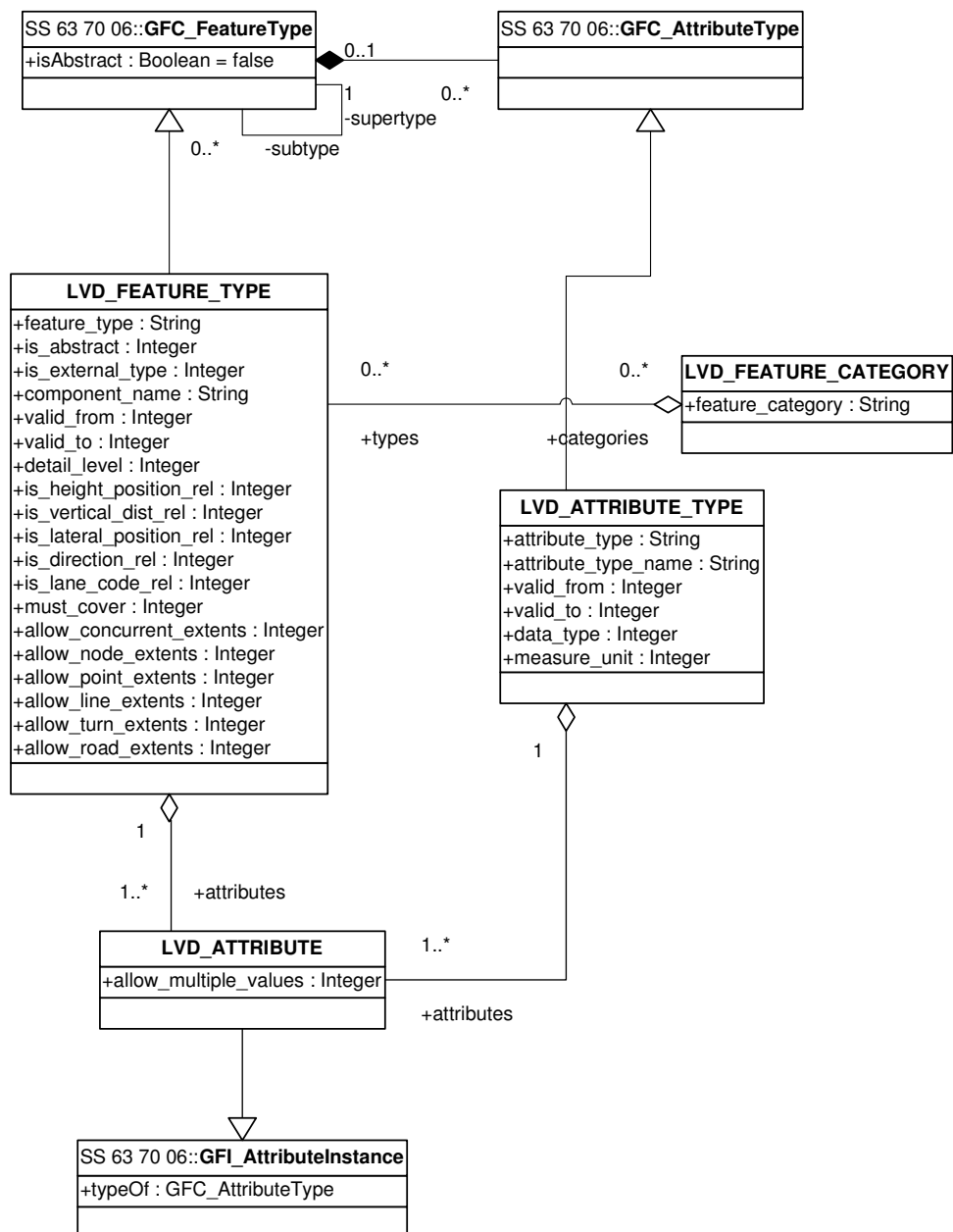


Fig 4.16: Modell av företeelsetyp, lvd_feature_type, i TEKIS-LV.

feature_type (attribut)

anger namn på företeelsetypen

datatyp: *String*

is_abstract (attribut)

flagga som indikerar huruvida företeelsetypen är abstrakt

datatyp: *Integer*

is_external_type (attribut)

flagga som indikerar huruvida företeelsetypen är extern. Om företeelsetypen är extern kan component_name användas för att registrera en extern företeelseklass som klienten använder för att instansiera lämplig klass för presentation/redigering av den externa företeelsen.

datatyp: *Integer*

external_table (attribut)

Databastabell som används för lagring av den externa företeelsetypen.

datatyp: *String*

external_schema (attribut)

Databasschema som används för lagring av den externa företeelsetypen.

datatyp: *String*

external_instance (attribut)

Databasinstans som används för lagring av den externa företeelsetypen.

datatyp: *String*

external_column (attribut)

Databaskolumn som används för lagring av den externa företeelsetypen.

datatyp: *String*

component_name (attribut)

anger extern component som kan instansieras av LV-klienten vid användning av externa företeelsetyper.

datatyp: *String*

valid_from (attribut)

anger giltighetsstart för företeelsetypen.

datatyp: *Integer*

valid_to (attribut)

anger giltighetsslut för företeelsetypen.

datatyp: *Integer*

is_height_position_rel (attribut)

anger huruvida höjdläge är relevant för företeelser av denna typ

datatyp: *Integer*

is_vertical_dist_rel (attribut)

anger huruvida höjdvstånd är relevant för företeelser av denna typ

datatyp: *Integer*

is_lateral_position_rel (attribut)

anger huruvida sidoläge är relevant för företeelser av denna typ

datatyp: *Integer*

is_lateral_dist_rel (attribut)

anger huruvida sidoavstånd är relevant för företeelser av denna typ

datatyp: *Integer*

is_direction_rel (attribut)

anger huruvida riktning är relevant för företeelser av denna typ

datatyp: *Integer*

is_lane_code_rel (attribut)

anger huruvida körfältskod är relevant för företeelser av denna typ

datatyp: *Integer*

must_cover (attribut)

anger om företeelser av denna typ måste ha fullständigt täckande utbredningar

datatyp: *Integer*

allow_concurrent_extents (attribut)

anger om företeelser av denna typ kan ha överlappande utbredningar

datatyp: *Integer*

allow_node_extents (attribut)

anger om företeelser av denna typ kan lägesanges med nodutbredningar

datatyp: *Integer*

allow_point_extents (attribut)

anger om företeelser av denna typ kan lägesanges med punktutbredningar

datatyp: *Integer*

allow_line_extents (attribut)

anger om företeelser av denna typ kan lägesanges med linjeutbredningar

datatyp: *Integer*

allow_turn_extents (attribut)

anger om företeelser av denna typ kan lägesanges med svängutbredningar

datatyp: *Integer*

allow_road_extents (attribut)

anger om företeelser av denna typ kan lägesanges med vägutbredningar

datatyp: *Integer*

4.8.2. Attributtyp

Ett annat viktig begrepp i datakatalogen är attributtypen vilken dels specificerar en mängd regler för ett attribut och dels definierar en giltig värdemängd för ett attribut. Begreppet attributtyp i TEKIS-LV ärver från begreppet *GFC_AttributeType* i standarden SS 637006 [2] och är modellerat enligt **fig 4.17** i TEKIS-LV.

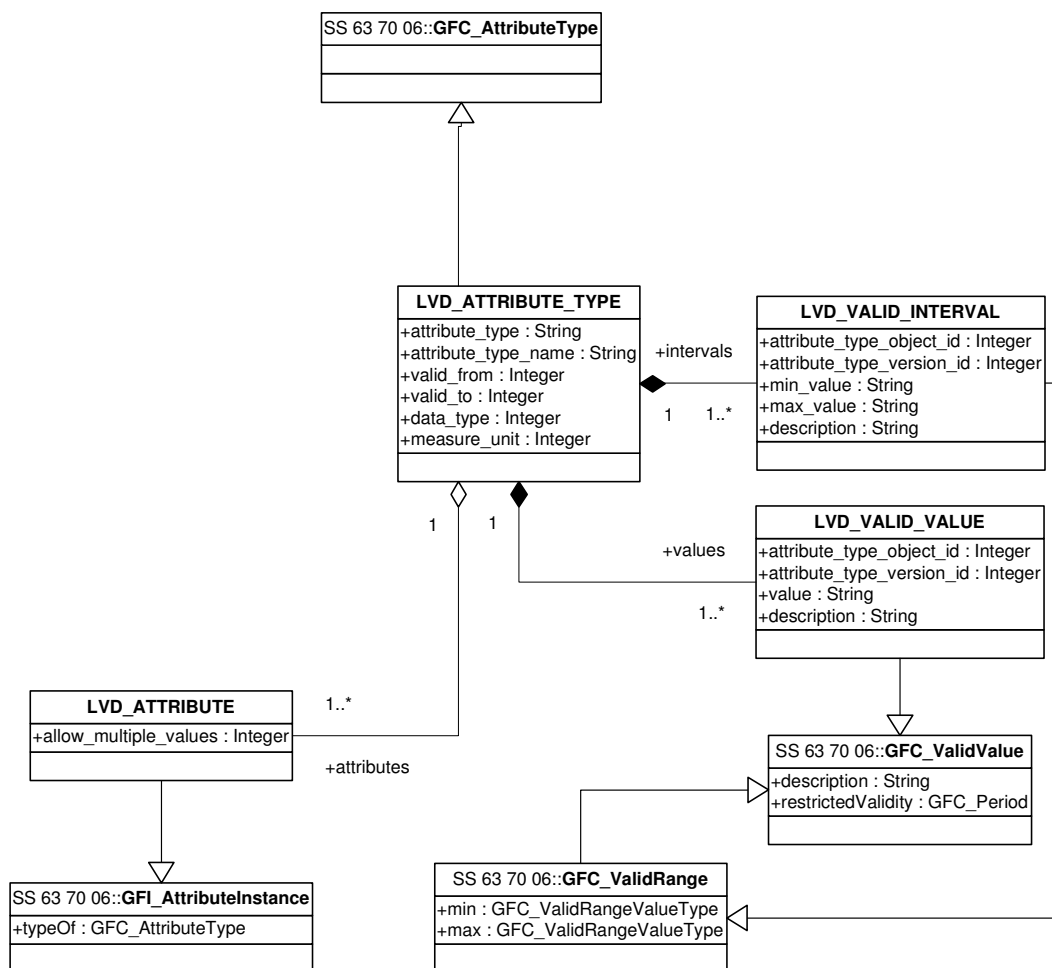


Fig 4.17: Modell av *lvd_attribute_type* i TEKIS-LV.

Attributtyper ärver precis som företeelsetyper från förändringsbart objekt, *lvd_object*, vilket medför att även de kan låsas, behörighetshanteras samt versions- och förändringshanteras.

attribute_type_name (attribut)

anger attributtypens namn

datatyp: *String*

valid_from (attribut)

anger attributets giltighetsstart

datatyp: *Integer*

valid_to (attribut)

anger attributets giltighetsslut

datatyp: *Integer*

data_type (attribut)

anger attributets datatyp

datatyp: *lvd_data_type* {boolesk, datum, datum och tid, tid, flyttal, heltal, referens, text, struktur}

measure_unit (attribut)

anger attributets måtenhet

datatyp: *lvd_measure_unit* {domän beskrivs av tabellen *lvd_measure_unit*}.

4.8.3. Attribut

Kopplingen mellan företeelsetyper och attributtyper sker via attribut. En företeelsetyp har en mängd attribut kopplade till sig där varje attribut har en attributtyp. Ett attribut i TEKIS-LV motsvarar kopplingen mellan begreppen *GFC_FeatureType* och *GFC_AttributeType* i standarden SS 637006 [2].

Begreppet *lvd_attribute* har följande attribut:

allow_multiple_values (attribut)

Anger om attributet kan vara ”flervärt”, d v s innehålla mer än ett värde.

datatyp: *Boolean*

attribute_name (attribut)

Anger namn på attributet.

datatyp: *String*

attribute_type (attribut)

Anger attributtyp.

datatyp: *String*

default_value (attribut)

Anger defaultvärde på attributet.

datatyp: *String*

must_exist (attribut)

Anger om detta attribut måste anges för att kunna spara.

datatyp: *Boolean*

is_private (attribut)

Anger om detta attribut ska vara privat, dvs inte synas i andra verksamheter.

datatyp: *Boolean*

is_external (attribut)

Anger om detta attribut ska vara externt, dvs lagras utanför TEKIS-LV:s datakatalog.

datatyp: *Boolean*

external_column_name (attribut)

Anger kolumnnamn för det `external_table_name` där det externa attributet lagras.

datatyp: *String*

4.9. Externa företeelsetyper

En extern företeelsetyp är en företeelsetyp som lagras i en extern datatrstruktur till skillnad från interna företeelsetyper vilka lagras i TEKIS-LV:s datakatalog. För en djupare beskrivning av detta hänvisas till *TEKIS-LV – Systembeskrivning* [8]. Metadata-beskrivningen av externa företeelsetyper hanteras av TEKIS-LV:s datakatalog för åtkomst och tolkning av externa data genom de generiska funktioner som finns i TEKIS-LV för sökning och presentation av externa företeelsedata.

4.9.1. Externa företeelser - företeelsetyp

Att beskriva den externa företeelsen innebär att man beskriver den precis som interna men att man dessutom binder varje attribut till en databaskolumn i en *extern databastabell*. För en extern företeelsetyp beskrivs den externa lagringen i *lvd_feature_type* genom attributen *is_external_type*, *external_table*, *external_schema*, *external_instance* och *external_column*.

För *externa attribut* och tillhörande funktionalitet för publicering beskrivs den externa lagringen i *lvd_attribute* av attributen *is_private*, *is_external* och *external_column_name*

Följande speciella regler ställs på externa företeelser och externa attribut.

Följande regler ställs på ovanstående attribut:

- Attributen *external_instance*, *external_schema*, *external_table_name*, *external_column_name* får bara vara NOT NULL om företeelsen är extern. (Det finns ett tänkbart undantag och det är om vi tillåter lagring i andra schema så skulle man kunna tillåta att *external_instance*, *external_schema* var NOT NULL även för ett icke extern företeelse.)
- Ett externt attribut får inte innehålla attributvärden.
- Ett externt attribut kan inte tillåta multipla värden (flervärt). (Notering: Oracles objekt-datatyper exempelvis SDO_GEOMETRY, tillåts ej heller eftersom det inte stöds av ADO.NET.)
- Ett extern attribut kan inte självt vara eller ingå i ett strukturerat attribut.

- En extern företeelse måste ha ett *internt* huvudattribut. Skälet till denna regel är att det måste finnas när man vill visa upp företeelserna vid tematisering eller efter en utsökning.

4.9.2. Externa företeelser – attributtyp

Eftersom ett externt attribut refererar till en kolumn, *lagringskolumnen*, finns det ett problem. Informationen för en extern attributtyp måste stämma överens med det som gäller för *lagringskolumnen*.

Detta berör t ex:

- **datatyp:** det är inte bra om datatypen på attributtypen är text medan lagringskolumnen är en NUMBER.
- **giltiga värden:** TEKIS-LV:s komponenter använder attributet *is_enumeration* hos *lvd_attribute_type* tillsammans med attributtypens giltiga värden, *lvd_valid_value*, för att exempelvis visa upp combo-boxar.

Följande regler införs:

- En extern attributtyp får inte vara en enumeration eftersom detta skulle innebära att TEKIS-LV skulle försöka visa den som en combobox utan giltiga värden. Man löser istället detta genom att definiera den externa företeelsen så att man joinar upp beskrivningsvärdet från domänen i en vy.
- En extern attributtyp får inte vara versionshanterad eftersom en kolumn inte kan vara det.

För sökningen används metadata till att bestämma följande:

- Vilka sökoperatörer som är tillåtna bestäms av datatypen.
- Vilka värden som är tillgängliga för att söka på bestäms via de giltiga värdena.

Givet ovanstående införs därför de externa attributtyperna *external_varchar*, *external_number*, etc som uppfyller ovanstående regler och som skall användas för alla externa attribut.

4.9.3. Externa företeelser – knytning till extern företeelse

Relationen mellan en *extern_företeelsetabell* och tabellen *lvd_feature_version* har alltid kardinaliteten [1:0] [1:0]. Observera att kopplingen till den externa företeelsen görs från företeelseversion och inte från företeelse. Det innebär att om företeelsetypen tillåter (redigerings- eller tids-) versioner så måste även externa data också göra det.

Att implementera relationen skulle kunnat gjorts på ett flertal olika sätt;

1. En främmande nyckel mot (*feature_object_id*, *feature_version_id*) på den externa företeelsetabellen.
2. En främmande nyckel från den externa företeelsetabellen till *lvd_feature_version*.
3. En relationstabell, kallad *extern_kopplingstabell*, mellan *lvd_feature_version* och den externa företeelsetabellen. Denna relationstabell skall ha både {*feature_ob-*

ject_id, feature_version_id} som primärnyckel och externa företeelsetabellens primärnyckel som alternativnyckel.

4. Den externa företeelsetabellens primärnyckel som attribut på företeelsen

Av dessa alternativ använder sig TEKIS-LV av alternativ 2. Nedan följer en beskrivning av våra överväganden inför det valet.

Diskussion:

Nackdelen med alternativ ett är att det kan vara svårt att få leverantörer av externa system att förändra sina tabeller och införa ett *feature_id*.

Nackdelen med alternativ två är att det inte går att införa "foreign key constraints" plus att man måste ställa krav på den externa företeelsetabellens primärnyckel (t ex att den inte består av flera kolumner, detta kan vara ett problem om den externa företeelsen är versionshanterad). Vad denna främmande nyckel pekar på bestäms av företeelsens typ.

Nackdelen med alternativ fyra är att det inte går att införa "foreign key constraints" plus att det förmodligen ger sämre prestanda.

Vad gäller alternativ tre måste man först **observera** att även om man inför en extra knytningstabell mellan den externa företeelsetabellen och tabellen *lvd_feature_version* så kan knytningstabellen alltid döljas genom att införa en *extern vy* som innehåller både den externa företeelsetabellens primärnyckel och företeelse-identiteten. Man behöver alltså inte veta namnet på denna tabell för att kunna generera SQL-frågor för sökningen.

Det är dessutom så att eftersom det är denna externa vy som innehåller den externa företeelsetabellens primärnyckel så måste denna vy joinas med i företeelsens list-dataset. Detta för att kunna visa företeelsen på ett *verksamhetsspecifikt sätt*.

Det är naturligtvis ändå så att alternativ tre inte är utan nackdel. Nackdelen är att när man skapar en basföreteelse så måste man veta hur man skall skapa en rad i denna tabell. Detta gör det hela mer komplext. Läs vidare som DelegateObjects under rubriken *Applikationsintegration* i dokumentet *TEKIS-LV – Systembeskrivning* [8].

Slutsats:

Givet ovan diskussion har alternativ två valts eftersom det innebär minst ingrepp vid integrationen av ett externt system i TEKIS-LV.

4.10. Utbredningar

Utbredningar anger en företeelses geometriska och topologiska läge i vägnätet. Begreppet utbredning i TEKIS-LV ärver från *NW_Extent* i standarden SS 637004-1 [1]. Utbredning är implementerat av *lvd_extent* i TEKIS-LV och beskrivs i **fig 4.18**.

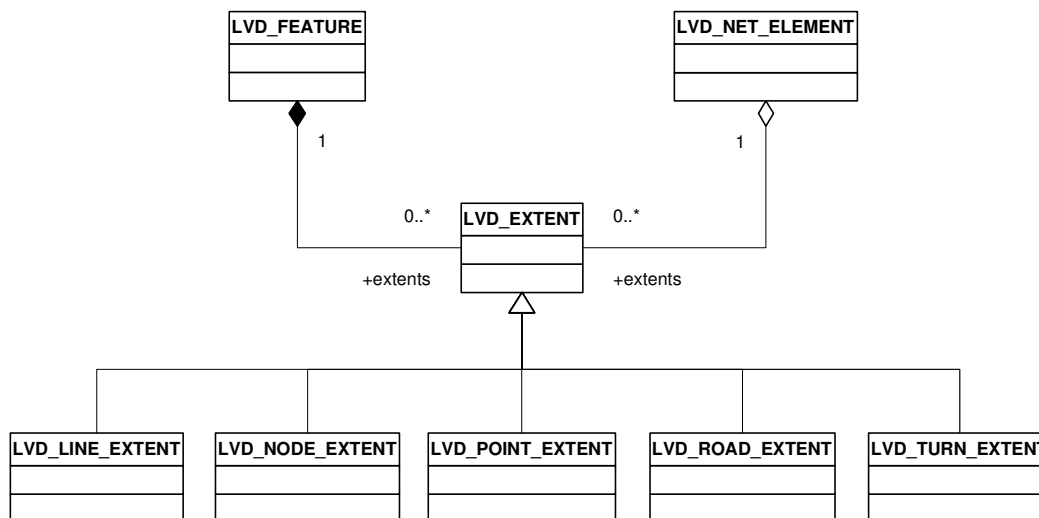


Fig 4.18: Schematisk modell av utbredningar i TEKIS-LV.

Klassen *lvd_extent* fungerar som sambandsklass mellan klasserna *lvd_feature* och *lvd_net_element* och har dessutom ett egenskapsattribut vilket anger utbredningens typ. Typen bestämmer hur företeelsen är knuten till vägnätet. I modellen finns det fem typer av knytningar vilka presenteras närmare i tabell 4.2 nedan.

Utbredningstyp	Beskrivning
Nodutbredning	En nodutbredning beskriver läget för företeelser som är knutna till en nod. Exempel på en sådan företeelse är Vändmöjlighet vilken anger möjligheten för olika fordon att vända i noden.
Punktutbredning	En punktutbredning beskriver läget för företeelser som är knutna till vägar och som saknar utsträckning, det vill säga är punktformade. Exempel på en sådan företeelse är Vägbom vilken anger en vägboms placering.
Linjeutbredning	En linjeutbredning beskriver läget för företeelser som är knutna till vägar och som har utsträckning, det vill säga är linjeformade. Exempel på en sådan företeelse är Bärlighet vilken beskriver ett vägavsnitts bärlighet.
Svängutbredning	En svängutbredning beskriver läget för företeelser som är knutna till noder och som beskriver en möjlig sväng från en referenslänk till en annan referenslänk via en nod.
Vägutbredning	En vägutbredning beskriver läget för företeelser som är knutna till en väg med ett visst vägnamn eller vägnummer.

Tabell 4.2: Beskrivning av de olika utbredningstyper som finns i TEKIS-LV.

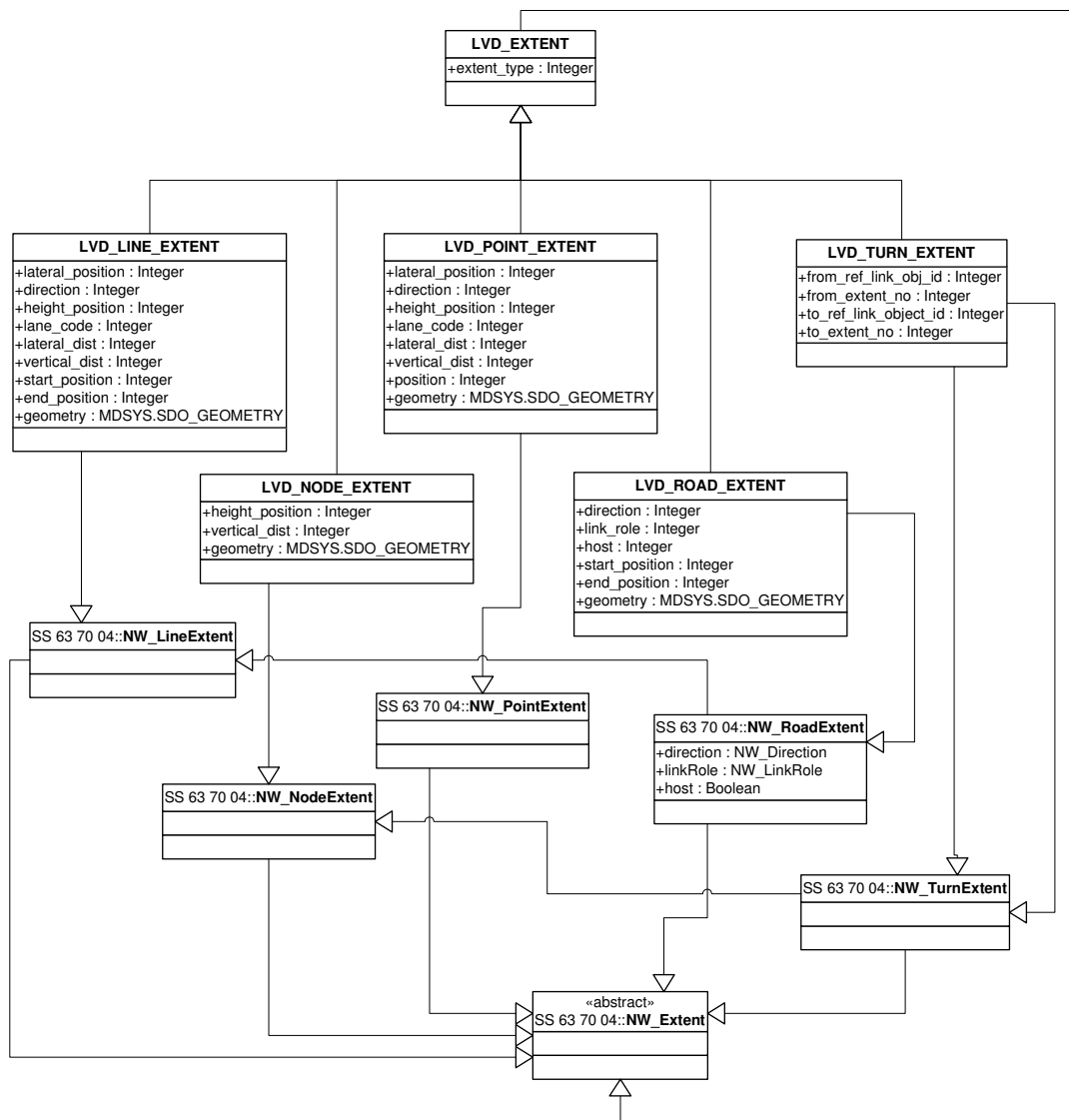


Fig 4.19: Modell av utbredningstyperna i TEKIS-LV.

De olika typerna av utbredningar motsvarar olika entiteter i standarden SS 637004-1 [1]. Begreppet nodutbredning motsvarar *NW_NodeExtent*, begreppet punktutbredning *NW_PointExtent*, begreppet linjeutbredning *NW_LineExtent*, begreppet svängutbredning *NW_TurnExtent* och begreppet vägutbredning *NW_RoadExtent*. Varje utbredningstyp har en motsvarande klass i TEKIS-LV:s lagringsmodell vilken ärver från utbredningsklassen. I **fig 4.19** presenteras de olika utbredningstyperna och deras knytning till utbredningsklassen.

extent_type (attribut)

anger typ av utbredning

datatyp: *lvd_extent_type* {*punktutbredning, linjeutbredning, vägutbredning, nodutbredning, svängutbredning*}

4.10.1. Punktutbredning

Punktutbredningar innehåller ett avstånd vilket definierar företeelsens läge längs den referenslänk den är knuten till.

Nodutbredningar, punktutbredningar och linjeutbredningar innehåller ett höjdläge som anger om företeelsen är knuten under, över eller på nätelementet samt ett höjdvstånd som anger hur långt under eller över företeelsen är knuten. Punktutbredningar och linjeutbredningar har dessutom sidolägen och sidoavstånd definierade vilka anger företeelsens läge vid sidan om referenslänken.

Förutom ovanstående attribut för generella utbredningar tillkommer följande för punktutbredningar:

lateral_position (attribut)

anger höjdläge

datatyp: *lvd_lateral_position* {*domän som beskrivs av tabellen lvd_lateral_position*}

direction (attribut)

anger riktning

datatyp: *lvd_direction* {*domän som beskrivs av tabellen lvd_direction*}

height_position (attribut)

anger höjdläge

datatyp: *lvd_height_position* {*domän som beskrivs av tabellen lvd_height_position*}

lane_code (attribut)

anger körfältskod

datatyp: *lvd_lane_code* {*domän som beskrivs av tabellen lvd_lane_code*}

lateral_distance (attribut)

anger offset

datatyp: *Double*

vertical_dist (attribut)

anger höjdvstånd

datatyp: *Double*

geometry (attribut)

slavgeometri som används för att cacha utbredningen för snabb uppläsning i GIS.

datatyp: *Point*

position (attribut)

anger läget relativt referenslänkens linjära referenssystem

datatyp: *Double*

4.10.2. Linjeutbredning

På samma sätt innehåller linje- och vägutbredningar två avstånd mellan vilka företeelsen är knuten till referenslänken. Klassen *lvd_line_extent* har samma attribut som *lvd_point_extent* förutom att position anges genom en start- och en slutposition.

start_position (attribut)

anger startläget relativt referenslänkens linjära referenssystem

datatyp: *Double*

end_position (attribut)

anger slutläget relativt referenslänkens linjära referenssystem

datatyp: *Double*

4.10.3. Vägutbredning

Vägutbredningar innehåller även information om länkroll och huruvida den är värd till andra vägutbredningar eller inte. Länkroller och värdattribut används då flera företeelser av samma typ som utbredds med en vägutbredning är knutna till samma referenslänk. Det är exempelvis användbart när man vill räkna fram den totala längden på europavägarna i en kommun. En första ansats vore att för varje europavägnummer beräkna längden för alla referenslänkar som har ett nummer och sedan summera över alla längder. Om två europavägar delar en del av vägnätet² kommer dock längden på vägvägsnittet adderas två gånger till den totala summan. Problemet löses genom att ett av vägnumren är satt till värd i de fall som en referenslänk har flera vägnummer knutna till sig.

direction (attribut)

anger riktning

datatyp: *lvd_direction* {domän som beskrivs av tabellen *lvd_direction*}

link_role (attribut)

anger länkroll

datatyp: *lvd_link_role* {domän som beskrivs av tabellen *lvd_link_role*}

host (attribut)

anger om denna utbredning är värd

datatyp: *Integer*

start_position (attribut)

anger startläget relativt referenslänkens linjära referenssystem

datatyp: *Double*

². Betänk E4:an och E20:ans utbredningar genom Stockholm kommun.

end_position (attribut)

anger slutläget relativt referenslänkens linjära referenssystem

datatyp: *Double*

geometry (attribut)

slavgeometri som används för att cacha utbredningen för snabb uppläsning i GIS.

datatyp: *Line*

4.10.4. Nodutbredning

Nodutbredningar finns inte implementerat i TEKIS-LV:s objektmodell. I lagringsmodellen för TEKIS-LV finns emellertid stöd för en senare implementation av nodutbredningar enligt beskrivningen som finns i SS 637004-1 [1].

Förutom ovanstående attribut för generella utbredningar tillkommer följande för nodutbredningar:

height_position (attribut)

anger höjdläge

datatyp: *lvd_height_position* {domän som beskrivs av tabellen *lvd_height_position*}

vertical_dist (attribut)

anger höjdvstånd

datatyp: *Double*

geometry (attribut)

slavgeometri som används för att cacha utbredningen för snabb uppläsning i GIS.

datatyp: *Point*

4.10.5. Svängutbredning

Svängutbredningar innehåller kopplingar till två punktutbredningar. Punktutbredningarna anger start och slut för svängutbredningen genom att definiera lägen på de referenslänkar från respektive till vilka svängutbredningen spänner.

Standarden SS 637004-1 [1] gör det möjligt att ange en utbrednings position med hjälp av en koordinat. Koordinatangivna positioner för utbredningar ingår dock inte i den grundläggande datamodellen och bortses därför ifrån tills vidare.

from_ref_link_object_id (asociationsroll)

anger referenslänk där svängutbredningen startar

from_extent_no (asociationsroll)

anger punktutbredning där svängutbredningen startar

to_ref_link_object_id (asociationsroll)

anger referenslänk där svängutbredningen slutar

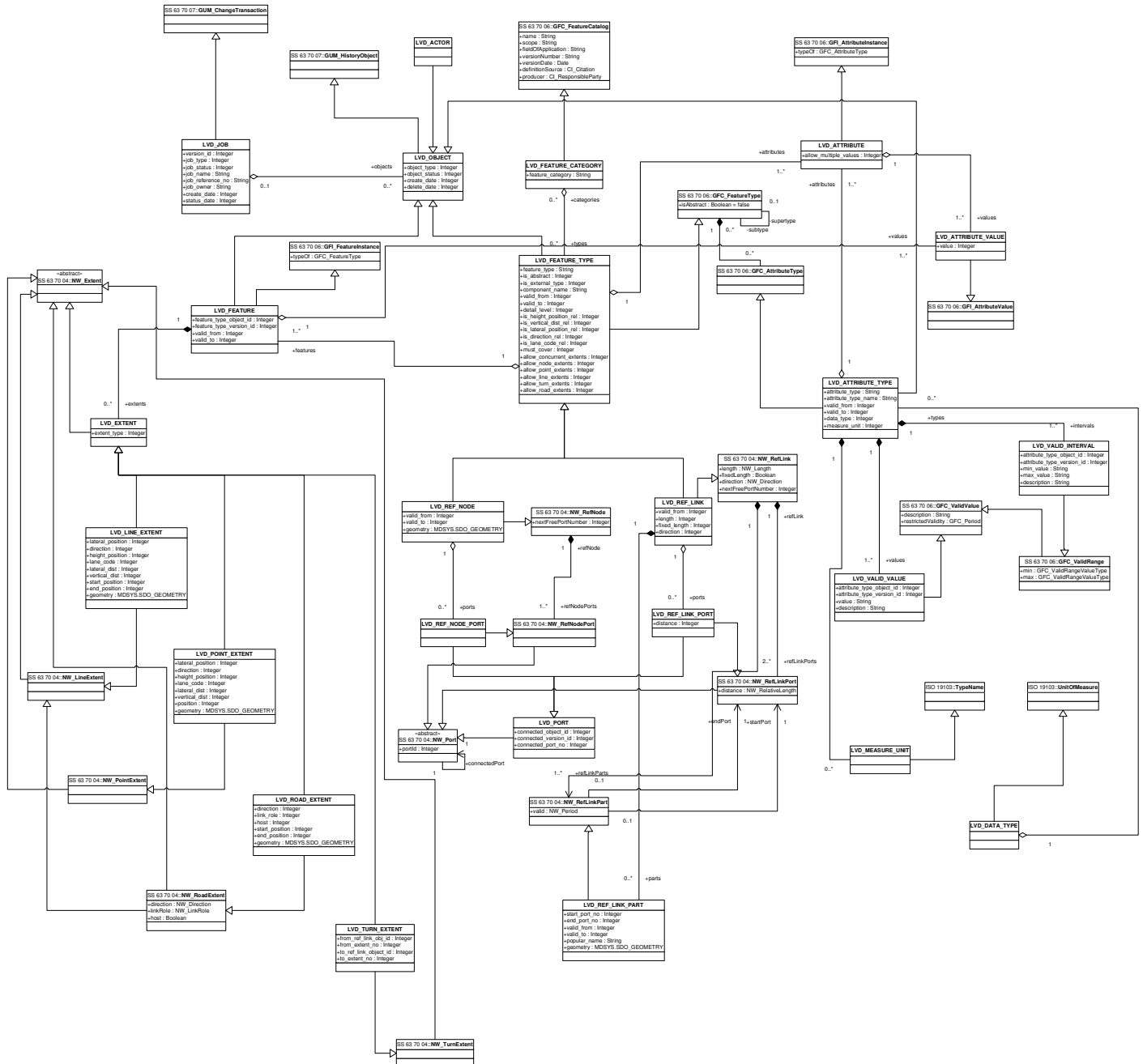
to_extent_no (asociationsroll)

anger punktutbredning där svängutbredningen slutar

4.11. Detaljeringsnivåer

Funktionalitet för hantering av detaljeringsnivåer är inte implementerad. I lagringsmodellen för TEKIS-LV finns emellertid stöd för en senare implementation av detaljeringsnivåer enligt beskrivningen som finns i SS 637004-1 [1].

Bilaga A, Fullständig tillämpningsmodell



Referenser

- [1] SIS (2006), *Geografisk information – Väg- och järnvägsnät – Del 1: Begreppsmodell och applikationsschema*, SS 637004-1, SIS Förlag
- [2] SIS (2006), *Geografisk information – Typoberoende representation av geografiska företeelser*, SS 637006, SIS Förlag
- [3] SIS (2006), *Geografisk information – Representation av förändringar i datamängder*, SS 637007, SIS Förlag
- [4] SIS (2004), *Samverkande GIS med ISO 19100 – en handbok om tekniskt ramverk för geografisk information*, Stanli
- [5] SIS (1996), *GIS-ordboken*, STG Hb 167, ISBN 91-7162-409-0, SIS Förlag
- [6] Tekis (2007), *TEKIS-LV – Administratören – Handbok*, 2-05, Tekis AB
- [7] Tekis (2007), *TEKIS-LV – Nätredigering – Användningsfall*, 2-02, Tekis AB
- [8] Tekis (2007), *TEKIS-LV – Systembeskrivning*, 1-05, Tekis AB
- [9] Nilsson och Hallberg (2001), *NVDB – Översiktlig informationsmodell*, Vägverket
- [10] Tekis (2007), *TEKIS-LV – Lagringsmodell – Diagram*, 1-03, Tekis AB
- [11] Tekis (2007), *TEKIS-LV – Lagringsmodell – Referens*, 1-04, Tekis AB