

Samuli Viita

UUDEN SUKUPOLVEN TYÖKALUT JA TYÖTAVAT IEC 61850 SÄHKÖASEMA- TOIMITUSTEN SUUNNITTELUSSA

Diplomityö
Informaatioteknologian ja viestinnän tiedekunta
Tarkastaja: Professori Sami Repo
Tarkastaja: Professori Pertti Järventausta
Huhtikuu 2023

TIIVISTELMÄ

Samuli Viita: Uuden sukupolven työkalut ja työtavat IEC 61850 sähköasematoimitusten suunnittelussa

Diplomityö

Tampereen yliopisto

Sähkötekniikan DI-ohjelma

Huhtikuu 2023

IEC 61850 on oleellinen osa nykyaikaisia sähköasematoimituksia. Standardi pyrkii mahdollistamaan eri valmistajien IED-laitteiden yhteentoimivuuden ja kattaa kaikki sähköasematoimituksen vaiheet määrittelystä ylläpitoon. Standardissa kuvattua suunnittelu- ja konfigurointiprosessia ei kuitenkaan ole laajamittaisesti hyödynnetty ja nykyisellään IEC 61850 –toteutukset rakentuvat usein laite kerrallaan useiden eri henkilöiden toimesta. Yhtenä rajoitteena standardissa kuvattujen toimintojen käyttöönotolle on pidetty saatavilla olevien työkalujen puutteellisuutta. Diplomityön tavoitteena on arvioida standardissa kuvatun suunnitteluprosessin soveltuvuutta nykyaikaiseen laiteistoriippumattomaan sähköasematoteutukseen ja muodostaa yhtenäinen toimintatapa IEC 61850 –toteutusten suunnitteluun kohdeyhtiössä. Keskeistä tämän kannalta on löytää käyttötarkoitukseen soveltuva kolmannen osapuolen suunnittelutyökalu.

Työssä käytiin läpi sähköasema-automaatiojärjestelmien rakennetta ja tulevaisuuden kehitysuuntia sekä IEC 61850 –standardia tarvittavilta osin. Työn aikana selvitettiin, mitä suunnitteluprosessin soveltamiseen käytettäviä ohjelmistoja on kaupallisesti saatavilla. Työssä perehdyttiin ohjelmiston valintaan käytettyihin menetelmiin, joista analyyttistä hierarkiaprosessia käytettiin löydettyjen ohjelmistojen arviointiin. Arvioiduista ohjelmistoista kaksi parasta valittiin tarkempaan vertailuun, joka toteutettiin luomalla oikeaan toimitusprojektiin perustuva testiprojekti. Molemmilla ohjelmistoilla luotiin tämän pohjalta eri laitevalmistajien laitteita sisältänyt IEC 61850 –konfiguraatio, jota kokeiltiin viedä eri IED-konfigurointiohjelmistoihin. Vertailun avulla pyrittiin valittujen ohjelmistojen lisäksi arvioimaan suunnitteluprosessia ja sen soveltamisessa mahdollisesti ilmeneviä rajoitteita.

Vertailut ohjelmistot edustivat keskenään hieman erilaisia lähestymistapoja IEC 61850 –järjestelmän konfiguroinnissa. Toisessa konfigurointi oli hyvin manuaalista, kun taas toisessa työntekijä oli automatisoitu, mikä tekee ohjelmistosta järjestelmäintegraattorin näkökulmasta houkuttelevamman vaihtoehdon. Vertailussa havaittiin kuitenkin, että laitevalmistajien ohjelmistot rajoittavat merkittävästi kolmannen osapuolen työkalujen ja suunnitteluprosessin käyttöä. Ainoastaan yhteen kolmesta IED-konfigurointiohjelmistosta pystyttiin tuomaan kolmannen osapuolen ohjelmistolla luotu IEC 61850 –järjestelmäkonfiguraatiodietoisto ilman suuria ongelmia. Oikealla kolmannen osapuolen työkalulla voitaisiin kuitenkin tehostaa IEC 61850 –järjestelmän konfigurointia lähitulevaisuudessa. Kolmannen osapuolen työkalun käyttö tulee kuitenkin rajoittamaan toisilaitteiden ja sitä olisi tarpeellista pilotoida oikeassa sähköasemaprojektissa tai testijärjestelmässä ennen laajempaa käyttöönottoa. Suunnitteluprosessiin kuuluva spesifointivaihe jäi työssä vähemmälle huomiolle ja standardissa määriteltyjen spesifikaatiodiestojen sisältö on tällä hetkellä melko suppea. Standardiin valmisteilla olevat laajennukset saattavat kuitenkin muuttaa tilannetta huomattavasti tulevaisuudessa. Työssä ei muodostettu yhtenäisiä toimintatapoja IEC 61850 –toteutusten suunnitteluun, mutta sen pohjalta saatiin hyvä näkemys saatavilla olevista kolmannen osapuolen konfigurointityökaluista ja niiden soveltamismahdollisuuksista, mikä voi edesauttaa toteutusten yhtenäistämistä tulevaisuudessa.

Avainsanat: IEC 61850, SCL, sähköasema-automaatio, sähköasemat

Tämän julkaisun alkuperäisyys on tarkastettu Turnitin OriginalityCheck –ohjelmalla.

ABSTRACT

Samuli Viita: Next generation tools and methods in engineering of IEC 61850 substation projects

Master's Thesis

Tampere University

Master's Programme in Electrical Engineering

April 2023

IEC 61850 standard is an essential part of modern substation deliveries. It aims to enable interoperability between different vendor IEDs, and it covers all phases of substation projects from specification to maintenance. However, the engineering process described in the standard has not been widely utilized and IEC 61850 –implementations are built one device at a time, usually by different people. Inadequate tools are seen as one constraint for implementing features described in the standard. Objective of this thesis is to assess the feasibility of the standardized engineering process in modern vendor independent substation projects and to form a uniform way of engineering IEC 61850 –systems in the target company. Essential for achieving these objectives is to find a suitable 3rd party engineering tool.

This thesis provides an overview of substation automation systems, the future trends of these systems as well as the IEC 61850 –standard. Currently commercially available engineering software and software selection methods were examined. Analytic hierarchy process was used to assess the software, from which two of the most highly ranked were selected for more in-depth comparison. The selected software tools were compared by creating a multi-vendor IEC 61850 –system configuration, based on a test project modelled after a real substation delivery. In addition to evaluating the selected software, objective of this comparison was to assess the engineering process and possible limitations that might come up in the implementation.

The evaluated software represented a slightly different approach in the configuration of IEC 61850 –systems. The other had more manual approach whereas in the other, the workflow was more automated, which makes it a more attractive option from a system integrator's point of view. However, it was observed during the comparison that vendor software impose major limitations to the application of the 3rd party tools and the engineering process. Importing an IEC 61850 –system configuration file created with a 3rd party software was possible only in one out of three IED-configuration software without any significant problems. With the right 3rd party tool however, it would be possible to enable more efficient configuration of IEC 61850 –systems in the near future. Nevertheless, using a 3rd party engineering tool will limit the options of secondary devices, and it should be tested on a real substation project or in a test system before wider deployment. The specification phase in the engineering process was not widely addressed in the thesis and the content of the standardized specification files is currently rather limited. The prepared extensions to the standard, could however affect this in the future. Uniform way of engineering IEC 61850 –systems was not formed in the thesis, but good understanding of available tools and their application was achieved, which may help to accomplish this task in the future.

Keywords: IEC 61850, SCL, substation automation, substations

The originality of this thesis has been checked using the Turnitin OriginalityCheck service.

ALKUSANAT

Tämä diplomityö tehtiin Enersense PN Oy:n sähköasemaliiketoimintaan. Haluan erityisesti kiittää Enersensellä työn ohjaajana toiminutta diplomi-insinööri Sami Lehtimäkeä kaikista työn aikana saamistani neuvoista ja kommenteista. Lisäksi haluan kiittää professori Sami Repoa työn kommentoinnista ja ohjaamisesta. Kiitokset myös perheelle ja ystäville tuesta pitkäköksi venähtäneen kirjoitusprosessin aikana.

Tampereella, 25.4.2023

Samuli Viita

SISÄLLYSLUETTELO

1. JOHDANTO	1
1.1 Työn tausta ja tavoitteet.....	1
1.2 Työn rakenne ja tutkimusmenetelmät	3
2. SÄHKÖASEMA-AUTOMAATIOJÄRJESTELMÄT	4
2.1 Suojausjärjestelmä	5
2.2 Ohjausjärjestelmä.....	7
2.3 Tietoliikenneprotokollat	8
2.4 Mittaukset	9
2.5 Digitaalinen sähköasema.....	11
2.6 Keskitetty suojaus- ja ohjausjärjestelmä	12
3. IEC 61850 SÄHKÖASEMATOIMITUKSISSA	14
3.1 Standardin historia.....	14
3.2 Standardin rakenne	15
3.3 Tietomalli	17
3.4 Tietoliikennepalvelut	21
3.4.1 Datasetit	23
3.4.2 Raportointi ja lokien keruu.....	24
3.4.3 GSE	25
3.4.4 Näytearvojen lähetys	26
3.5 Tietoliikenneprofiilit ja -protokollat	27
3.6 Työkalut.....	31
3.7 SCL	32
3.8 Suunnitteluprosessi	36
3.9 Signaalien nimeäminen	39
3.10 SCL-tiedostojen validointi	41
4. OHJELMISTOTYÖKALUJEN ARVIOINTI JA VALINTA	43
4.1 Ohjelmistojen arviointimenetelmät	43
4.2 Ohjelmiston valinta	45
4.3 Arvioinnin tulokset	50
5. SÄHKÖASEMA-AUTOMAATIOJÄRJESTELMÄN KONFIGUROIINTI UUDEN SUKUPOLVEN TYÖKALUILLA	52
5.1 Tyypikonfiguraatiot ja ICD-tiedostot	53
5.2 Konfigurointiprosessi	54
5.3 Järjestelmäspesifikaatio.....	56
5.4 Järjestelmäkonfiguraatio	59
5.5 SCD-tiedostojen vienti Siemensin ICT-työkaluun.....	65

5.6	SCD-tiedostojen vienti GE:n ICT-työkaluun	66
5.7	SCD-tiedostojen vienti Hitachin ICT-työkaluun.....	68
5.8	Dokumentointiominaisuudet.....	70
6.	YHTEENVETO.....	73
	LÄHTEET	77
	LIITE A: TESTIPROJEKTIN PÄÄKAAVIO	84

KUVALUETTELO

Kuva 1.	<i>Esimerkki keskitetyn suojaus- ja ohjausjärjestelmän arkkitehtuurista. [40].....</i>	13
Kuva 2.	<i>Standardin osien väliset yhteydet. [2].....</i>	16
Kuva 3.	<i>Loogiset solmut sekä niiden mallintamat komponentit ja toiminnot GIS-kojeistossa. [53].....</i>	17
Kuva 4.	<i>Tietomallin rakenne. [2, s. 24].....</i>	19
Kuva 5.	<i>Metamallin tasot. Perustuu lähteisiin [55], [56].</i>	20
Kuva 6.	<i>Meta-metamalli. Perustuu lähteeseen [55].</i>	21
Kuva 7.	<i>Palvelimen sisältämä hierarkia ja sen kuvaamiseen liittyvät palvelut [56, s. 30].</i>	23
Kuva 8.	<i>OSI-viitemalli ja profiilit. [52].....</i>	27
Kuva 9.	<i>IEC 61850 –tietoliikenneprofiilit. Perustuu lähteisiin [51], [52], [60].</i>	29
Kuva 10.	<i>GOOSE-sanomien lähetys ennen tapahtumaa ja sen jälkeen [60, s. 35].</i>	30
Kuva 11.	<i>GOOSE- tai SV-sanoma Ethernet-kehyksessä. Perustuu lähteisiin [49], [50].</i>	30
Kuva 12.	<i>XML-skeeman rakenne ja elementtien muodostumissuhteet. [50, s. 37].</i>	33
Kuva 13.	<i>SCL:n oliomalli ilman kantaluokkia ja attribuutteja. [50].....</i>	35
Kuva 14.	<i>Standardissa kuvattu työnkulku vaihe vaiheelta. [49]</i>	37
Kuva 15.	<i>Esimerkki bottom-up-prosessista. Perustuu lähteeseen [76].</i>	39
Kuva 16.	<i>IEC 61850 –signaalin nimen muodostuminen. [50, s. 46].</i>	40
Kuva 17.	<i>Tuoteliitännäinen nimeäminen. [50, s. 47].....</i>	40
Kuva 18.	<i>Toimintoliitännäinen nimeäminen. [50, s. 48]</i>	41
Kuva 19.	<i>Ohjelmistojen vertailua varten muodostettu AHP hierarkia</i>	48
Kuva 20.	<i>Helinks STS –työtilan aloitusvalikko.</i>	55
Kuva 21.	<i>SCL Managerin Project Explorer –näkyvä.</i>	56
Kuva 22.	<i>Osa malliprojektin pääkaaviosta piirrettynä Helinks STS:llä.</i>	57
Kuva 23.	<i>Kenttä AE02 SCL Managerilla piirrettynä.</i>	58
Kuva 24.	<i>GOOSE-sovellus Helinks STS:ssä.</i>	59
Kuva 25.	<i>Tyhjä FCDA-viittaus SCL Managerin Dataset information –ikkunassa.</i>	61
Kuva 26.	<i>Helinks STS:llä piirretty väyläkaavio.</i>	63

TAULUKKOLUETTELO

<i>Taulukko 1.</i>	<i>Loogisten solmujen ryhmittely. [54, s. 19].....</i>	<i>18</i>
<i>Taulukko 2.</i>	<i>Saatyn asteikko. Perustuu lähteeseen [89, s. 15].....</i>	<i>48</i>
<i>Taulukko 3.</i>	<i>Satunnaisindeksin arvot eri kokoisille matriiseille [89, s. 25].....</i>	<i>50</i>
<i>Taulukko 4.</i>	<i>Kriteereistä muodostettu parivertailumatriisi ja painovektori.</i>	<i>51</i>
<i>Taulukko 5.</i>	<i>Ohjelmistovertailun tulokset.</i>	<i>51</i>
<i>Taulukko 6.</i>	<i>Testiprojektiin valitut IED-laitteet ja ohjelmistot.....</i>	<i>53</i>

LYHENTEET JA MERKINNÄT

2PK	kaksoispääkisko
2PK+AK	kaksoispääkisko-apukisko
ACSI	Abstract Communication Service Interface
APPID	Application Identifier
ASD	Application Specification Description
BAP	Basic Application Profile
BCU	Bay Control Unit
BRCB	Buffered Report Control Block
CID	Configured IED Description
CIGRE	Conseil International des Grands Réseaux Électriques
COTS	Commercial of-the-Shelf
CPC	Centralized Protection and Control
CSV	Comma-Separated Values
DA	Data Attribute
DO	Data Object
DNP	Distributed Network Protocol
ENTSO-E	European Network of Transmission System Operators for Electricity
FCD	Functionally Constrained Data
FCDA	Functionally Constrained Data Attribute
FPN	Flexible Product Naming
FTD	Function Template Description
GIS	Gas Insulated Switchgear
GoCB	GOOSE Control Block
GOOSE	Generic Object Oriented Substation Event
GSE	Generic Substation Event
GSSE	Generic Substation State Event
HSR	High-availability Seamless Redundancy
ICD	IED Capability Description
ICT	IED Configuration Tool
IED	Intelligent Electronic Device
IEEE	Institute of Electrical and Electronics Engineers
IEC	International Electrotechnical Commission
IID	Instantiated IED Description
IMU	Intelligent Merging Unit
IOU	Input Output Unit
IP	Internet Protocol
ISD	IED Specification Description
ISO	International Organization for Standardization
LCB	Log Control Block
LD	Logical Device
LN	Logical Node
MCAA	Multicast Application Association
MMS	Manufacturing Message Specification
MSVCB	Multicast Sampled Value Control Block
MU	Merging Unit
NCIT	Non-Conventional Instrument Transformer
OSI	Open Systems Interconnection
OSMOSE	Optimal System-Mix Of flexibility Solutions for European electricity
PIU	Process Interface Unit
PRP	Parallel Redundancy Protocol
PTP	Precision Time Protocol
RCB	Report Control Block

RTF	Rich Text Format
RSTP	Rapid Spanning Tree Protocol
SAMU	Stand-Alone Merging Unit
SDH	Synchronous Digital Hierarchy
SED	System Exchange Description
SCD	System Configuration Description
SCL	System Configuration description Language
SCT	System Configuration Tool
SCSM	Specific Communication Service Mappings
SOE	Sequence of Events
SONET	Synchronous Optical NETWORKing
SSD	System Specification Description
SST	System Specification Tool
SV	Sampled Value
SVCB	Sampled Value Control Block
RTU	Remote Terminal Unit
TCI	Tag Control Information
TCP	Transmission Control Protocol
TPAA	Two Party Application Association
TPID	Tag Protocol Identifier
URCB	Unbuffered Report Control Block
UCA	Utility Communication Architecture
UCAIug	UCA International Users Group
UDP	User Datagram Protocol
UML	Unified Modelling Language
USVCB	Unicast Sampled Value Control Block
VID	VLAN Identifier
VLAN	Virtual Local Area Network
XML	eXtensible Markup Language
λ_{\max}	suurinta ominaisvektoria vastaava ominaisarvo
A	parivertailumatriisi
CI	johdonmukaisuusindeksi
CR	johdonmukaisuussuhde
RI	satunnaisindeksi
w	prioriteettivektori

1. JOHDANTO

IEC 61850 on kansainvälinen standardisarja, joka on vakiintunut osaksi nykyaikaisia sähköasema-automaatioitoimituksia. Se kattaa toimituksen kaikki vaiheet automaatiojärjestelmän määrittelystä käyttöönottoon, testaukseen ja ylläpitoon [1, s. 22]. Standardi pyrkii mahdollistamaan eri valmistajien IED-laitteiden (*Intelligent Electronic Device*) yhteentoimivuuden, jolla tarkoitetaan niiden kykyä viestiä ja toimia yhdessä samassa tietoverkossa. IED on määritelmästä riippuen yhden tai useamman mikroprosessorin sisältävä laite, joka kykenee vastaanottamaan tai lähettämään tietoa ja ohjauksia ulkoisesta lähteestä [2] tai laajemmin mikä tahansa sähkölaitteen hallinnassa käytettävä mikroprosessorin ja ohjelmiston sisältävä laite [3]. Termiä voidaan käyttää myös vielä yleisemmin kuvaamaan mitä tahansa mikroprosessoripohjaisia laitteita tietokoneita lukuun ottamatta [3].

Yhteentoimivuus nojaa standardissa erityisesti kolmeen asiaan: standardoituun tietomalliin, jota käytetään järjestelmän osien välillä vaihdettavan tiedon kuvaamiseen; IED-laitteiden sekä koko järjestelmän ominaisuuksien ja konfiguraation kuvaamiseen määritettyyn kieleen, joka mahdollistaa tietojen vaihdon eri työkalujen välillä; ja tietoliikennepalveluihin, jotka mahdollistavat laitteiden välisen tiedonvaihdon. [2] Tietoliikennepalvelut perustuvat abstraktiin malliin, joka mahdollistaa niiden toteuttamisen millä tahansa vaatimukset täyttävällä tietoliikenneprotokollalla [4]. Tämä mahdollistaa mukautumisen tietoliikennetekniikan kehitykseen sekä standardin hyödyntämisen uudenaikaisilla tavoilla, mikä voi edellyttää erilaisia tietoliikennepalveluita [2].

1.1 Työn tausta ja tavoitteet

Nykyisellään IEC 61850 –toteutus sähköasematoimituksissa rakentuu usein laite kerrallaan ja mahdollisesti eri henkilöiden toimesta, jolloin toteutuksen kokonaiskuva saattaa jäädä epäselväksi. Ratkaisu tähän ongelmaan voisi olla standardissa määritelty suunnitteluprosessi (engl. *engineering process*), jota ei ole kuitenkaan laajamittaisesti hyödynnetty. Prosessi kattaa myös sähköasema-automaatiojärjestelmän tietoliikenteen konfiguroinnin, joten tarkempi käänös voisi olla esimerkiksi suunnittelu- ja konfigurointiprosessi. Yhtenä rajoitteena – ehkä yleisemminkin standardissa määriteltyjen ominaisuuksien käyttöönotolle – on pidetty saatavilla olevien työkalujen puutteellisuutta [5]. Tämä niin kutsuttu *top-down*-prosessi mahdollistaa sähköaseman spesifikaatioiden

esittämisen virtuaalisessa muodossa, joka hyödyttää erityisesti omaisuudenhaltijaa. Lisäksi se parantaa suunnitelmien ja konfiguraatioiden uudelleenkäytettävyyttä. [6] Suunnitteluvaiheessa tehdyillä valinnoilla on myös mahdollista tehostaa järjestelmän koetusta ja käyttöönottoa [7].

Diplomityön tavoitteena on arvioida standardin osassa 6 esitetyn suunnitteluprosessin soveltuvuutta nykyaikaiseen laitteistoriippumattomaan sähköasematoteutukseen sekä muodostaa yhtenäinen toimintatapa IEC 61850 –toteutusten suunnitteluun yhtiön sisällä. Keskeistä tavoitteen saavuttamisen kannalta on arvioida ja löytää soveltuva kolmannen osapuolen suunnittelutyökalu, joka mahdollistaa suunnitteluprosessin toteuttamisen standardissa kuvatulla tavalla. Vaikka suunnitteluprosessin hyödyntäminen osoittautuisi vielä epäkäytännölliseksi tai mahdottomaksi, sopivan suunnittelutyökalun löytyminen voisi olla hyödyllistä myös muista syistä. Tällä hetkellä projekteissa IEC 61850 –järjestelmän konfiguroinnissa käytettävä työkalu on yleensä valmistajakohtainen ja valitaan sen mukaan minkä valmistajan laitteita projektissa käytetään. Samassa projektissa voi olla käytössä tämän vuoksi useita eri työkaluja. Tämä johtaa siihen, että automaatiojärjestelmän konfiguroinnista ja käyttöönotosta vastaavien henkilöiden on hallittava useita eri työkaluja. Useilla eri työkaluilla toteutettu järjestelmä voi hankaloittaa myös järjestelmän ylläpitoa ja laajentamista tulevaisuudessa. Laitteistoriippumaton työkalu, jota voitaisiin hyödyntää kaikissa sähköasema-automaatioprojekteissa vähentäisi tarvetta opetella eri työkalujen käyttöä ja ylläpitää osaamista. Työn avulla pyritään saamaan lisätietoja edellä mainituista työkaluista ja niiden soveltamisesta laitteistoriippumattoman sähköasematoimituksen suunnittelussa. Lisäksi työn tavoitteena on vastata standardin kehityksen ja yleistymisen mukanaan tuomiin osaamistarpeisiin.

Käytännön tavoitteiden lisäksi työssä pyritään löytämään vastaus seuraaviin tutkimusongelmiin:

- Mitä ovat uuden sukupolven työkalut ja työtavat IEC 61850 –sähköasematoimitusten suunnittelussa?
 - a. Millaisia rajoitteita erilaiset projektit ja asiakkaat asettavat työkaluille ja suunnittelumenetelmille?
 - b. Millaisia rajoitteita käytetyt laitteet asettavat työkaluille ja suunnittelumenetelmille?
- Millä tavoin uuden sukupolven työkalut ja työtavat ovat hyödynnettävissä laitteistoriippumattomassa IEC 61850 –toteutuksessa?

- a. Miten uudet suunnittelumenetelmät ja työkalut ovat hyödynnettävissä nykyaikaisissa toteutuksissa?
- b. Miten uusia suunnittelumenetelmiä ja -työkaluja voidaan hyödyntää tulevaisuudessa?
- c. Mitä ovat uusien menetelmien edut ja haitat vanhoihin nähden?

IEC 61850 –standardisarjaa käsitteleviä diplomitöitä on tehty aiemmin useita kuten esimerkiksi [8], [9] ja [10], mutta niissä ei ole niinkään keskitytty suunnitteluprosessiin tai siihen liittyviin työkaluihin. Aihetta on jonkin verran sivuttu kansainvälisen sähkövoimajärjestelmiin keskittyvän järjestön CIGRE:n (*Conseil International des Grands Réseaux Électriques*) julkaisuissa kuten [5] ja [6], myös muita aiheeseen liittyviä julkaisuja [7], [11], [12] on tehty muutamia. Suunnitteluprosessiin liittyvää kehitystyötä on viime vuosina tehty myös osana EU:n järjestelmävastaavien kantaverkkoyhtiöiden yhdistyksen ENTSO-E:n (*European Network of Transmission System Operators for Electricity*) OS-MOSE-projektia (*Optimal System-Mix Of flexibility Solutions for European electricity*) [13]. Tässä diplomityössä aihetta pyritään lähestymään erityisesti sähköasematoimitusten järjestelmäintegraattorin näkökulmasta. Suomessa järjestelmäintegraattorin rooli on useimmiten sähköasemaurakoitsijalla tai laitevalmistajalla, verkkoyhtiöiden keskittyessä pääasiassa omaisuudenhallintaan.

1.2 Työn rakenne ja tutkimusmenetelmät

Aihetta alustetaan ensin luvussa 2 käymällä läpi sähköasema-automaation peruskäsitteitä, määritelmiä, järjestelmän komponentteja sekä mahdollisia kehityssuuntia. Sähköasema-automaatiojärjestelmiä tarkastellaan etenkin siirtoverkkojen ja suurjännitteisten jakeluverkkojen näkökulmasta, mutta useimmat asiat pätevät myös keskijännitetasolla. Käsitteiden ja määritelmien avaamisen lisäksi tällä pyritään taustoittamaan IEC 61850:aan johtaneita tekijöitä sekä selkeyttämään standardisarjan tuomia hyötyjä vanhoihin menetelmiin nähden. Luvussa 3 käsitellään tarkemmin IEC 61850 –standardisarjaa etenkin sähköasematoimitusten näkökulmasta. Luku 4 käsittelee ohjelmistojen valintaa. Tätä varten tutustuttiin ohjelmistojen valintaa aikaisemmin käsitelleisiin diplomitöihin, kuten [14] ja [15] sekä aiheesta tehtyihin tutkimuksiin. Luvussa käydään läpi ohjelmiston valinnassa käytetyt menetelmät sekä arviointikriteerit, joiden perusteella valittiin kaksi ohjelmistoa tarkempaa empiiristä tutkimusta varten. Valittuja ohjelmistoja käsitellään luvussa 5 ja tuloksia luvussa 6.

2. SÄHKÖASEMA-AUTOMAATIOJÄRJESTELMÄT

CIGRE:n B3 tutkimuskomitean teknillisessä raportissa *The Impact of New Functionalities on Substation Design* [16] on korostettu tietoliikennetekniikan merkitystä sähköasema-automaation kehityksessä. IEC 61850 –standardin on arvioitu olevan ehkä huomattavin yksittäinen muutos sähköasemilla pitkään aikaan. Muutokset tuovat mukanaan myös haasteita, kuten uudenlaisia osaamistarpeita. Osaamistarpeista on eritelty monitoimisten mikroprosessoripohjaisten releiden asettelussa vaadittava ohjelmointiosaaminen sekä automaatiojärjestelmien ylläpidossa tarvittava yleinen IT-osaaminen. Nopeasti kehittyvä tietoliikenne- ja mikroprosessoritekniikka voi johtaa myös nopeammin sähköasemien toisiojärjestelmissä käytetyn tekniikan vanhenemiseen. Lisäksi esimerkiksi ohjelmistojen versionhallinnan, *firmware*-päivitysten ja tietoturvan on katsottu aiheuttavan haasteita. IEC 61850 –standardin käyttöönotto edellyttää myös, että verkko-yhtiöt määrittelevät omat vaatimuksensa järjestelmien toteutukseen liittyen. [16] Teknillinen raportti on julkaistu vuonna 2009 ja siinä esitetyt havainnot perustuvat vuonna 2004 toteutettuun kyselyyn sekä komitean jäsenten kokemukseen. Vaikka uusien sähköasemien toisiojärjestelmissä käytetään edelleen paljon tavanomaista tekniikkaa ja IEC 61850 –standardin kaikkia ominaisuuksia ei hyödynnetä yleisesti välttämättä vielä monessakaan paikassa, voidaan raportissa esitetyt havainnot pitää melko hyvin paikkaansa pitävinä. Suojareleiden asettelu ei välttämättä vaadi merkittävää ohjelmointiosaamista, mutta se on varmasti monimutkaistunut varhaisempiin laitesukupolviin nähden. Sen sijaan tietoverkkotekniikan lisääntyminen sähköasemilla on johtanut siihen, että erilaisista tietoliikenneanalysointilaitteista on tullut esimerkiksi sähköasemien käyttöönottotestauksissa lähestulkoon välttämättömiä työkaluja.

Tässä luvussa on pyritty käymään läpi sähköasema-automaatiojärjestelmien tärkeimmät komponentit ja niiden käyttötarkoitukset sekä esittelemään mahdollisia tulevaisuuden kehityssuuntia. Suojausjärjestelmää ei aina lueta osaksi sähköasema-automaatiota ja sitä käsitellään usein täysin erillisenä kokonaisuutena. Suojaus ja ohjaus on kuitenkin tänä päivänä yleensä jollain tapaa integroitu tai ne vähintään keskustelevat tietoliikenteen välityksellä, joten niitä ei voida enää käsitellä täysin itsenäisinä teknologioina [17]. IEC 61850 ei myöskään erottele näitä toisistaan, vaan standardin määritelmän mukaan sähköasema-automaatiojärjestelmä on järjestelmä, joka suojaa, ohjaa sekä valvoo sähköasemaa [18]. Tämän vuoksi sähköasema-automaatiolla tarkoitetaan tässä

yhteydessä kokonaisuutta, joka käsittää sekä sähköaseman ohjaus- että suojausjärjestelmät.

2.1 Suojausjärjestelmä

Suojausjärjestelmä koostuu standardin SFS-IEC 60050-448 määritelmän mukaan suojausjärjestelmästä, mittamuuntajista, johdotuksista, laukaisupiireistä, tehollisista ja mahdollisesti myös tietoliikennejärjestelmistä. Suojausjärjestelmä voi sisältää suojausperiaatteesta riippuen koko suojausalueen tai ainoastaan sen yhden pään. Myös jälleenykyntäautomatiikka luetaan osaksi suojausjärjestelmää mutta katkaisijoita ei. [19, s. 335], [20]

Suojareleet ovat laitteita, jotka tarkkailevat sähkövoimajärjestelmän tilaa, ja vian havaitessaan antavat laukaisukäskyn katkaisijalle vikaantuneiden komponenttien irtikytkemiseksi. Tällä pyritään siihen, että viasta aiheutuvat haitat pysyvät mahdollisimman vähäisinä. Suojareleet mittaavat verkon sähköisiä suureita yleensä mittamuuntajien välityksellä, jotka muuntavat ensiöarvot releille sopivaan muotoon. Tästä poikkeuksen muodostavat primäärireleet, joiden käyttö on kuitenkin tänä päivänä vähäistä. [21, s. 13,15] Suojareleet voidaan jakaa käytetyn tekniikan mukaan eri sukupolviin ja usein puhutaan sähkömekaanisista, staattisista ja mikroprosessoripohjaisista releistä. Tämä vaihtelee kuitenkin lähteittäin ja lähteestä riippuen saatetaan käyttää tarkempaa jaottelua.

Ensimmäiset suojareleet olivat toimintaperiaatteeltaan sähkömekaanisia ja niiden toiminta perustui joitain poikkeuksia lukuun ottamatta sähkömagneettisten vuorovaikutusten mekaanisiin komponentteihin kohdistamiin voimiin [22, s. 31]. Sähkömekaanisia releitä on ollut käytössä pitkään, ne ovat verrattain halpoja ja luotettavia [23, s. 60], mutta koska ne sisältävät liikkuvia osia, ne vaativat säännöllistä huoltoa. Sähkömekaaniset releet saavat tarvitsemansa energian mittamuuntajista, eivätkä näin ollen tarvitse erillistä tehollähdettä, mutta niiden taakka voi olla kuitenkin tämän vuoksi kymmeniä voltiampeereita. [21, s. 22]

Staattisten releiden valmistaminen tuli mahdolliseksi puolijohdeteknologian kehittymisen myötä [22, s. 40]. Elektroniputkia hyödyntäneitä releitä oli valmistettu jo tätä ennen, mutta ne eivät olleet yhtä luotettavia kuin sähkömekaaniset releet ja vaativat enemmän huoltoa [23, s. 99]. Staattisissa releissä haluttu vaste saadaan aikaan sähkömekaanisista releistä poiketen ilman mekaanista liikettä. Ne ovat sähkömekaanisiin releisiin verrattuna herkempiä ja nopeampia sekä vaativat vähemmän huoltoa. [23, s. 66–67]

Koska ne saavat käyttämänsä energian erillisestä teholähteestä, myös niiden taakka on pienempi. Yhdellä staattisella releellä on lisäksi mahdollista toteuttaa useita suojaustoimintoja. [21, s. 22–25] Staattiset releet ovat kuitenkin herkkiä sähköasemaolosuhteissa esiintyville ylijännitteille sekä muille ympäristötekijöille ja etenkin ensimmäiset relemallit kärsivät usein ympäristöolosuhteiden aiheuttamista rikkoontumisista [22, s. 40].

Ensimmäiset mikroprosessoripohjaiset releet tulivat markkinoille vuonna 1979. Niiden kehitykseen vaikuttivat edistykset suurtiheysintegroinnissa (engl. *Very Large-Scale Integration, VLSI*) ja ohjelmistotekniikassa. Varhaiset mikroprosessoripohjaiset releet eivät vielä toteuttaneet monimutkaisia suojausalgoritmeja eikä niiden suorituskyky ollut kovin hyvä. Sen sijaan ne hyödynsivät analogisen ja digitaalisen tekniikan yhdistämistä, toteuttivat perustoiminnallisuuksia ja olivat kustannustehokkaita. Hyväksi todettuja laitteita vastaan kilpailleiden mikroprosessorireleiden taloudellisuudella ja lisääntyneellä toiminnallisuudella oli luultavasti merkittävin vaikutus niiden hyväksyntään. 1980-luvun lopulla esitellyt monitoimireleet vähensivät tuote- ja asennuskustannuksia huomattavasti. Mikroprosessoripohjainen tekniikka mahdollistaa useita toimintoja, jotka eivät olleet saatavilla aiempien sukupolvien suojausreleissä, kuten useat asetteluryhmät, häiriötallenteet, ohjelmoitavan logiikan, itsevalvonnan sekä viestinnän eri releiden ja ohjausjärjestelmän välillä. Niiden taakka on myös varhaisempiin teknologioihin nähden huomattavasti pienempi. [24]

Uusien ominaisuuksien myötä releistä on tullut monimutkaisempia. Useita sähkömekaanisten tai staattisten releiden toimintoja korvaavat monitoimiset releet sisältävät ohjelmoitavia toimintoja, joka tekee niistä joustavampia. Releet sisältävät kuitenkin huomattavan määrän asetteluita, minkä vuoksi asetteluiden hallinta ja testaus voi olla haastavaa. Asetteluiden hallintaan onkin saatavilla tätä tarkoitusta varten kehitettyjä ohjelmistoja. Releitä koestettaessa toimintoja voidaan joutua poistamaan käytöstä, jolloin on varmistuttava siitä, että oikeat asettelut otetaan uudestaan käyttöön, kun rele on koestettu. [24] Asetteluiden suuren määrän vuoksi niihin voi tulla myös helpommin virheitä ja releiden monimutkaisuudesta johtuen virhelaukaisuja esiintyy enemmän kuin muilla reletyypeillä [19, s. 345].

Varhaisempiin relesukupolviin verrattuna mikroprosessoripohjaisten releiden kustannukset ovat toimintoa kohden pienemmät, koska niiden komponentit, tuotantolaitteet ja -tekniikat ovat edullisempia. Mikroprosessoripohjaisten releiden elinkaari on kuitenkin riippuvainen niissä käytettyjen kaupallisesti saatavilla olevien komponenttien elinkaaresta. Sama ongelma koskee lisäksi myös staattisia releitä. Mikroprosessoripohjaisten

releiden elinkaarta lyhentää osaltaan myös jokaisen sukupolven myötä lisääntyvä toiminnallisuus, joka tekee laitteista nopeammin vanhentuneita ja vaikeuttaa osaamisen ylläpitoa. Laitteiden lisäksi myös niiden ohjelmistot päivittyvät, jonka vuoksi niiden versioista on pidettävä kirjaa. [24] Releiden ohjelmistoissa voi olla myös ohjelmointivirheitä [19, s. 345].

2.2 Ohjausjärjestelmä

Sähkövoimajärjestelmää hallitaan verkon käytön toimesta suorien valvonta- ja ohjaustoimenpiteiden avulla. Tällä tarkoitetaan järjestelmän tehotasapainon ylläpitämistä sekä siirron ohjaamista toimimaan mahdollisimman käyttövarmasti ja taloudellisesti. Ohjaus- ja valvontatoimenpiteet voidaan toteuttaa joko paikallisesti tai kaukokäyttöjärjestelmän avulla. Nämä toiminnot mahdollistetaan sähköaseman ohjausjärjestelmällä, joka sisältää tietoliikenne-rajapinnat eri laitteiden ja järjestelmien välillä. [19, s. 385–386] Ohjausjärjestelmä voi vastata myös tahdistuksista, jännitteen- ja kuormansäädöstä sekä lukuksista [25, s. 776]. Ennen kaukokäyttöjärjestelmien yleistymistä, sähköasemat olivat tyypillisesti miehitettyjä, jolloin asemalla toiminut henkilökunta valvoi järjestelmää paikallisesti ja yhteydenpito keskusvalvomoon tapahtui puhelimitse [26, s. 15-2].

Sähköaseman ohjausjärjestelmän käsitettä ei ole suojausjärjestelmän tavoin määritelty standardissa. Samalla tavoin määriteltynä ohjausjärjestelmän voitaisiin katsoa koostuvan ainakin kaukokäytön ala-asemasta, HMI:sta (*Human Machine Interface*), johdotuksista, teholähteistä sekä tietoliikennejärjestelmistä. Lisäksi ohjausjärjestelmään voi kuulua erilaisia apureleitä, mitta-arvomuuntimia ja IED-laitteita. Sähköisten suureiden mittaukseseen ohjausjärjestelmä hyödyntää samoja mittamuuntajia kuin suojausjärjestelmä.

Kaukokäytön ala-asema eli RTU (*Remote Terminal Unit*) on ollut perinteisesti keskeinen osa sähköaseman ohjausjärjestelmää. RTU:n pääasiallinen tarkoitus on välittää tietoa voimajärjestelmän tilasta valvomoon ja toteuttaa valvomosta lähetetyt ohjauskäskyt. RTU sisältää tätä tarkoitusta varten vähintään tietoliikenneliitännän ja I/O-liitännät. Primääriprosessiin lisäksi sen avulla voidaan valvoa myös muita sähköasemaan liittyviä tietoja. [26, s. 7–10] RTU:sta saatetaan käyttää myös nimitystä *data concentrator* [26, s. 7–11] tai *gateway* [25], jos se ei sisällä lainkaan I/O-liitäntöjä ja kerää tietoja ainoastaan väylän kautta. Useimmiten samaa laitetyyppiä voi kuitenkin käyttää molempiin tarkoituksiin, poikkeuksena ehkä ohjelmistopohjaiset toteutukset, joita voidaan ajaa missä tahansa vaatimukset täyttävässä tietokoneessa. Tässä työssä termejä ei erotella toisistaan ja jatkossa käytetään nimitystä ala-asema tai RTU toteutustavasta riippumatta.

Nykyisin on yleisempää toteuttaa järjestelmän I/O-liittynät RTU:n sijaan hajautetusti erillisillä kenttäohjausyksiköillä (engl. *Bay Control Unit, BCU* tai *bay controller*). Kenttäohjausyksikkö voi vastata kentän ensiolaitteiden ohjauksista, tahdistuksista sekä lukituksista ja siihen voidaan tuoda kenttään liittyvät mittaukset [26, s. 7-12]. Lähteen [3] mukaan kenttäohjausyksikkö voi myös kerätä tietoja kentän muilta IED-laitteilta kenttäkohtaisen väylän kautta. Tämä ei ole ainakaan tänä päivänä yleinen toteutustapa vaan kaikki IED-laitteet kommunikoivat suoraan ylemmän tason laitteiden kuten HMI:n tai RTU:n kanssa. Sähköaseman yhteiset tilatiedot, ohjaukset, merkinannot, hälytykset ja mittaukset voidaan hajautetussa järjestelmässä tuoda tätä tarkoitusta varten varatuille kenttäohjausyksiköille. Kenttäohjausyksiköiden sijaan ohjaustoimintoja saatetaan sisällyttää myös suojaroleisiin.

Sähköaseman paikallisohjaus voidaan toteuttaa HMI-tietokoneella, joka koostuu usein keskusyksiköstä, näytöstä, näppäimistöä ja hiirestä [3]. Näiden lisäksi tietokonepohjainen HMI vaatii usein erillisen HMI-ohjelmiston. HMI:lla voidaan tarkoittaa myös mikroprosessoripohjaisen releen etupaneelia [3]. Sähköaseman HMI voi koostua myös valvomohuoneeseen sijoitetusta kytkimistä, lampuista ja mittareista koostuvasta mimiikka-aulusta sekä hälytyskeskuksesta, joihin tiedot on viety johdottamalla. Tällaiset järjestelmät olivat tavanomaisia ennen tietokonepohjaisten järjestelmien yleistymistä ja niitä voi olla edelleen käytössä. [25, s. 776–777], [27, s. 142–143]

2.3 Tietoliikenneprotokollat

Kaukokäyttöprotokollat [26, s. 15-4] ja sähköaseman sisäiset väyläprotokollat [3, s. 24-5], olivat aluksi useimmiten laitevalmistajien kehittämiä suljettuja protokollia. Jos eri valmistajien laitteita haluttiin käyttää samassa järjestelmässä, oli sähköasema-automaatiojärjestelmässä käytettävä joko protokollamuuntimia tai ala-aseman oli tuettava useita protokollia [3]. Protokollien standardointiin johti lopulta halu saavuttaa yhteentoimivuus eri laitevalmistajien laitteiden välillä [28, s. 129].

Avoimet IEC 60870-5-101, IEC 60870-5-104 ja DNP3 ovat tänä päivänä yleisimmät sähköverkon kaukokäytön protokollat. IEC 60870-5 –standardisarja kehitettiin vuosina 1988–2000, jonka jälkeen joitain standardisarjan osia on vielä päivitetty. DNP3 (*Distributed Network Protocol*) perustuu IEC 60870-5 –standardisarjan osiin 1–3 ja se kehitettiin vastaamaan markkinoiden kysyntään ennen IEC 60870-5-101 valmistumista. Protokollasta tehtiin julkinen vuonna 1993 ja sitä ylläpitää DNP-käyttäjäryhmä. IEC 60870-5 –protokollat ovat yleisesti käytössä Euroopassa, kun taas DNP3 on yleisesti

käytössä useimmissa englanninkielisissä maissa sekä paikoissa, joissa pohjoisamerikkalaisilla laitevalmistajilla on merkittävää vaikutusvaltaa. [29, s. 58-2–58-4]

Sähköasemien sisäisessä sarjamuotoisessa tietoliikenteessä on käytetty avoimia IEC 60870-5-103, DNP3 ja Modbus protokollia [3]. IEC 60870-5-103 on IEC 60870-5 –standardisarjaan kuuluva standardi, joka mahdollistaa IED-laitteiden välisen yhteentoimivuuden sähköaseman sisällä [28, s. 138]. Se pohjautuu saksalaiseen VDEW-tietoliikennestandardiin ja muiden IEC 60870-5 –standardisarjan protokollien tavoin se on ollut suosittu Euroopassa. Modbus on vuonna 1979 julkaistu, alun perin ohjelmoitavia logiikoita varten suunniteltu protokolla. Sekä Modbus että DNP3 toimivat myös Ethernetin päällä. [3] Muita käytettyjä protokollia ovat ABB:n 1980-luvulla kehittämä suljettu SPA-protokolla, Echelonin kehittämä avoin LON ja Profibus [30], [31].

2.4 Mittaukset

Sähköasema-automaatiojärjestelmä kerää voimajärjestelmän toimintaan liittyvää mitaustietoa johtolähdöistä, muuntajista, kiskoista ja generaattoreista. Myös energiamittaukset ovat tärkeitä kaupallisista syistä. Sähkötekniisten suureiden mittaus tapahtuu mittamuuntajien välityksellä, joiden ulostulo muunnetaan edelleen joko digitaaliseen muotoon tai analogiseksi virta- tai jänniteviestiksi IED-laitteissa, mitta-arvon muuntimissa tai mittareissa. Virta- ja jänniteviestit voidaan muuntaa edelleen digitaaliseksi kaukokäytön ala-asemassa tai muussa ohjauslaitteessa. [26, s. 6-5]

Mittamuuntajien tarkoitus on erottaa mittaussiipi ensiöpiiristä galvaanisesti, muuttaa mitta-alaa ja mahdollistaa mitta- ja suojalaitteiden standardointi ennalta määrättyihin mittaussarvoihin, suojella mittareita ylikuormitukselta ja mahdollistaa mittareiden sekä releiden sijoitus pois mittausta paikalta. Mittamuuntajien toiminta perustuu usein sähkömagneettisen induktion, mutta jännitteenmittauksessa voidaan käyttää myös kapasitiivisia jännitemuuntajia tai virtamuuntajiin sijoitettuja kapasitiivisia ulostuloja. Virtamuuntajat jaetaan käyttötarkoituksensa mukaan suojaustarkoituksiin ja mittaustarkoituksiin soveltuviin virtamuuntajiin, jotka poikkeavat toisistaan sekä vaatimuksiltaan että mitoitukseltaan. [19, s. 198] Tämän vuoksi virtamuuntaja sisältää sekä mittaussydämen että yhden tai useamman suojaussydämen, joilla on kuitenkin yhteinen ensiö, sijoitustila, ja eristys [21, s. 101–102].

Rautasydämeen perustuvat niin kutsutut konventionaaliset mittamuuntajat eivät ole kehittyneet merkittävästi viime vuosikymmeninä, eivätkä ne ole välttämättä yhteensopivia modernien mittalaitteiden kanssa [32]. Epäkonventionaaliset mittamuuntajat (engl.

Non-Conventional Instrument Transformers, NCIT) ovat digitaalisten sähköasemien kannalta keskeisiä laitteita ja ne mahdollistavat useita hyötyjä konventionaalisiin mittamuuntajiin nähden:

- Avoimesta toisiopiiristä ei aiheudu vaarallisia ylijännitteitä eikä virtamuuntaja voi ylikuumentumisen seurauksena räjähtää.
- Epäkonventionaalinen mittamuuntaja ei tarvitse eristysöljyä tai -kaasua korkeammillakaan jännitetasoilla. Konventionaalisia valuhartsieristeisiä virtamuuntajia on saatavilla ainakin alle 123 kV:n käyttöjännitteelle.
- Epäkonventionaalisen mittamuuntajan ensiöpiirissä aiheuttamat häviöt voivat olla konventionaalisen mittamuuntajan aiheuttamiin häviöihin nähden pienemmät.
- Epäkonventionaaliset mittamuuntajat ovat pienikokoisia ja ne on mahdollista integroida osaksi muita ensiölaiteita.
- Epäkonventionaalinen mittamuuntaja on mahdollista liittää suoraan prosessiväylään ilman erillistä A/D-muunnosta.
- Epäkonventionaalisen mittamuuntajan jatkuva kunnonvalvonta on mahdollista.
- Epäkonventionaalisen mittamuuntajan mittauskaista on laajempi eikä rautasydämen saturaation aiheuttamaa vääristymistä tapahdu. [19, s. 213], [33], [34]

Epäkonventionaalisilta mittamuuntajilta saatu raaka mittaustieto syötetään MU-laitteisiin (*Merging Unit*), jotka toteuttavat signaalinkäsittelyn ja muuntavat ulostulon IEC 61850 –standardin mukaiseksi SV-virraksi (*Sampled Value*). Myös perinteisten mittamuuntajien analogiamittaukset on mahdollista digitalisoida MU-laitteilla SV-virraksi ja siirtää prosessiväylässä sähköasema-automaatiojärjestelmän IED-laitteille. Tällöin ensiöjärjestelmän elinkaari ei ole riippuvainen toisiojärjestelmästä. [3] Merging unitista, jossa on standardoidut analogiset tai digitaaliset sisääntulot, käytetään nimitystä *SAMU (Stand-Alone Merging Unit)* [35].

Epäkonventionaaliset virtamuuntajat tai -anturit voivat olla optisia, jolloin niiden toiminta perustuu yleensä joko Faradayn ilmiöön tai kuituhilan heijastaman valon aallonpituuden muutokseen. Optisten menetelmien lisäksi virranmittaus voi perustua esimerkiksi Rogowskin käämiin tai Hall-anturiin. [32] Epäkonventionaaliset optiset jänniteanturit perustuvat useimmiten Pockelsin ilmiöön. Toinen käytetty optinen mittausmenetelmä perustuu käänteiseen pietsosähköiseen ilmiöön. Muita epäkonventionaalisia

jännitteenmittausmenetelmiä ovat kondensaattori- ja vastusjakajat tai näiden yhdistelmät. [25, s. 402]

2.5 Digitaalinen sähköasema

Tavanomaisen sähköaseman ensiö- ja toisilaitteiden rajapintana toimiva kuparikaapelointi koostuu analogisista ja binäärisistä piireistä sekä jännitteenjakeluista. Kentältä valvomorakennukseen kaapeloitavia ensiölaitteita on tyypillisesti useita. Kaapeleiden määrä voi olla laitteiden sijoittelusta sekä toisiojärjestelmän monimutkaisuudesta riippuen huomattava. Käyttövarmuusvaatimusten takia päälaitteiden lisäksi on asennettava ja johdotettava myös varalaitteita. Energiamittaukset edellyttävät mittamuuntajilta riittävää tarkkuutta, kun taas suojaustarkoituksiin käytettäviltä mittauksilta vaaditaan suurempaa dynaamista aluetta. Tämän vuoksi käytettävät mittamuuntajat ovat monisydämiä. Jokainen kaapeli on myös erikseen suunniteltava, asennettava, koestettava ja ylläpidettävä. Pitkät kaapelointietäisyydet altistavat laitteet sähkömagneettisille häiriöille ja esimerkiksi laiterikkojen aiheuttamille vahingoille. Kaapeliviat voivat aiheuttaa virhetoimintoja ja virtamuuntajien tapauksessa ne ovat myös turvallisuusriski. [36]

Digitaalisella sähköasemalla tarkoitetaan sähköasemaa, joka sisältää IEC 61850 – standardisarjassa määritellyt asema- ja prosessiväylät. Digitaalisessa sähköasemassa prosessista saatavat signaalit välitetään digitaalisessa muodossa valokuitukaapeleita pitkin IED-laitteille ja asematasolla laitteiden välinen tietoliikenne on toteutettu MMS- (*Manufacturing Message Specification*) ja GOOSE-sanomilla (*Generic Object Oriented Substation Events*). [37] Mikäli sähköasema hyödyntää IEC 61850 –standardia täysin, kaikki kuparikaapelit toisilaitteiden jännitteenjakeluita lukuun ottamatta on korvattu valokuitukaapeleilla. Ensiölaitteet voivat liittyä prosessiväylään MU-laitteiden lisäksi IO-yksiköiden (*Input Output Unit, IOU*) avulla, jotka vastaanottavat ohjauksia ja välittävät tilatietoja GOOSE-sanomien avulla. Yhdessä näistä laitteista voidaan käyttää nimitystä PIU (*Process Interface Unit*). [36]

Prosessiväylä tuo mukanaan muutamia oleellisia etuja. Kaapeloinnin määrä pienenee tavanomaiseen järjestelmään verrattuna ja näin ollen myös kaapelointikustannukset pienenevät huomattavasti. Muutokset suojaus- ja ohjaustoimintoihin eivät enää edellytä muutoksia kaapelointiin ja johdotuksiin. Turvallisuus paranee myös tavanomaisten virtamuuntajien tapauksessa, koska avoin virtapiiri on mahdollista saada aikaan ainoastaan virtamuuntajan ja MU:n välissä. Virtamuuntajien saturaatio on myös epätodennäköisempää, koska toisiopiirin resistanssi on merkittävästi pienempi. [36]

Erilaisia digitaalisten sähköasemien pilottiprojekteja on ollut maailmalla useita. Esimerkkejä tällaisista ovat SP Transmissionin FITNESS-projekti Isossa-Britanniassa [38], PECO:n digitaalinen sähköasemaprojekti Yhdysvalloissa [37] ja Fingridin pilottiprojekti Pernoonkoskella [39].

2.6 Keskitetty suojaus- ja ohjausjärjestelmä

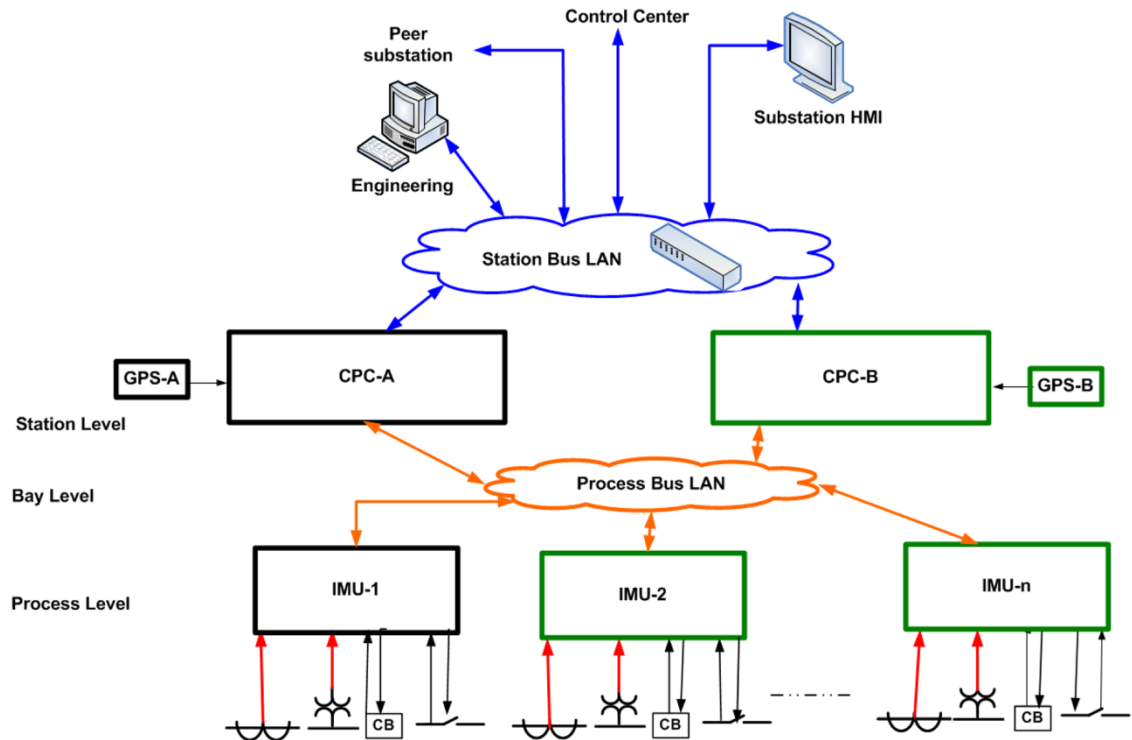
Keskitetty suojaus- ja ohjausjärjestelmä (engl. *Centralized Protection and Control, CPC*) on IEEE:n (*Institute of Electrical and Electronics Engineers*) sähkövoimajärjestelmän releistyskomitean K15-työryhmän määritelmän mukaan korkeatehoisesta laskenta-alustasta koostuva järjestelmä, joka kykenee tarjoamaan suojaus-, ohjaus-, valvonta-, viestintä- ja omaisuudenhallintatoimintoja keräämällä toimintojen vaatimat tiedot käyttämällä nopeita, aikatahdistettuja mittauksia sähköaseman sisällä. Käsitteenä keskitetty suojaus- ja ohjausjärjestelmä on vanha ja K15-työryhmän raportin mukaan ensimmäinen ehdotus tällaisesta järjestelmästä on julkaistu 1969 ja ensimmäinen pilotti toteutettu 1972. Varhaiset kokeilut keskittyivät kuitenkin yleisesti digitaaliseen suojaukseen. Erityisesti keskitettyyn suojaukseen ja ohjaukseen painottuvia kokeiluja alettiin toteuttamaan 1980- ja 1990-lukujen taitteessa. [40]

Keskitetty suojaus- ja ohjausjärjestelmä tarjoaa perinteisiin kenttäkohtaisiin ratkaisuihin verrattuna useita etuja. Keskitetyssä järjestelmässä määriteltävien, konfiguroitavien, koestettavien ja ylläpidettävien laitteiden määrä on pienempi. Perinteisessä järjestelmässä jokainen suojarule sisältää lukuisia asetteluita, minkä lisäksi laitteiden firmware-versioita on seurattava ja päivitettävä säännöllisesti. Laitemäärän pienentyminen helpottaa järjestelmän hallintaa ja ylläpitoa myös näiltä osin. Sähköaseman koosta riippuen, perinteistä järjestelmää luotettavamman keskitetyn järjestelmän suunnittelu ja konfigurointi voi olla kustannustehokkaampaa. Keskitetty järjestelmä sisältää myös vähemmän kyberhyökkäyksissä hyödynnettäviä liityntäpisteitä, mikä voi tehdä siitä tavansa omaista järjestelmää tietoturvallisemmän ratkaisun. Keskitetty järjestelmä vaatii kuitenkin kokonaan uudenlaisen lähestymistavan suojaus- ja ohjausjärjestelmien toteutukseen, käyttöön ja ylläpitoon. [40]

Raportin [40] mukaan nykyaikaisen keskitetyn suojaus- ja ohjausjärjestelmän toteuttamisen mahdollistavat pitkälti samat teknologiat, joita käytetään digitaalisilla sähköasemilla. Keskitetyssä suojaus- ja ohjausjärjestelmässä käytetty laskenta-alusta voi kuitenkin perustua kaupallisesti ”suoraan hyllyltä” saataviin (engl. *commercial off-the-shelf, COTS*) palvelintietokoneisiin. Tämä helpottaa järjestelmän elinkaaren hallintaa ja takaa käyttäjälle laajemman määrän vaihtoehtoja laitteistotoimittajan valintaan. Yhtenä etuna

on myös se, että järjestelmätoimittajat voivat keskittyä kaupallisesti saatavilla olevilla alustoilla toimivien suojaus- ja ohjaussovellusten kehittämiseen. [40]

Kuvassa 1 on esimerkki keskitetyn suojaus- ja ohjausjärjestelmän mahdollisesta arkkitehtuurista. Suojaus- ja ohjauslaitteet on varmennettu ja ne keskustelevat IMU-laitteiden kanssa prosessiväylän välityksellä. IMU eli *Intelligent Merging Unit* on PIU:n kaltainen laite, joka voi toimia tietoliikenneyhteyksien pettäessä myös varasuojana [40].



Kuva 1. Esimerkki keskitetyn suojaus- ja ohjausjärjestelmän arkkitehtuurista. [40]

Joidenkin sähköasema-automaatiojärjestelmän vähemmän aikakriittisten toimintojen, kuten vianpaikannuksen, toteuttamista pilvipalveluna on esitetty lähteessä [41]. Pilvipalveluja hyödyntämällä olisi mahdollista edelleen vähentää sähköasemalle sijoitettavien toisilaitteiden määrää sekä kasvattaa laskentakapasiteettia [41].

3. IEC 61850 SÄHKÖASEMATOIMITUKSISSA

Tarve tehokkaalle ja standardoidulle laitteiden väliselle tietoliikenteelle on standardin [2] mukaan peräisin integroitujen piirien kehityksen mukanaan tuomasta mahdollisuudesta rakentaa useista IED-laitteista koostuvia automaatiojärjestelmiä. Aluksi IED-laitteiden välinen tietoliikenne toteutettiin suljetuilla laitevalmistajakohtaisilla protokollilla. Jos eri laitevalmistajien IED-laitteet haluttiin saada kommunikoimaan keskenään, tarvittiin tätä tarkoitusta varten erillisiä protokollamuuntimia. Eri laitteiden yhteentoimivuutta tukemaan nähtiin tarpeelliseksi kehittää standardoitu semantiikka, eri protokollien kanssa yhteensopivat abstraktit tietoliikennepalvelut, konfiguraatiokuvaukset ja suunnitteluprosessi. Tästä seuranneen standardointiprosessin tavoitteena on ollut kehittää tietoliikennestandardi, joka täyttää sähköverkon automaatiojärjestelmälle asetetut toiminnallisuuteen ja suorituskykyyn liittyvät vaatimukset, säilyen samalla relevanttina teknologian kehityksestä riippumatta. [2]

IEC 61850 pyrkii muun muassa tarjoamaan mahdollisimman pitkälle olemassa oleviin tietoliikennestandardeihin pohjautuvat tietoliikenneprofiilit, jotka perustuvat avoimiin ja itsekuvautuvia (engl. *self-descriptive*) laitteita tukeviin protokolliin. Itsekuvautuvalla tarkoitetaan, että laite sisältää tiedot omasta konfiguraatiostaan standardoidussa muodossa ja niihin pääsee käsiksi tietoliikennejärjestelmän välityksellä. Standardissa määritellyt tietoliikennepalvelut tulisi pystyä toteuttamaan erilaisilla nykyaikaisilla tietoliikenneprotokollilla. Tietoliikenteen syntaksin ja semantiikan on perustuttava voimajärjestelmään liittyvien dataobjektien käyttöön. Standardi pyrkii myös varmistamaan, että sähköjärjestelmän topologia ja siihen liittyvä informaatio sekä IED-laitteiden välinen tietoliikenne voidaan määrittää koneluettavassa muodossa. [2]

3.1 Standardin historia

Monet standardiin määritellyistä toiminnallisuuksista ovat lähteen [42] mukaan peräisin EPRI:n (*Electrical Power Research Institute*) UCA-projektista (*Utility Communication Architecture*). UCA oli EPRI:n vuonna 1988 käynnistämä projekti, jonka tavoitteena oli edistää verkkoyhtiöille toimitettujen tietokonejärjestelmien yhteentoimivuutta. Projektissa laadittiin vaatimusmäärittely, jonka pohjalta arvioitiin verkkoyhtiöiden tietoliikennetarpeisiin soveltuvia standardeja. Standardeista valittiin joukko protokollia, joiden pohjalta määritettiin tietoliikenneprofiilit UCA:n vuonna 1990 julkaistuun ensimmäiseen versioon. UCA 1.0 hyödynsi seitsemänkerroksista OSI-viitemallia (*Open Systems*

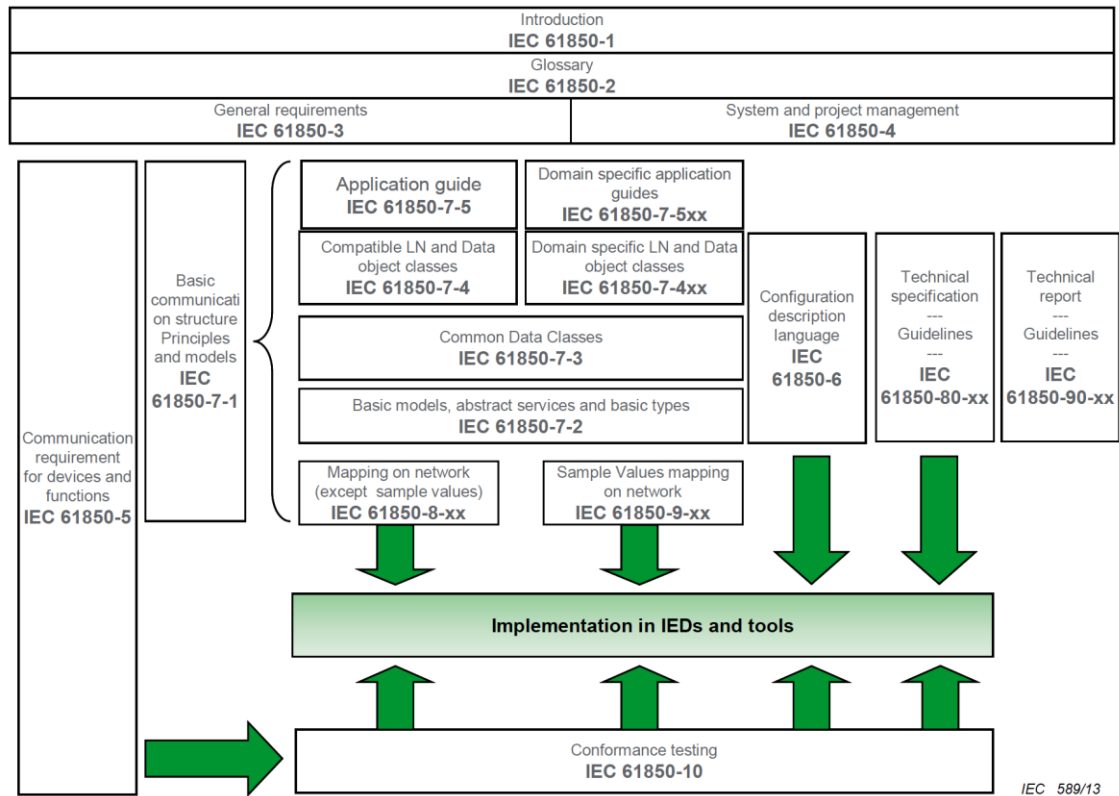
Interconnection), jonka eri kerroksilla toimivat protokollat ja niille määritetyt parametrit muodostavat kyseisen sovellusympäristön tietoliikenneprofiilin. OSI-viitemallin hyödyntäminen mahdollistaa soveltuvan siirtomedian ja linkin valitsemisen tapauskohtaisesti vaikuttamatta sovelluskerroksen toteutukseen. Lisäksi se mahdollistaa myös uusien tietoliikennetekniikoiden hyödyntämisen tulevaisuudessa olemassa olevia toteutuksia muuttamatta. Se käytti sovelluskerroksella MMS-protokollaa useimmissa reaaliaikaisissa tiedonkeruu- ja ohjaussovelluksissa. Protokollien käyttötapaa ei kuitenkaan määriteltä riittävän tarkasti tai ollenkaan, joka yhdessä MMS:n laajuuden ja yleisluontoisuuden kanssa johti luultavasti siihen, että yhteentoimivuusongelmia ei saatu ratkaistua ja UCA:n ensimmäisen version käyttöönotto jäi rajalliseksi. [43]

Osana UCA:n toisen version kehitystyötä luotiin oliopohjaiset mallit sähköasemien ensiö- ja toisilaitteille. Laitteiden toteuttama tietoliikenne on mallinnettu abstrakteilla tietoliikennepalveluilla, joille on määritetty standardoitu kuvaus (engl. *mapping*) MMS-protokollaan. Yhdessä näiden on tarkoitus mahdollistaa laitteiden yhteentoimivuus. UCA 2.0:n tietoliikenneprofiileihin sisällytettiin myös TCP/IP-protokollaperheen (*Transmission Control Protocol/Internet Protocol*) protokollia sekä kolmekerroksinen protokollapino resursseiltaan rajoittuneemmille laitteille ja matalakaistaisille verkoille. [43]

Samaan aikaan kun UCA:n toista versiota kehitettiin, IEC:n TC 57 (*IEC Technical Committee 57*) aloitti IEC 61850 –standardisarjan kehitystyön lähteestä riippuen joko vuonna 1995 [44] tai 1996 [42]. Koska molempien ryhmien tavoitteet olivat samankaltaiset, ne päättivät yhdistää työnsä vuonna 1997. Yhdistymisen seurauksena syntyi nykyinen IEC 61850 –standardisarja, joka sisältää UCA 2.0 spesifikaation lähes kokonaisuudessaan sekä useita lisäyksiä. [42], [45], [46] Standardin pääosat julkaistiin vuosina 2002–2005 ja alkuperäisiä osia on päivitetty vuodesta 2009 eteenpäin. Alun perin standardi käsitti ainoastaan sähköaseman sisäisen tietoliikenteen, mutta se on myöhemmin laajentunut myös sähköasemien ulkopuolelle. [2]

3.2 Standardin rakenne

Standardisarja koostuu useasta osasta ja ajantasainen listaus löytyy IEC:n verkkosivuilta [47]. Standardin eri osat ja niiden väliset yhteydet on esitetty kuvassa 2.

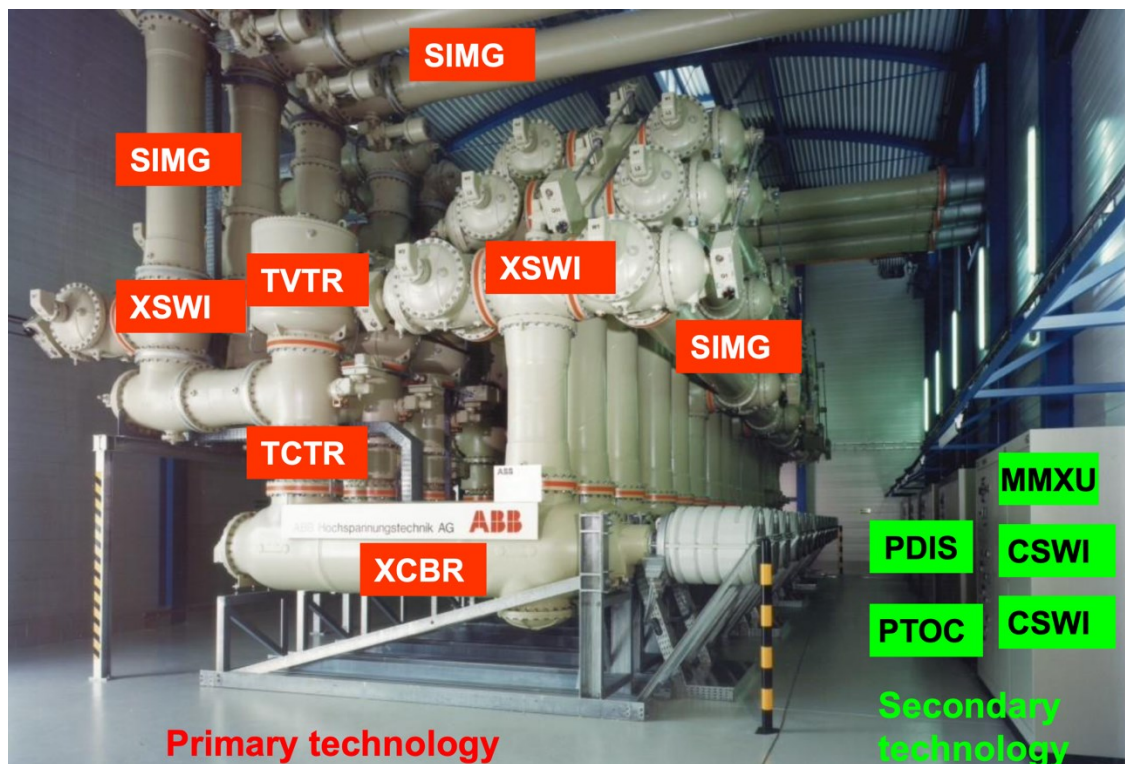


Kuva 2. Standardin osien väliset yhteydet. [2]

Ensimmäinen osa IEC 61850-1 sisältää johdannon ja yleiskatsauksen standardiin, IEC 61850-2 sisältää sanastoa ja termejä [2]. IEC 61850-3 määrittää tietoverkon laatuun, ympäristöolosuhteisiin ja liitännäisiin palveluihin liittyvät yleiset vaatimukset [48]. IEC 61850-4 määrittää automaatiojärjestelmän suunnitteluun, elinkaareen ja laadunvarmistamiseen liittyvät vaatimukset [49]. Osassa 5 puolestaan määritetään automaatiojärjestelmän toiminnot ja näiden tietoliikennevaatimukset [48]. Osassa 6 on määritelty XML-pohjainen (*Extensible Markup Language*) kuvauskieli SCL (*System Configuration Language*), joka mahdollistaa yksittäisen IED-laitteeseen tai koko järjestelmään liittyvien määrittelyiden tallentamisen ja tiedonvaihdon eri työkalujen välillä, sekä miten SCL-tiedostoja tulisi hyödyntää järjestelmän suunnittelussa [50]. Standardisarjassa käytetty tietomalli, tietoliikennearkkitehtuuri ja näihin liittyvät toiminnot on määritelty osissa 7-X [48]. Standardin osista 8-X ja 9-X käytetään nimitystä SCSM (*Specific Communication Service Mapping*). Niissä määritellään standardin aiemmissa osissa kuvattujen tietoliikennepalveluiden toteutus eri tietoliikenneprotokollilla [51], [52]. Osassa 10 on määritelty vaatimukset ja käytännöt laitteiden vaatimuksenmukaisuustestausta varten. 80-X sarjan dokumentit ovat tietoliikenteen toteuttamiseen liittyviä ylimääräisiä spesifikaatioita ja 90-X sarjan dokumentit standardin laajennuksiin ja parannuksiin liittyviä ylimääräisiä raportteja [2].

3.3 Tietomalli

Primääriprosessin osat ja siihen liittyvät toiminnot on mallinnettu standardisarjassa oliopohjaiseen (engl. *object oriented*) tietomalliin [46], joka toimii pohjana kaikelle tiedonvaihdon. Standardin mallinnusmenetelmä perustuu toimintojen hajottamiseen (engl. *decomposition*) pienempiin loogisiksi solmuiksi (engl. *Logical Node, LN*) kutsutuihin osiin, jotka voivat kuvata jotain yksittäistä primääriprosessin komponenttia tai automaatiojärjestelmän toimintoa. [48] Tätä on havainnollistettu kuvassa 3 GIS-kojeiston (*Gas Insulated Switchgear*) avulla. Kuvassa näkyvä neljäkirjaiminen lyhenne on loogisen solmun nimi. Kuvassa punaisella taustalla esitetyt loogiset solmut mallintavat ensiojärjestelmän komponentteja ja vihreällä esitetyt toisiojärjestelmään liittyviä toimintoja. *XSWI* mallintaa erotinta, *XCBR* katkaisijaa, *TVTR* jännitemuuntajaa ja *SIMG* eristekeasun valvontaa. Toisiojärjestelmän toiminnoista *PTOC* mallintaa aikaperusteista ylivirtasuojausta, *PDIS* distanssisuojausta, *MMXU* kolmivaiheista mittausta ja *CSWI* kytkinlaitteiden ohjausta.



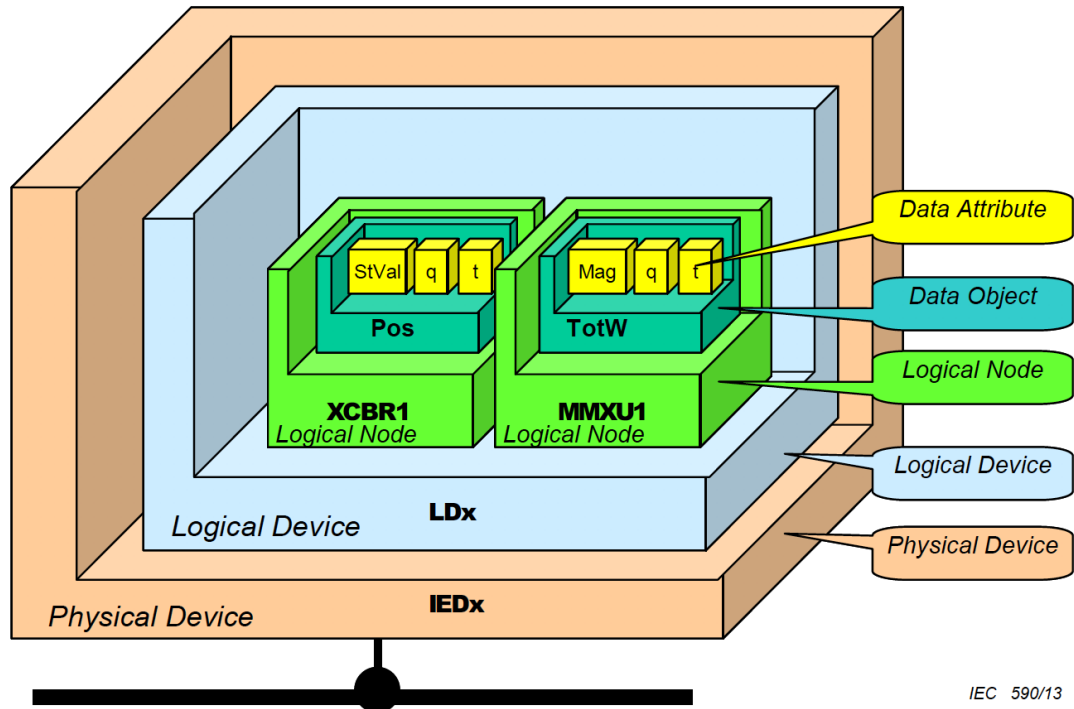
Kuva 3. Loogiset solmut sekä niiden mallintamat komponentit ja toiminnot GIS-kojeistossa. [53]

Loogiset solmut on jaettu ryhmiin niiden käyttötarkoituksen mukaan. Loogisen solmun nimen ensimmäinen kirjain vastaa ryhmän tunnusta, johon looginen solmu kuuluu [54, s. 19]. Standardin editiossa 2.0 [54] määritetyt ryhmät on esitetty taulukossa 1.

Taulukko 1. Loogisten solmujen ryhmittely. [54, s. 19]

Ryhmän tunnus	Ryhmän nimi
A	Automatic control
C	Supervisory control
D	Distributed energy resources
F	Functional blocks
G	Generic function references
H	Hydro power
I	Interfacing and archiving
K	Mechanical and non-electrical primary equipment
L	System logical nodes
M	Metering and measurement
P	Protection functions
Q	Power quality events detection related
S	Supervision and monitoring
T	Instrument transformers and sensors
W	Wind power
X	Switchgear
Y	Power transformer and related functions
Z	Further (power system) equipment

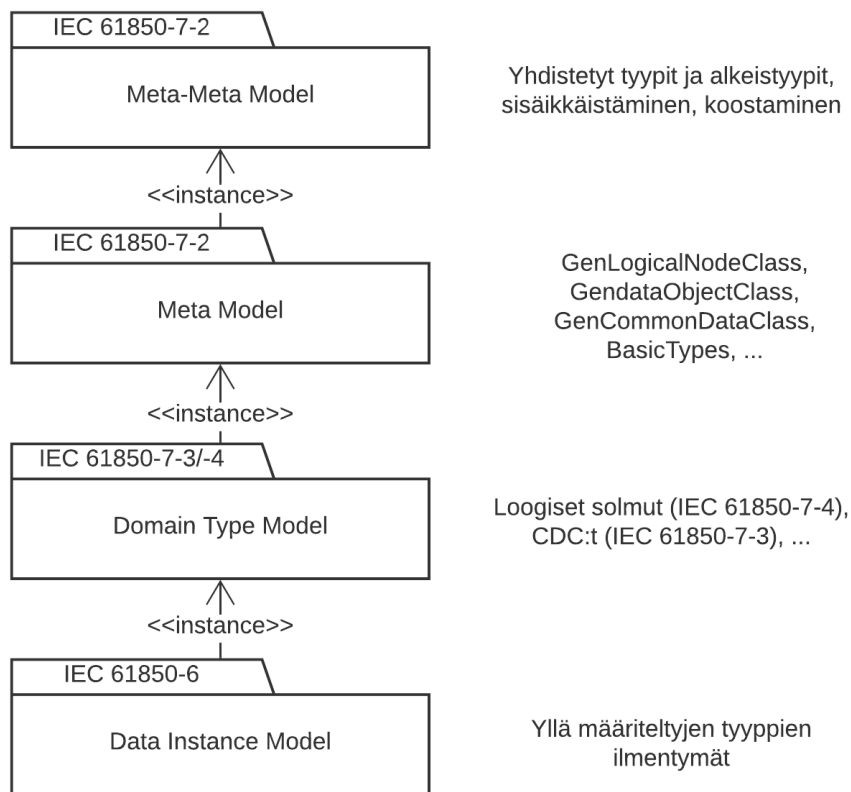
Loogiset solmut koostuvat edelleen pienemmistä osista: dataobjekteista tai -olioista (engl. *Data Object, DO*). Dataobjektit sisältävät joukon data-attribuutteja (engl. *Data Attribute, DA*), joiden avulla ne kuvaavat jotain ominaisuutta tai suuretta, kuten esimerkiksi asentotietoa. Loogisista solmuista koostetaan loogisia laitteita (engl. *Logical Device, LD*), jotka voivat edelleen kuvata jotain suurempaa reaali maailman kokonaisuutta, kuten sähköaseman kenttää. Looginen laite kuuluu aina yhteen fyysiseen IED-laitteeseen eikä se voi olla yhteinen usealle IED-laitteelle, eikä näin ollen voi sisältää loogisia solmuja useasta IED-laitteesta. [48, s. 21–22] Tietomallin rakenne on esitetty kuvassa 4.



Kuva 4. Tietomallin rakenne. [2, s. 24]

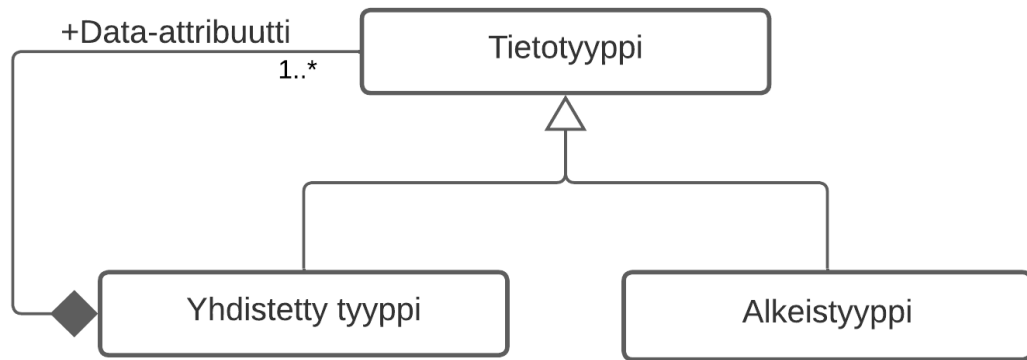
Oliopohjainen mallinnus on ohjelmistokehityksessä laajasti hyödynnetty menetelmä. Oliopohjaisen lähestymistavan on sanottu tekevän standardin kehitystyöstä joustavam-
paa ja yksinkertaistavan suunnittelutehtäviä. Tämä mahdollistaa myös UML:n (*Unified
Modelling Language*) käytön mallien kuvaamisessa, joka helpottaa mallien tulkitse-
mistä. [46] UML tai sitä muistuttavia kaavioita esiintyy etenkin standardin osissa 6 ja 7-
2.

Standardisarjan tietomalli voidaan kuvata myös neljätasoisien käsitelmallin avulla. Tästä
lähestymistavasta käytetään myös nimitystä metamallinnus. Metamallin jokaisen tason
toimijat ovat ilmentymiä vastaavasta ylemmän tason toimijasta. [55] Tämä on esitetty
UML-pakkauskaavion avulla kuvassa 5. Käytännössä ylemmän kerroksen malli määrit-
tää periaatteet, jonka pohjalta alempien kerrosten mallit voidaan muodostaa.



Kuva 5. Metamallin tasot. Perustuu lähteisiin [55], [56].

Metamallin ylin kerros eli meta-metamalli määrittelee hyvin yleisellä tavalla alemman tason malleissa käytettävät tietotyypit. Tietotyyppi voi olla joko yhdistetty tyyppi (engl. *composite type*), joka koostuu muista tietotyypeistä ja joiden avulla voidaan muodostaa sisäkkäisiä rakenteita; tai alkeistyyppi (engl. *primitive type*), joita ei ole mahdollista hajottaa edelleen pienempiin osiin. [55] Meta-metamalli on esitetty UML-notaation avulla kuvassa 6. Aliluokista (*yhdistetty tyyppi* ja *alkeistyyppi*) kantaluokkaan (*tietotyyppi*) osoittava yhtenäinen nuoliviiva kuvaa periytymissuhdetta (engl. *generalization*). Olio-ohjelmoinnissa periytymisellä (engl. *inheritance*) tarkoitetaan, että aliluokka perii kaikki kantaluokkansa ominaisuudet. [57, s. 136] Tässä sillä on yleisemmin kuvattu, että alkeistyyppi ja yhdistetty tyyppi ovat tietotyyppejä. Mustasta vinoneliöstä alkava viiva kuvaa luokkien välistä muodostumissuhdetta (engl. *composite aggregation* tai *composition*) [57, s. 134], [58, s. 55]. Sen avulla on ilmaistu, että yhdistetty tyyppi koostuu yhdestä tai useammasta tyypistä, jotka eivät voi olla olemassa ilman sitä.



Kuva 6. Meta-metamalli. Perustuu lähteeseen [55].

Seuraava taso eli metamalli koostuu luokista, joita käytetään tietomallien ja tiedonvaihdon kuvaamiseen [56, s. 16]. Metamalli määrittää tietomalliin liittyvät konseptit abstraktilla tasolla, rajoittamatta niiden käyttöä mihinkään tiettyyn sovellusalueeseen [55]. Tiedonvaihtoon liittyviä luokkia ja palveluita käsitellään tarkemmin alaluvussa 3.4. Sovellusaluekohtainen *domain type model* määrittää joukon yleisiä dataluokkia (engl. *Common Data Class, CDC*), dataobjekteja ja LN-luokkia, joita käytetään IED-laitteiden tietomallien rakentamiseen [56, s. 21]. Datainstanssimalli (engl. *data instance model*) kuvaa IED-laitteen sisältämät sovellusaluekohtaisten tietotyyppien instanssit ja miten ne on nimetty [55]. Datainstanssimalli kuvataan IEC 61850-6:ssa määritetyn SCL-kielen avulla. SCL-konfiguraatiokieltä ja SCL-tiedostoja käsitellään laajemmin alaluvussa 3.7.

3.4 Tietoliikennepalvelut

Tietoliikenteen toimintaperiaatteet on kuvattu standardissa metamalliin kuuluvien luokkien ja niihin sisältyvien palveluiden avulla, joihin viitataan käsitteellä *abstract communication service interface (ACSI)*. Abstraktiolla tarkoitetaan tässä yhteydessä, että ainoastaan tietoliikennejärjestelmän kautta näkyvä osa fyysisestä järjestelmästä ja toiminoista on mallinnettu, ja tiedonvaihto on määritetty käsitteellisellä tasolla, erillään tietoliikenteen konkreettisesta toteutustavasta [56, s. 11]. Tämä mahdollistaa sovellustoteutuksen erottamisen tietoliikenteestä [2, s. 16,27] ja tietoliikenteen toteuttamisen millä tahansa vaatimukset täyttävällä protokollalla [4].

Tämän työn kannalta on tarpeellista kuvata tarkemmin seuraavat mallit:

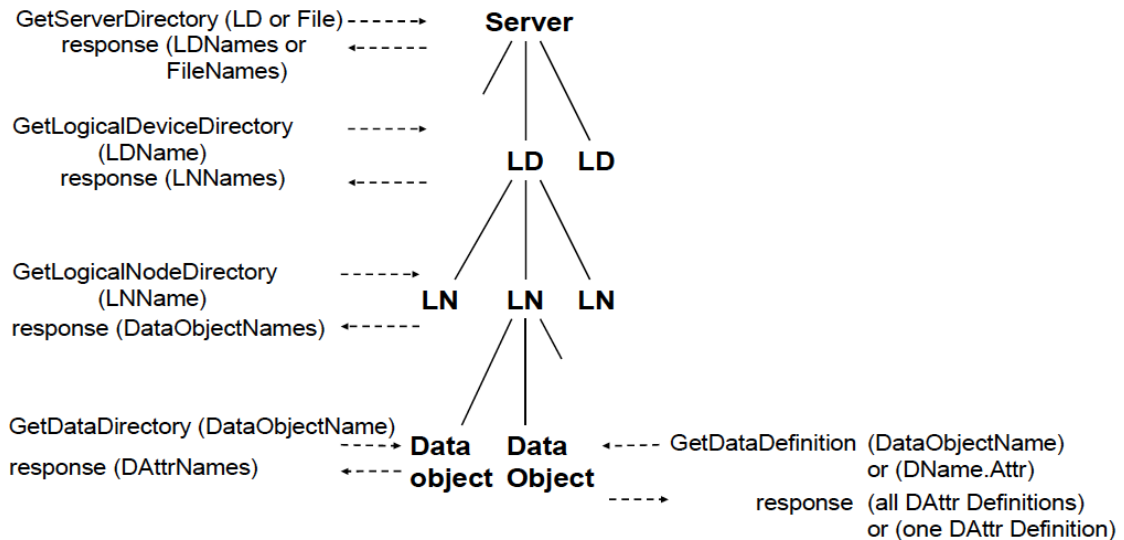
- Dataset-malli (*DATA-SET class model*), joka mahdollistaa dataobjektien ja -attribuuttien ryhmittelyn.

- Raportointi ja lokien keruu –mallit (*REPORT-CONTROL-BLOCK class model* ja *LOG-CONTROL-BLOCK class model*), jotka mahdollistavat tietojen tapahtumapohjaisen lähettämisen, kyselyn ja lokiin kirjaamisen.
- GSE–malli (*Generic Substation Event class model*), joka mahdollistaa tilatietojen lähettämisen julkaisija–tilaaja-periaatteella (engl. *publisher-subscriber*) nopeasti ja järjestelmänlaajuisesti.
- Näytearvojen lähetysmalli (*Transmission of sampled value class model*), joka mahdollistaa näytteistettyjen mittausrvojen lähettämisen julkaisija-tilaaja periaatteella nopeissa sykleissä. [56]

Muita tietoliikennepalveluita kuvaavia malleja ovat:

- Sovellusassosiaatiomalli (*Application association model*), joka määrittää perusteet laitteiden väliselle tietoliikenteelle. Se koostuu kahdesta luokasta, joista kahden osapuolen välinen sovellusassosiaatio (engl. *Two Party Application Association, TPAA*) määrittää kaksisuuntaiseen yhteydelliseen tiedonvaihtoon liittyvät palvelut ja *multicast*-sovellusassosiaatio (engl. *Multicast Application Association, MPAA*) mahdollistaa yksisuuntaisen tiedonvaihdon yhdeltä lähettäjältä eli julkaisijalta usealle vastaanottajalle eli tilaajalle.
- Ohjausmalli (*CONTROL class model*), joka kuvaa esimerkiksi laitteiden ohjaukseen tarvittavat palvelut.
- Asetteluryhmämalli (*SETTING-GROUP-CONTROL-BLOCK class model*), joka määrittää asetteluryhmien vaihtoon ja muokkaamiseen liittyvät palvelut.
- Aika ja aikasykronointi –malli (*Time and time synchronization model*), joka mahdollistaa laitteiden ja järjestelmän aikasykronoinnin.
- Tiedostonsiirtomalli (*File transfer model*), joka määrittää palvelut tiedostojen lähettämiseen ja vastaanottamiseen. [56]

Kaikki ACSI:n mallit ja palvelut toimivat *GenServerClass*-luokan määrittämän palvelininstanssin alla [56, s. 17]. Myös *GenServerClass* sekä muut tietomallinnuksessa käytetyt *GenLogicalDeviceClass*, *GenLogicalNodeClass* ja *GenDataObjectClass* luokat sisältävät tietoliikennepalveluita. Niillä mahdollistetaan standardissa määritetty laitteiden itsekuvaavuus tarjoamalla asiakkaille pääsy kaikkeen palvelimen sisältämään informaatioon [56, s. 30]. Kuvassa 7 on esitetty palvelimen sisältämä hierarkia sekä siihen liittyvät *GetDirectory*- ja *GetDefinition*-palvelut.



Kuva 7. Palvelimen sisältämä hierarkia ja sen kuvaamiseen liittyvät palvelut [56, s. 30].

Data-attribuutteja voidaan luokitella ja suodattaa toiminnallisen rajoitteen (engl. *Functional Constraint, FC*) avulla. Toiminnallinen rajoite määrittää mitä palveluita data-attribuutteihin voidaan kohdistaa. Esimerkiksi tilatietoa kuvaavalla data-attribuutilla *stVal* on toiminnallinen rajoite *ST*. Tämä tarkoittaa, että data-attribuutti voidaan lukea, korvata (engl. *substitute*), raportoida ja kirjata lokiin, mutta sen päälle ei voida kirjoittaa. Järjestettyyn joukkoon dataobjektin sisältämiä data attribuutteja, joilla on sama toiminnallinen rajoite, viitataan käsitteellä toiminnallisesti rajoittunut data (engl. *Functionally Constrained Data, FCD*). Yksittäiseen dataobjektin sisältämään data-attribuuttiin, alidataobjektiin tai alidata-attribuuttiin viitataan käsitteellä toiminnallisesti rajoittunut data-attribuutti (engl. *Functionally Constrained Data Attribute, FCDA*). [56]

Raporteille, lokeille, SV:lle ja GSE:lle on määritetty ohjauslohkot (engl. *control blocks*), jotka on mallinnettu samaan tapaan kuin dataobjektit. Ohjauslohkoja käytetään kuitenkin tietoliikennepalveluiden konfigurointiin, toisin kuin dataobjekteja, jotka toimivat rajapintana sovellustason toiminnolle. [56, s. 70]

3.4.1 Datasetit

Dataset on järjestetty joukko viittauksia toiminnallisesti rajoitettuun dataan tai data-attribuutteihin, joita kutsutaan datasetin jäseniksi. Datasetit ovat tärkeitä tapahtumapohjaisen tiedonvaihdon kannalta: koko tietojoukon lähetys voidaan sitoa missä tahansa yksittäisessä datasetin jäsenessä tapahtuvaan muutokseen. Datasetin jäsenet ja niiden järjestys ovat sekä lähettäjän että vastaanottajan tiedossa. Näin ollen riittää, että

ainoastaan datasetin nimi ja jäsenten sen hetkiset arvot lähetetään, jolloin tietoliikennejärjestelmän kaista hyödynnetään mahdollisimman tehokkaasti. [56, s. 61] Datasetteja voidaan luoda dynaamisesti *CreateDataSet*-palvelun avulla tai ne voidaan konfiguroida staattisesti.

3.4.2 Raportointi ja lokien keruu

Raportointi ja lokien keruu –mallit kuvaavat menetelmät tietojen tapahtumapohjaisen lähettämisen tai tallentamisen IED:lle lokiin myöhempää kyselyä varten. Mallit mahdollistavat tapahtumasekvenssin (engl. *Sequence of Events, SOE*) tallentamisen lokiin tai tapahtumien raportoinnin ajallisesti oikeassa järjestyksessä. Raportteja lähetetään ainoastaan tarvittaessa lähetykselle määritettyjen ehtojen toteutuessa tai asiakkaan käynnistäessä yleiskyselyn. [56, s. 91] Lokeja ei juurikaan hyödynnetä sähköasematoimituksissa, kun taas raportteja käytetään poikkeuksetta aina, jos sähköasemalla on IEC 61850 –asemaväylä.

Raporttien lähetystä hallitaan raporttien ohjauslohkoilla (engl. *Report Control Block, RCB*). Yhteen ohjauslohkoinstanssiin voi yhdistää ainoastaan yksi asiakas kerrallaan, jolloin ohjauslohkoinstanssi lähettää raportteja tälle asiakkaalle eksklusiivisesti. Jotta useampi asiakas voisi vastaanottaa samat arvot, on ohjauslohkosta luotava useita instansseja. Ohjauslohkoja on määritetty kaksi hieman toisistaan poikkeavaa tyyppiä. Puskuroimaton raporttien ohjauslohko (engl. *Unbuffered Report Control Block, URCB*) lähettää raportit puskuroimattomasti. Tämä tarkoittaa sitä, että jos assosiaatiota puuttuu tai yhteys ei ole riittävän nopea, tapahtumia voi kadota. Puskuroitu raporttien ohjauslohko (engl. *Buffered Report Control Block, BRCB*) mahdollistaa tapahtumien lähettämisen puskuroidusti, jotta dataobjektien arvoja ei hukata yhteyden katketessa tai vuonhallinnan rajoittaessa lähettämistä. [56, s. 93] Puskuroitu raporttien ohjauslohko takaa myös tapahtumasekvenssin säilymisen joillain rajoituksilla [48, s. 46]. Puskuroimattomia raportteja käytetään pääasiassa jatkuvien mittaustietojen lähettämiseen ja puskuroituja esimerkiksi erilaisten tapahtumien ja hälytysten lähettämiseen.

Raportointiin verrattuna lokien keruu on ulkoisista sovellusassosiaatioista ja muusta tietoliikenteestä riippumaton toiminnallisuus. Se voi toimia joissain tapauksissa huomattavasti nopeammin kuin ulkoiseen tietokantaan raportoiva tietoliikenneprosessi. Ulkoiset sovellukset voivat noutaa lokin osia erillisen historiatietokannan ylläpitämistä varten. [56, s. 118]

Lokimalli koostuu lokien ohjauslohkosta (engl. *Log Control Block, LCB*) ja varsinaista lokia mallintavasta *LOG*-luokasta. Loki voidaan käsittää rengaspuksurina, joka kirjoittaa vanhimpien arvojen päälle. Asiakkaan näkökulmasta se on kuitenkin lineaarinen puskuri, jonka merkinnät voidaan erottaa *EntryID*- ja *TimeOfEntry*-attribuuteista. *EntryID* on laskuri, jota käytetään lokimerkintöjen tunnistamiseen. Laskuri pyörähtää ympäri, kun sen maksimiarvo on saavutettu. *TimeOfEntry* on aika, jolloin merkintä on tallennettu lokiin. Asiakas voi tehdä kyselyitä lokiin näiden attribuuttien perusteella. Ohjauslohkoilla hallitaan data-attribuuttien tallentamista lokiin. Jokainen ohjauslohko yhdistää jonkin datasetin lokiin ja useammalla ohjauslohkolla on mahdollista kirjoittaa yhteen lokiin. [56, s. 119]

Ehdot raporttien lähettämiseksi ja lokiin kirjaukselle määritetään ohjauslohkojen *TrgOps*-parametreilla, joita ovat:

- *data-change (dchg)* eli arvon vaihtuminen toiseen arvoon
- *quality-change (qchg)* eli laadun muutos
- *data-update (dupd)* eli arvon päivittyminen, myös samaan arvoon
- *integrity* eli syklinen lähetys/kirjaus
- *general-interrogation* eli yleiskysely, joka on tuettu ainoastaan raporteissa. [56]

3.4.3 GSE

GSE-malli määrittää menetelmät I/O-tietojen nopeaan ja luotettavaan, järjestelmänlaajuiseen lähettämiseen. Se mahdollistaa samojen tietojen välittämisen useammalle laitteelle samanaikaisesti multicast- tai *broadcast*-palveluiden avulla. Tätä varten on määritetty kaksi ohjauslohkoluokkaa ja sanomarakennetta. Näistä GOOSE tukee useiden erilaisten datasetteihin järjestettyjen tietojen lähettämistä ja GSSE (*Generic Substation State Event*) mahdollistaa tilatietojen lähettämisen bittipareina. GSSE vastaa UCA 2.0:ssa määritettyä GOOSE-mallia ja se on määritetty vanhentuneeksi (engl. *deprecated*) standardin editiossa 2. [56, s. 131] GSSE:tä ei ole käytetty sähköasematoimittuksissa ainakaan viime vuosina, joten sitä ei käsitellä tässä yhteydessä tarkemmin.

Vastaanottava laite saa GOOSE-sanomien välityksellä tiedon tilamuutoksista ja tapahtuman aikaleiman. Aikaleiman perusteella vastaanottava laite voi asettaa tapahtumaan liittyviä ajastimia. GOOSE-sanomien lähetys aloitetaan laitteen käynnistyttyä tai otettaessa palvelu käyttöön ja sanomia lähetetään jatkuvasti pitkissä sykleissä, vaikka mitään tilamuutoksia ei tapahtuisi. Tällä tavoin myös myöhemmin aktivoidut laitteet

saavat ajantasaiset tiedot muilta laitteilta. [56] Lisäksi tämä mahdollistaa siirtotien ja lähettävän laitteen jatkuvan valvomisen vastaanottavassa laitteessa. Vastaavista viantunnistusmekanismeista on käytetty nimitystä *heartbeat* [59].

GOOSE-sanomien lähetystä hallitaan GOOSE-ohjauslohkoilla (engl. *GOOSE Control Block, GoCB*). Ohjauslohkon *GoEna*-attribuutin avulla voidaan asettaa GOOSE-sanomien lähetys päälle tai pois päältä. Jos ohjauslohkon attribuuteissa on ristiriitaisuuksia, sanomien lähetystä ei ole mahdollista asettaa päälle. Ohjauslohkon attribuuttien arvoja ei ole myöskään mahdollista muuttaa, jos *GoEna*-attribuutin arvo on tosi. Ohjauslohkon tietojen puutteellisuus voidaan ilmaista *NdsCom*-attribuutilla, jos esimerkiksi viittaus datasettiin puuttuu. Ohjauslohko sisältää myös käyttäjän aseteltavissa olevan GOLD eli *GOOSE Identifier* –tunnisteen, joka on tyypiltään merkkijono. Muiden ohjauslohkojen tavoin *GoCB* sisältää *ConfRev*-attribuutin, joka ilmaisee kuinka monta kertaa ohjauslohkossa viitattuun datasettiin, on tehty muutoksia. Muutoksiksi lasketaan datasetin jäsenten lisääminen tai poistaminen, järjestyksen muuttuminen tai datasettiin viittaavan attribuutin muuttuminen ohjauslohkossa. [56, s. 133–134]

3.4.4 Näytearvojen lähetys

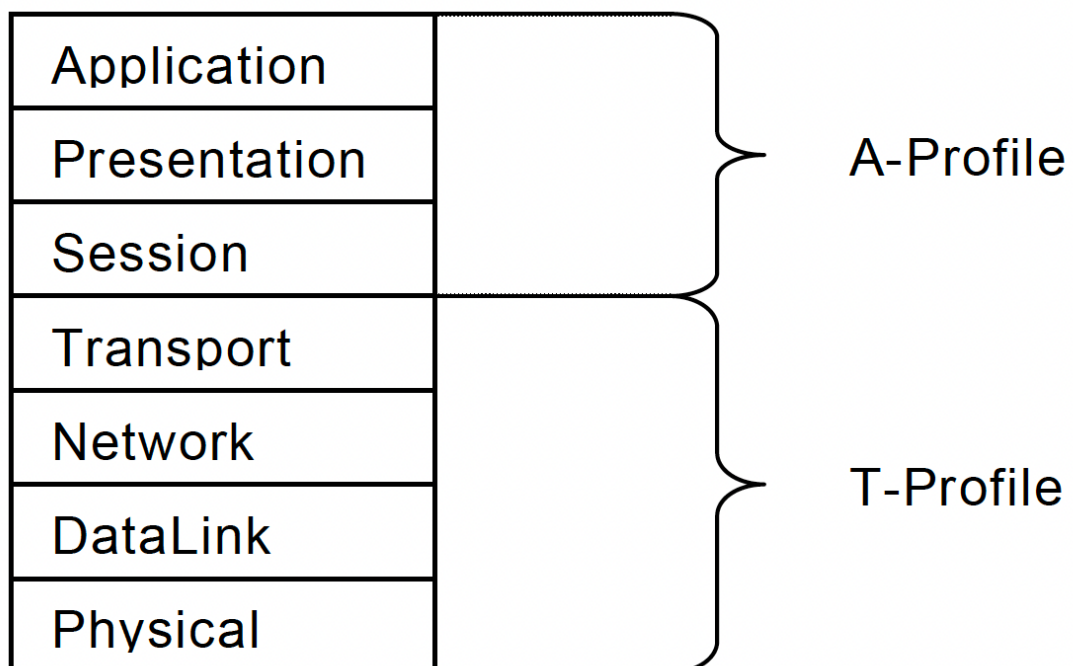
Näytearvojen lähetysmalli määrittää tavan lähettää määrättyllä taajuudella näytteistetyjä datasettiin ryhmiteltyjä arvoja. Malli pyrkii minimoimaan näytteenoton ja lähetysten yhdistetyn viivevaihtelun (engl. *jitter*). Lähetys tapahtuu joko *unicastina* tai *multicastina*. [56, s. 140–141]

Näytearvojen lähetystä hallitaan näytearvo-ohjauslohkoilla (engl. *Sampled Value Control Block, SVCB*), joita on kahta tyyppiä: *multicast-näytearvo-ohjauslohko* (engl. *Multicast Sampled Value Control Block, MSVCB*) ja *unicast-näytearvo-ohjauslohko* (engl. *Unicast Sampled Value Control Block, USVCB*). Ohjauslohkot ovat attribuuteiltaan nimiä lukuun ottamatta melkein identtisiä. *USVCB*:llä on lisäksi *Resv*-attribuutti, jonka avulla asiakas voi varata ohjauslohkon, jolloin muut asiakkaat eivät voi tehdä muutoksia sen attribuutteihin. Näytearvo-ohjauslohkot sisältävät kaikki samat attribuutit kuin *GOOSE-ohjauslohkot*, *NdsCom*-attribuuttia lukuun ottamatta. Näiden lisäksi *SVCB*:t sisältävät näytteenottotaajuuden hallintaan liittyvät attribuutit *SmpMod* ja *SmpRate* sekä *OptFlds*-attribuutin, jolla asiakas voi valita valinnaisia kenttiä *SV*-sanoman otsakkeeseen sisällytettäväksi. *SVCB*:n *ConfRev*-attribuutti poikkeaa myös *GoCB*:stä siten, että jäsenen lisäämistä datasettiin ei huomioida, mutta muutokset *SmpMod*-, *SmpRate*- tai *OptFlds*-attribuutissa huomioidaan. [56, s. 141–150]

3.5 Tietoliikenneprofiilit ja -protokollat

IEC 61850-5 jakaa automaatiojärjestelmän sisällä lähetetyt sanomat eri tyypeihin niiden toteuttaman toiminnallisuuden ja niille sallitun siirtoajan mukaan. Siirtoajalla tarkoitetaan standardissa sanoman siirtoon kuluva kokonaisaikaa, joka käsittää sanoman koodauksen ja lähettämisen, siirron tietoverkon yli sekä vastaanottamisen ja dekodauksen. [18] Sanomatyypien luokittelu poikkeaa hieman standardin eri osien [18], [51] ja [60] välillä. Sanomatyypit voidaan jakaa 6–7 päätyyppiin riippuen siitä, luetaanko aikasykronointisanomat yhdeksi tyyppiä. IEC 61850-5:ssä näin ei ole tehty, eikä aikasykronointisanomille määritetä siinä suorituskykyyn liittyviä vaatimuksia.

Suorituskykyyn liittyvät vaatimukset on täytetty standardissa erilaisilla tietoliikenneprofiileilla. Tietoliikenneprofiilit on jaettu sovellus- ja kuljetusprofiileihin, jotka muodostuvat OSI-mallin eri kerrosten protokollista ja niihin liittyvistä sopimuksista. Kuvasta 8 nähdään, että sovellusprofiili (engl. *application profile*) määrittää OSI-mallin kolme ylintä ja kuljetusprofiili (engl. *transmission profile*) neljä alinta kerrosta. Standardissa määritellyt tietoliikennepalvelut voidaan toteuttaa näiden profiilien erilaisilla kombinaatioilla. [51], [52]



Kuva 8. OSI-viitemalli ja profiilit. [52]

Standardin osissa 8-1 ja 9-2 määritetyt profiilit on esitetty kuvassa 9. Asiakas–palvelin-sovellusprofiili (engl. *client-server*) käyttää sovelluskerroksella MMS-protokollaa ja alemmilla kerroksilla muita ISO-standardoituja protokollia. Kuvassa esitetyn TCP/IP-

kuljetusprofiiliin lisäksi asiakas–palvelin-sovellusprofiili voi käyttää myös ISO-standardeihin pohjautuva kuljetusprofiilia, jonka toteuttaminen ei ole kuitenkaan pakollista [51, s. 26]. Aikasynkronointi toteutetaan sovellustasolla SNTP:llä (*Simple Network Time Protocol*), joka käyttää kuljetuskerroksella UDP:tä (*User Datagram Protocol*). Tiukempia aikasynkronointivaatimuksia varten IEC/IEEE 61850-9-3 [61] määrittää myös PTP:tä (*Precision Time Protocol*) käyttävän aikasynkronointiprofiilin, joka puuttuu kuvasta. PTP toimii GOOSE- ja SV-sanomien tavoin suoraan siirtoyhteyskerroksen päällä. IEC 61850-90-5 määrittää myös R-GOOSE- (*Routable-GOOSE*) ja R-SV-profiilit (*Routable-SV*), jotka lisäävät GOOSE- ja SV-profiileihin verkko- ja kuljetuskerroksen [62].

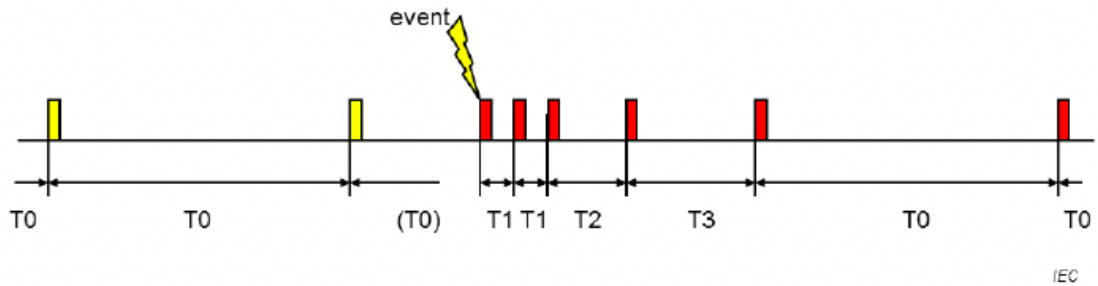
Siirtoyhteyskerroksen redundanssiprotokollien käyttö on kaikissa profiileissa valinnaista. SV-profiili sallii kuitenkin ainoastaan PRP:n (*Parallel Redundancy Protocol*) ja HSR:n (*High-availability Seamless Redundancy*), muiden profiilien salliessa myös RSTP:n (*Rapid Spanning Tree Protocol*) käytön. 802.1Q-tunniste mahdollistaa Ethernet-kehysten priorisoinnin ja erottelun virtuaalisiin lähiverkkoihin (*Virtual Local Area Network, VLAN*) [63, s. 244]. Prioriteetilla voidaan vaikuttaa siihen, missä järjestyksessä esimerkiksi kytkin käsittelee ja lähettää saapuvat kehykset eteenpäin [63, s. 136,197–201]. 802.1Q-tunniste on pakollinen ainoastaan GOOSE- ja SV-profiileissa. Fyysisellä kerroksella voidaan käyttää joko 100BASE-T, 100BASE-FX tai 1000BASE-LX Ethernetiä [60].

SV	GOOSE			Sovelluskerros	
			SNTP	MMS Protokollapino	Esitystapakerros
					Istuntokerros
			UDP	TCP	Kuljetuskerros
			IP		Verkkokerros
HSR tai PRP (V)	HSR, PRP tai RSTP (V)			Siirtoyhteyskerros	
802.1Q	802.1Q	802.1Q (V)	802.1Q (V)		
ISO/IEC 8802-3 EtherType					
100BASE-T/100-BASE-FX/1000BASE-LX				Fyysinen kerros	

Kuva 9. IEC 61850 –tietoliikenneprofiilit. Perustuu lähteisiin [51], [52], [60].

MMS on verkkokerroksen päällä toimiva asiakas–palvelin-protokolla, jolloin sen toiminta ei välttämättä rajoitu sähköaseman lähiverkkoon. MMS-protokollassa asiakkaan ja palvelimen välinen tiedonvaihto perustuu joko vahvistettuihin tai vahvistamattomiin palveluihin. Vahvistetuissa palveluissa MMS-asiakas lähettää kyselyn IED:n MMS-palvelimelle, joka lähettää pyydetyt tiedot vastaussanomassa asiakkaalle. Vahvistamattomissa palveluissa palvelin lähettää tietoja asiakkaalle spontaanisti asiakkaan tekemiin määrittelyihin perustuen. [60, s. 33] MMS-protokollaa on käsitelty laajemmin esimerkiksi lähteessä [64].

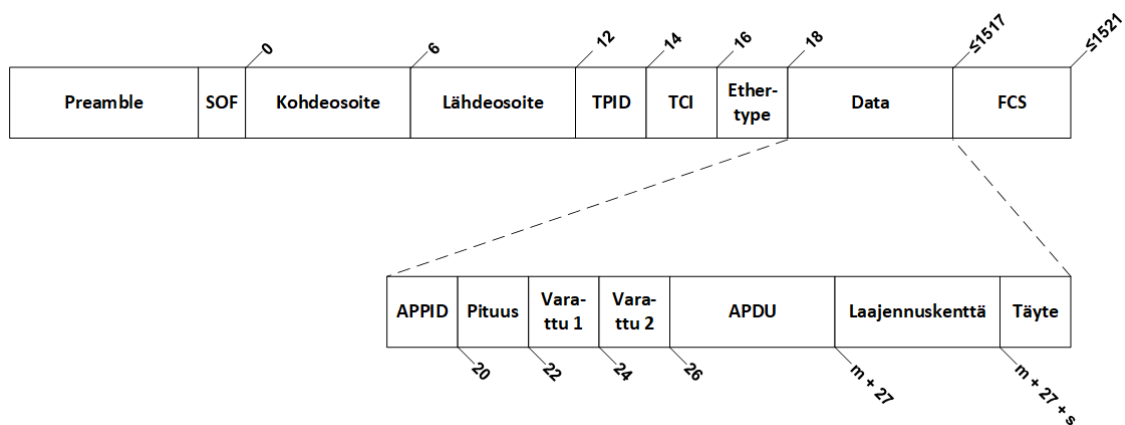
GOOSE on julkaisija–tilaaja-periaatteella toimiva protokolla, joka mahdollistaa aikakriittisten tietojen lähettämisen yhdelle tai useammalle vastaanottajalle lähiverkon sisällä Ethernetin multicast-toiminnon avulla [60, s. 34]. GOOSE-sanoman lähettäminen lähiverkon ulkopuolelle on mahdollista tunneloimalla SDH- (*Synchronous Digital Hierarchy*) tai SONET-verkon (*Synchronous Optical Networking*) yli tai käyttämällä R-GOOSE-profiilia [62]. Koska sanomia ei kuitata, ne lähetetään läpimenon varmistamiseksi useita kertoja peräkkäin. Julkaisijan olemassaoloa valvotaan lähettämällä sanomat toistuvasti pidemmällä intervallilla. Kuvassa 10 on esitetty GOOSE-sanomien lähetys aika-akselilla. Keltainen suorakulmio kuvaa GOOSE-sanoman lähetystä ennen tapahtumaa ja punainen tapahtuman jälkeen. Kun tapahtumia ei ole ollut pitkään aikaan, käytetään suurinta mahdollista uudelleenlähetysaikaa T0. GOOSE-sanomaan liitetty tapahtuma lyhentää uudelleenlähetysaikaa pienimpään mahdolliseen aikaan T1. Tämän jälkeen uudelleenlähetysaikaa pidennetään asteittain intervaleihin T2 ja T3, kunnes palataan pysyvään tilaan (engl. *steady state*) ja lähetystä jatketaan intervallilla T0. [60, s. 34–35]



Kuva 10. GOOSE-sanomien lähetys ennen tapahtumaa ja sen jälkeen [60, s. 35].

SV-protokollaa käytetään aikakriittisten analogiatietojen lähettämiseen antureilta IED-laitteille syklisesti korkealla taajuudella multicastin avulla [60, s. 36]. Myös unicast on ollut aiemmin tuettu, mutta se on määritetty vanhentuneeksi standardin editiossa 2.1. [52]

Kuvassa 11 on esitetty GOOSE- tai SV-sanoman ja sitä kuljettavan Ethernet-kehysten rakenne ilman siirtoyhteyskerroksen redundanssiprotokollin liittyviä osia. PRP ja HSR kasvattavat kehysten pituutta kuudella oktetilla. *Ether-type*-kenttä ilmaisee kumman protokollan mukaisen sanoman kehys sisältää.



Kuva 11. GOOSE- tai SV-sanoma Ethernet-kehyksessä. Perustuu lähteisiin [49], [50].

TPID- (*Tag Protocol Identifier*) ja TCI-kentät (*Tag Control Information*) ovat osa 802.1Q-tunnistetta. TPID ilmaisee, että kehys sisältää 802.1Q-koodattua tietoa ja sen arvo on aina 0x8100. TCI sisältää prioriteetin ja VID-tunnisteen (*VLAN Identifier*), jonka avulla kehys voidaan määrittää virtuaaliseen lähiverkkoon kuuluvaksi. Jos virtuaalisia lähiverkkoja ei haluta käyttää, asetetaan VID nollassi. IEEE 802.1Q –standardia noudattavat kytkimet kuitenkin poistavat 802.1Q-tunnisteet, jos VID on nolla. Tämän vuoksi priorisointia käytettäessä tulisi valita jokin muu VID. [51,52]

APPID (*Application Identifier*) tunnustetta voidaan käyttää sanomia lähettävien sovellusten tunnistamiseen. APPID:n kaksi merkitsevintä bittiä määräytyvät käytetyn protokollan tai käyttötarkoituksen mukaan. 14 vähiten merkitsevää bittiä voidaan käyttää sanomien suodattamiseen ja vähentämään prosessorin kuormitusta. APPID suositellaan valitsemaan siten, että se on yksilöllinen sähköaseman sisällä. [60, s. 118]

Multicast-sanomia vastaanottavien laitteiden kuormitukseen voidaan vaikuttaa myös multicast-osoitteiden suodatuksella. Ethernet-ohjain purkaa ja välittää vastaanotetut kehykset sovellukselle vain, jos multicast-osoite kuuluu sen tilaamiin osoitteisiin [60]. Osoitteita valitessa tulisi ottaa huomioon ohjainten mikropiirien hajautusalgoritmeissa olevat erot ja suositukset osoitteiden valintaan tulisi saada laitevalmistajalta [51], [52]. Jos käytetyt IED-laitteet eivät tue multicast-suodatusta, voidaan suodatus toteuttaa myös kytkimissä. Multicast-sanomien leviämistä lähiverkossa voidaan lisäksi rajoittaa VLAN-määrittelyiden avulla.

Kehyksen Pituus-kenttä ilmaisee Ethertype PDU-otsakkeen (*Protocol Data Unit*) koon oktetteina APPID:sta APDU:n (*Application Protocol Data Unit*) loppuun. Varattu 1 – kenttä sisältää S-bitin, jota käytetään ilmaisemaan, että sanoman julkaisija on testilaitte eikä alkuperäinen konfiguraatiodostossa määritetty julkaisija. Se sisältää myös kolme bittiä, jotka on varattu tulevia standardoituja sovelluksia varten. Loput Varattu 1 –kentästä on varattu IEC 62351-6 standardissa määritellyjä turvallisuusominaisuuksia varten. Varattu 2 –kenttä on varattu kokonaan turvallisuusominaisuuksia varten. Myös laajennuskentän sisältö on määritetty IEC 62351-6:ssa. [51], [52], [60] APDU:n sisältö on määritetty IEC 61850-7-2:ssa ja se on hieman erilainen GOOSE- ja SV-sanomille.

3.6 Työkalut

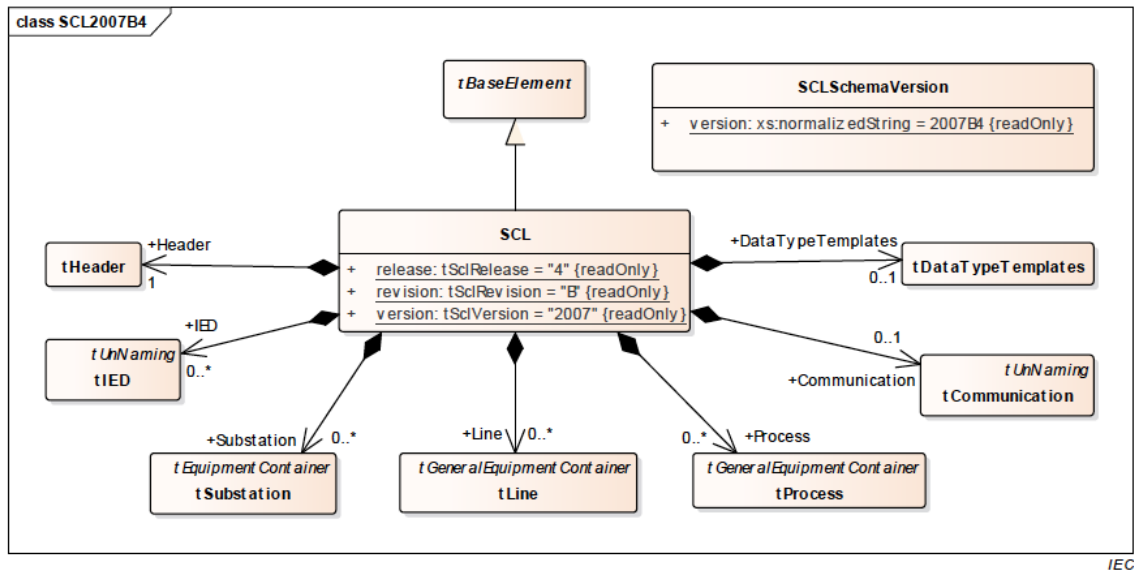
Standardi määrittää kolme erilaista suunnittelu- ja konfigurointityökalua. *SST (System Specification Tool)* on tarkoitettu prosessin sekä siihen liittyvien toimintojen ja nimien kuvaamiseen pääkaaviotasolla projektin määrittelyvaiheessa. *SCT:tä (System Configuration Tool)* käytetään järjestelmän IED-laitteiden välisen tietoliikenteen sekä IED-laitteiden toimintojen ja prosessin välisten loogisten yhteyksien määrittämiseen. Standardin mukaan SST- ja SCT-työkalut pohjautuvat tyypistä riippuen erilaisiin mallipohja- tai IED-laitetietokantoihin ja sama ohjelmisto sisältää usein sekä SST:n että SCT:n ominaisuudet. *ICT (IED Configuration Tool)* on laite- tai laitevalmistajakohtainen työkalu, jota käytetään IED-laitteiden konfigurointiin. IED-laitteelta vaaditut toiminnot ja järjestelmäkonfiguraatio voidaan tuoda työkaluun SCL-muodossa, mutta tarkemmat laitekohtaiset määrittelyt on tehtävä työkalun sisällä. [49, s. 17–18]

Standardi määrittää näiden lisäksi myös dokumentointityökalun, jonka tulisi pystyä luomaan projekteille yhtenäistä ja standardinmukaista dokumentaatiota. Työkalun olisi kyettävä tuottamaan erilaista laitteistoihin, ohjelmistoihin ja parametreihin liittyvää dokumentaatiota sekä ylläpitämään revisiohistoriaa, joka sisältää tiedot kaikista työkaluun tallentuneista muutoksista. [49, s. 19]

Useimmat kolmannen osapuolen ohjelmistot kuten esimerkiksi Helinks STS [65], Kalkitechin ASE61850 SCL Manager [66] ja GridSoftwaren SCL Matrix [67] sisältävät sekä SST- että SCT-työkalut. Sen sijaan laitevalmistajien omat työkalut kuten Hitachin IET600 [68] ja Siemensin IEC 61850 System Configurator [69] ovat ainoastaan SCT-työkaluja. Poikkeuksena Schneider Electricin EcoStruxure Power Automation System Engineering - EPAS-E [70], joka sisältää myös SST:n. Edellä mainitut laitevalmistajien SCT-työkalut ovat suunnitteluprosessin kannalta laitevalmistajasta riippumattomia eli niitä voidaan käyttää riippumatta siitä, minkä valmistajan tuotteita järjestelmä sisältää. ICT-työkalut ovat yhteensopivia ainoastaan kyseisen laitevalmistajan tuotteiden kanssa, kuten esimerkiksi Hitachin PCM600 [71] ja Siemensin DIGSI 5 [72].

3.7 SCL

SCL on XML-pohjainen konfiguraatiokieli, jonka ensisijainen tarkoitus on mahdollistaa eri valmistajien konfiguraatiotyökalujen välinen tiedonvaihto. SCL:n avulla voidaan kuvata primääriprosessi, IED-laitteiden parametrit ja tietoliikennekonfiguraatio, tietoliikennejärjestelmän konfiguraatio sekä näiden väliset yhteydet. SCL käyttää XML:n versiota 1.0 ja sen syntaksi on määritetty XML-skeemassa. Kaikkia oliomallille asetettuja rajoitteita ei ole pystytty määrittämään suoraan XML-skeemassa ja osa niistä on kuvattu erikseen standardissa. [50] XML-skeeman rakenne ja sen elementtien muodostumissuhteet on esitetty kuvassa 12.



Kuva 12. XML-skeeman rakenne ja elementtien muodostumissuhteet. [50, s. 37]

Kaikissa SCL-tiedostoissa on tunnistetietoja sisältävä otsake (engl. *header*). Otsakkeen sisään voidaan sisällyttää myös tiedoston revisiohistoria, joka ei ole kuitenkaan pakollista. Otsakkeen lisäksi tiedostot voivat sisältää kolmiosaisen tietomallin kuvaamiseen käytetyt prosessi-, IED- ja tietoliikenneosiot sekä IED-laitteiden tietotyypipohjista koostuvan *data type templates* -osion. [50]

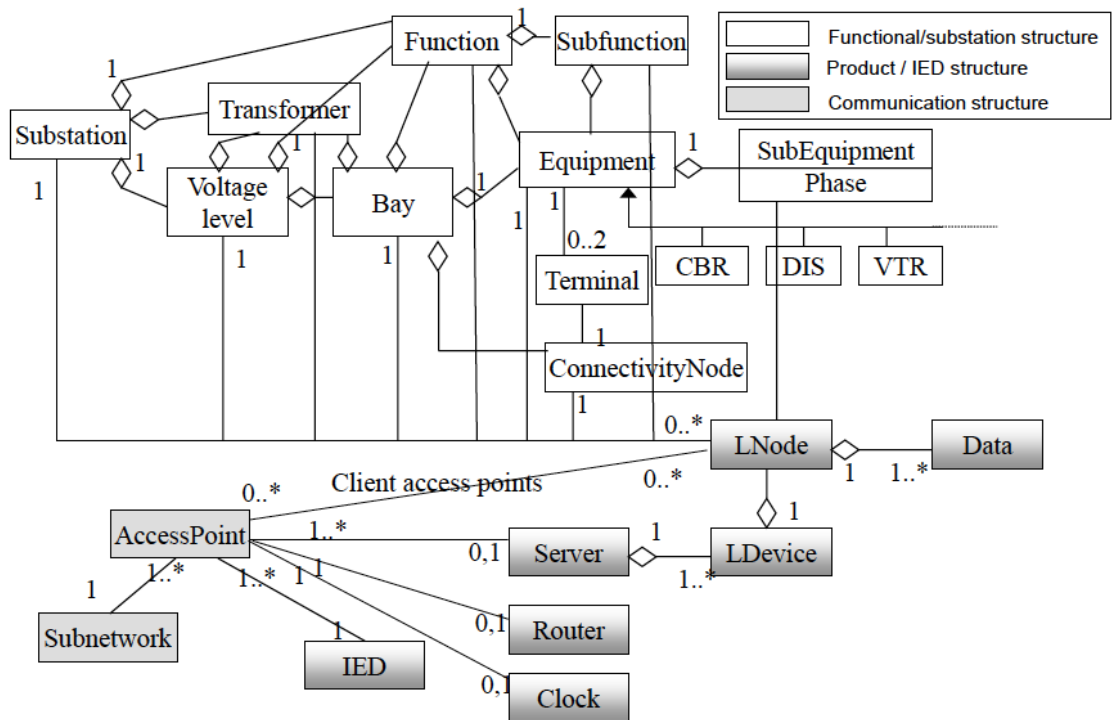
Prosessiosioon voidaan sisällyttää kuvaus primääriprosessista, kuten sähköasemasta ja siihen kuuluvista ensilaitteista. Useita *Substation*-elementtejä ja voimajohtoja kuvaavia *Line*-elementtejä on mahdollista koota *Process*-elementin alle, jolloin myös yksittäistä sähköasemaa laajemman kokonaisuuden kuvaaminen on mahdollista. *Process*-elementtiä voidaan käyttää myös jonkin muun prosessin kuvaamiseen. Prosessiosion kuvaamalla prosessimallilla on kaksi tarkoitusta. Yhdistää loogiset solmut primääriprosessiin sekä määrittää loogisille solmuille prosessiin viittaava tunnus. Jos prosessiosioon liitetään loogisia solmuja, se voi kuvata järjestelmän rakenteen lisäksi myös sen toiminnallisuutta. [50]

IED-osio sisältää laitekohtaiset tietoliikenteen liityntäpisteet (engl. *access points*), loogiset laitteet, loogisten solmujen ja dataobjektien instanssit sekä listan tuetuista tietoliikennepalveluista. Ne muodostavat IED-osion kuvaaman IED-laitteen konfiguraation tai esikonfiguraation. Liityntäpiste voi olla osa palvelinta (engl. *server*), jolloin se mahdollistaa pääsyn palvelimen sisältämään tietomalliin. Asiakas (engl. *client*) taas voi käyttää kaikkia IED:n liityntäpisteitä. SCL-tiedostossa on oltava jokaista IED-laitetta kohden yksi IED-osio. [50]

Tietoliikenneosio kuvaa yhteiseen aliverkkoon kytkettyjen IED-laitteiden liityntäpisteet sekä niiden parametrit. Liityntäpisteellä voi olla yksi palvelimeen viittaava osoite ja lisäksi reaaliaikaiseen tiedonsiirtoon liittyvien ohjauslohkojen osoitteet. Liityntäpisteen fyysinen toteutustapa voidaan kuvata *PhysConn*-elementin avulla. [50]

Data type templates –osio sisältää loogisten solmujen tyyppipohjia, joiden pohjalta loogisten solmujen instanssit luodaan. Jokaisessa IED-laitteeseen luodussa loogisessa solmussa on viittaus *LNodeType*-elementtiin, jotka voidaan erottaa toisistaan yksilöllisen tunnusteen avulla. *LNodeType*-elementti koostuu tietomallin alemman tason osista, jotka jälleen viittaavat omiin tyyppipohjiinsa, jotka koostuvat taas alemman tason osista, aina perustyypppeihin asti. [50]

SCL:n tietomallissa prosessi- ja IED-osa ovat rakenteeltaan hierarkkisia ja verrattavissa standardissa IEC 81346 määritettyihin toiminnan ja tuotteen mukaisiin rakenteisiin. Hierarkioita käytetään muodostettaessa signaalinnimiä ja viittauksia tietomallin olioihin. Kuvassa 13 on esitetty pelkistetty kuvaus SCL:n oliomallista, joka ei sisällä kantaluokkia ja luokkien attribuutteja. [50] Kuvasta nähdään miten prosessiosan instanssien muodostama hierarkia ja eri osien instanssit kytkeytyvät toisiinsa. Valkoisella vino-neliöllä merkittyä assosiaatiosuhdetta kutsutaan koosteeksi (engl. *shared aggregation* tai *aggregation*). Kooste eroaa muodostumisesta siten, että luokan instanssi voi kuulua useampaan isäntäluokkaan samanaikaisesti eikä sen elinkaari ole välttämättä sidottu sen isäntäluokkaan. [57, s. 134–135] Luokkien välille piirretyt suorat viivat tarkoittavat kaksisuuntaista assosiaatiota. Tuoteosan loogiset solmut eivät varsinaisesti sisällä viit-tausta prosessiosaan, joten tuote- ja prosessiosien väliset assosiaatiot ovat todellisuu-dessa yksisuuntaisia.



IEC

Kuva 13. SCL:n oliomalli ilman kantaluokkia ja attribuutteja. [50]

SCL-tiedostoja on määritetty eri käyttötarkoituksia varten kuusi eri tyyppiä, jotka eroteetaan toisistaan tiedostopäätteen avulla. Jokainen tiedostotyyppi noudattaa SCL:lle asetettuja sääntöjä. Standardissa on määritetty seuraavat tiedostotyypit:

- *ICD (IED Capability Description)* kuvaa IED-tyyppin ominaisuudet ja sisältää aina yhden IED-osion sekä vaadittavat tietotyyppipohjat. Se voi sisältää myös prosessiosion, jolloin siihen liitettyjen loogisten solmujen käyttötarkoitus on ennalta määrätty. Lisäksi laitetypille on voitu määrittää oletusosoitteet tietoliikenneosiossa.
- *IID (Instantiated IED Description)* sisältää yksittäisen osittain konfiguroidun laitteen tiedot eli laitteen nimi, osoitteet ja tietomalli on voitu jo määrittää. Tiedosto voi sisältää myös datasettejä, ohjauslohkoja ja sisääntuloja sekä loogisten solmujen liitännät prosessiosioon. Tiedostoa voidaan käyttää IED:n tietojen päivittämiseen tai joissain tapauksissa uusi laite voidaan viedä IID-tiedostona SCT:hen.
- *CID (Configured IED Description)* on valmistajakohtainen tiedosto, joka sisältää yhden IED-instanssin tietoliikennekonfiguraation. Tiedoston standardinmukainen käyttötarkoitus on ainoastaan tietojen siirto ICT:stä IED:hen.

- *SSD (System Specification Description)* sisältää kuvauksen sähköaseman primääriprosessista, siihen liittyvistä toiminnoista ja loogisista solmuista. Siinä on aina vähintään sähköasemaosio ja tarvittaessa data type templates –osio.
- *SCD (System Configuration Description)* kokoaa yhteen kaikki järjestelmän IED:t tietoliikennemäärittelyineen sekä tarvittavat tietotyypipohjat. Se sisältää myös sähköasema- ja tietoliikenneosiot.
- *SED (System Exchange Description)* sisältää IEC 61850 –projektikohtaiset IED-laitteiden rajapinnat, joita voidaan käyttää eri projektien SCT-työkalujen välisessä tiedonvaihdossa. [50, s. 33–34]

Työn kirjoitushetkellä vielä valmisteilla olevan IEC 61850-90-30:n (ent. 61850-6-100 [73]) on tarkoitus esitellä lisäksi ainakin kolme uutta tiedostotyyppiä, jotka tukevat erityisesti järjestelmän spesifiointia:

- *ISD (IED Specification Description)* sisältää kuvauksen laitevalmistajariippumattomasta virtuaalisesta IED-laitteesta, jota voidaan käyttää IED-laitteiden vaatimusten määrittämiseen järjestelmän spesifiointivaiheessa [5].
- *FTD- (Function Template Description)* ja *ASD- (Application Specification Description)* tiedostoja käytetään IEC 61850-7-6:ssa esiteltyjen perussovellusprofiilien (engl. *Basic Application Profile, BAP*) määrittämiseen SCL-muodossa [74].

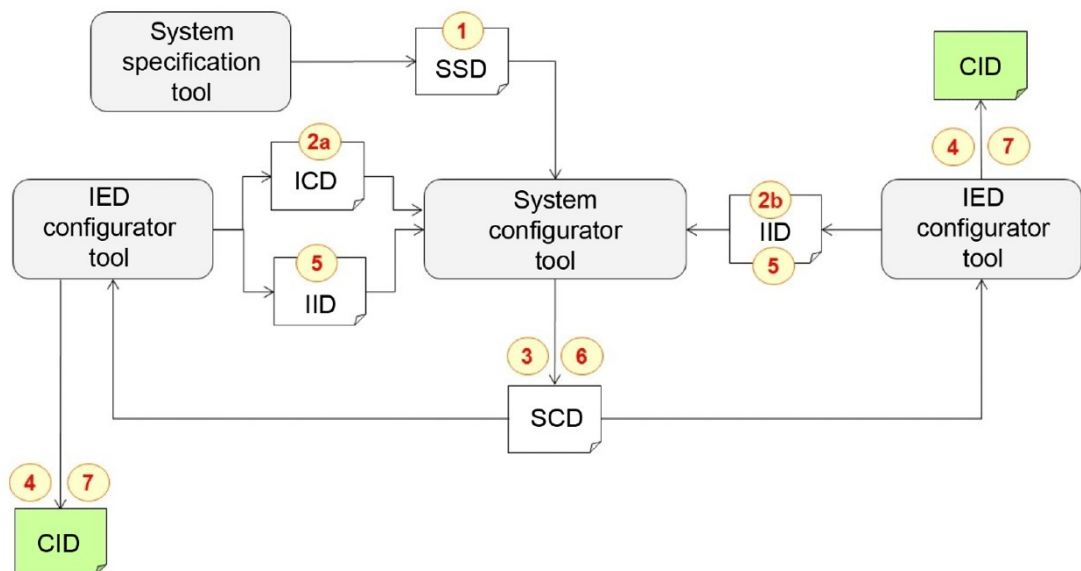
3.8 Suunnitteluprosessi

IEC 61850-6 määrittää mitä asioita SCL-tiedostoilla on pystyttävä kuvaamaan ja miten niitä suositellaan käytettäväksi osana järjestelmän suunnittelua ja konfigurointia. Sähköasema-automaatiojärjestelmän suunnittelu ja konfigurointi voidaan standardin mukaan aloittaa joko allokoimalla esikonfiguroidut laitteet prosessin osiin, tuotteisiin tai toimintoihin, tai suunnittelemalla ensin prosessiin liittyvä toiminnallisuus, jolloin toiminnot allokoidaan fyysisiin laitteisiin myöhemmin niiden ominaisuuksien perusteella [50, s. 18]. Standardissa todetaan myös, että lähestymistapa on usein jotain näiden väliltä, esimerkiksi prosessin eri osat voidaan esikonfiguroida ja monistaa tarvittavan monta kertaa [50, s. 18].

Järjestelmän konfigurointi aloitetaan luomalla ICD-tiedostojen pohjalta IED-instanssit SCT-ohjelmistossa. Vaihtoehtoisesti voidaan käyttää myös IID-tiedostoja, joihin on esikonfiguroitu nimi ja prosessin toimintoihin liittyvät osoitteet. SCT-ohjelmiston luoma SCD-tiedosto viedään lopuksi ICT-työkaluihin lopullisten IED-instanssien konfiguraation

viimeistelyä varten. Mahdolliset muutokset ICT-työkalusta voidaan tämän jälkeen viedä takaisin SCT-työkaluun uutta SCD-revisiota varten IID-tiedostossa. [50, s. 22] IEC 61850-6 ei varsinaisesti ota kantaa SSD-tiedoston käyttöön osana järjestelmän konfigurointia muuten kuin, että SCT:n on pystyttävä tuomaan SSD-tiedosto esimerkiksi järjestelmäkonfiguraation pohjaksi tai verrattavaksi konfiguroituun järjestelmään [50, s. 19].

Vastaava työnkulku SSD-tiedoston kanssa on kuvattu vaiheittain IEC 61850-4:ssä ja esitetty kuvassa 14. Ensin SST-ohjelmisto luo SSD-tiedoston, joka viedään SCT-ohjelmistoon. Seuraavaksi ICT-ohjelmistot luovat IID- ja ICD-tiedostot, jotka viedään SCT-ohjelmistoon. Kolmannessa vaiheessa SCT luo SCD-tiedoston SSD-, ICD- ja IID-tiedostojen pohjalta, joka viedään ICT-ohjelmistoihin. Neljänneksi ICT viimeistelee konfiguraation SCD:n avulla ja luo CID-tiedoston. Jos ICT-ohjelmiston täytyy tehdä muutoksia SCD-tiedoston sisältöön, vaihe neljä korvataan vaiheella viisi, jossa ICT tekee muutokset ja tämän seurauksena syntynyt IID-tiedosto viedään SCT-ohjelmistoon. Vaiheessa kuusi SCT luo päivitetyn SCD-tiedoston ottaen huomioon IID-tiedoston päivitettyt tiedot ja päivitetty SCD-tiedosto viedään ICT-ohjelmistoihin. Vaiheessa 7 luodaan päivitetty CID-tiedostot, jotka viedään IED-laitteisiin. [49, s. 19–20] IEC 61850-4:ssä kuvataan lisäksi vastaavat työkulut SCT-työkalun vaihtoon ja tietojen vaihtoon eri SCT-projektien välillä.

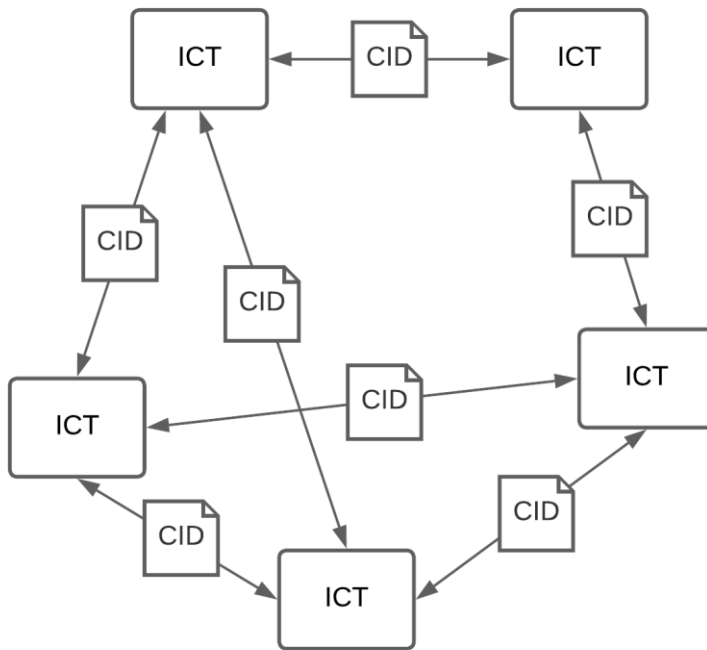


Kuva 14. Standardissa kuvattu työnkulku vaihe vaiheelta. [49]

Edellä kuvatusta lähestymistavasta, jossa järjestelmän konfigurointi aloitetaan pääkaavioesityksen ja siihen liittyvien toimintojen määrittelystä SSD-tiedostoon, käytetään nimitystä *top-down engineering*. Määritelmästä on kuitenkin olemassa erilaisia

näkemyksiä. Lähteessä [6] esitetystä menetelmästä ICD-tiedostot viedään myös SST-työkaluun. Tällöin valmistajakohtainen tietomalli olisi käytössä jo järjestelmän määrittelyvaiheessa, jolloin käytetyt IED-laitteet olisi mahdollisesti jo valittu ja lukittu ennen järjestelmän konfigurointia. Joskus samaa nimitystä saatetaan käyttää myös menetelmästä, joka muistuttaa muuten kuvassa 14 esitettyä, mutta ilman SSD-tiedostoa. IEC 61850-6 [50] määrittää ICT- ja SCT-työkaluille roolit konfigurointitehtävien ja vastuiden selkeyttämiseksi. SST-työkalulle vastaavaa määrittelyä ei ole tehty.

Lähteen [12] määrittelyn mukaan myös *bottom-up*-menetelmässä ICD-tiedostot viedään SCT-ohjelmistoon ja valmistajakohtaisen tietomallin loogiset solmut linkitetään prosessiin, joka on määritetty SSD-tiedostossa. Tässä määrittelyssä menetelmiä erottaa ainoastaan IED-laitteiden tietomallin vapaa muokattavuus, kun taas lähteen [6] määrittelyn mukaan *bottom-up*-menetelmässä SSD- tai edes SCD-tiedostoa ei luoda välttämättä ollenkaan. Lähteessä [6] esitetty näkemys on näistä varmasti yleisempi ja sitä tukee esimerkiksi ohjelmistovalmistajien esitykset [75, 76]. Vaikka SCD-tiedosto luotaisiinkin, se ei yleensä sisällä substation-osiota. Kuvassa 15 on esitetty esimerkki *bottom-up*-prosessista, jossa järjestelmän konfiguraatio on toteutettu kokonaisuudessaan ICT-ohjelmistolla tai -ohjelmistoilla ja laitekonfiguraatioiden välinen tiedonvaihto tapahtuu CID-tiedostoilla. IEC 61850 –konfiguraatio rakennetaan ensin yksitellen ICT:ssä, jonka jälkeen eri IED-laitteiden tarvitsemat tiedot viedään CID-tiedostolla ohjelmistosta tai laitekonfiguraatiosta toiseen tietoliikennemäärittelyiden viimeistelemiseksi. Jos esimerkiksi haluttaisiin toteuttaa GOOSE-sovellus olisi ensin luotava tarvittavat datasetit ja ohjauslohkomäärittelyt jokaiselle laitteelle erikseen ICT:ssä ja tämän jälkeen vietävä CID-tiedostot jokaisen GOOSE-sanomia vastaanottavan laitteen konfiguraatioon.



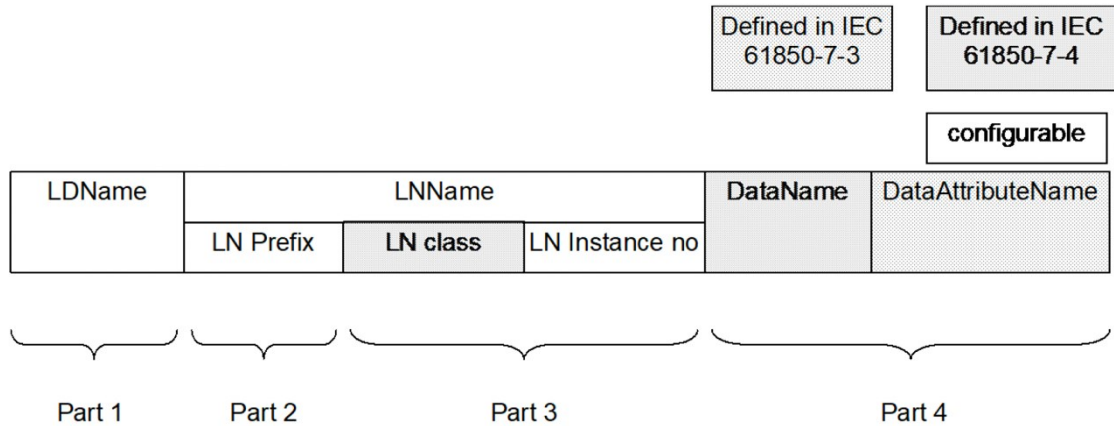
Kuva 15. Esimerkki bottom-up-prosessista. Perustuu lähteeseen [76].

3.9 Signaalien nimeäminen

SCL sisältää kaksi olioiden nimeämiseen liittyvää parametria. Tekninen avain (engl. *technical key*), joka määritetään attribuutilla *name*, on tarkoitettu signaalien nimeämiseen sekä teknisissä piirustuksissa käytettäväksi. Sitä käytetään myös SCL:n sisäisissä viittauksissa. Kuvaus (engl. *description*) on käyttäjän määrittämä tekstimuotoinen nimitys, joka määritetään attribuutilla *desc*. Kuhunkin olioon voidaan viitata polulla, joka muodostuu tietomallin eri tasojen nimistä kyseiseen tasoon asti. Jos olio kuuluu useampaan hierarkiaan, siihen voidaan viitata useammalla polulla. Edellisten lisäksi on olemassa tunniste *Text*, jota voidaan käyttää yleisen tekstimuotoisen kuvauksen lisäämiseen. [50, s. 45–46] Joissain testaustyökaluissa *desc* attribuuttia voidaan käyttää signaalien nimen esittämiseen selkokielisemmässä muodossa teknisen avaimen sijaan [7]. Kuvaukset voidaan määrittää esimerkiksi siten, että niistä muodostuu signaalilistan mukainen signaalinimi.

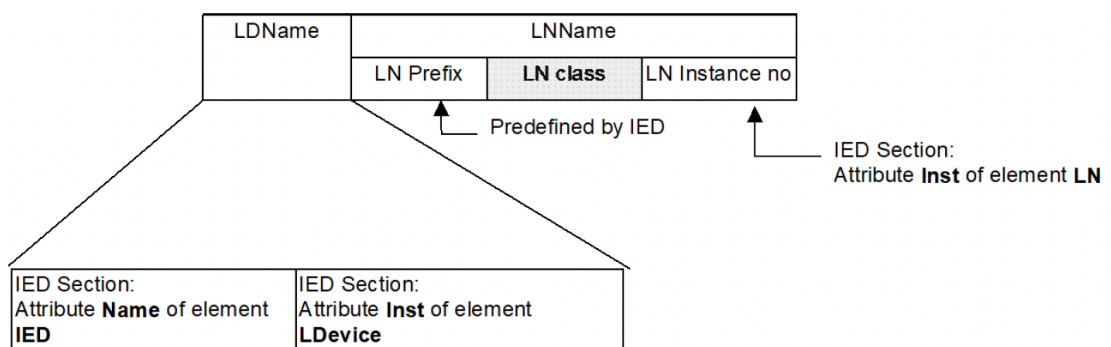
Signaalien nimet koostuvat neljästä osasta, jotka on esitetty kuvassa 16. Osa yksi on käyttäjän määriteltävissä oleva loogisen laitteen nimi. Osa kaksi on loogisen solmun etuliite, jota käytetään erottamaan samaan luokkaan kuuluvat loogiset solmut toisistaan saman IED:n tai loogisen laitteen sisällä. Osa kolme koostuu loogisen solmun nimestä sekä solmun instanssinumerosta. Instanssinumeroa käytetään erottamaan samaan luokkaan kuuluvat loogiset solmut toisistaan, jos niillä on sama etuliite tai niillä ei ole

etuliitettä ollenkaan. Neljäs osa koostuu datan objektin ja data attribuutin nimestä. Signaalin nimen todellinen esitystapa tietoliikenteessä riippuu käytetystä protokollasta. [50, s. 46–47]



Kuva 16. IEC 61850 –signaalin nimen muodostuminen. [50, s. 46]

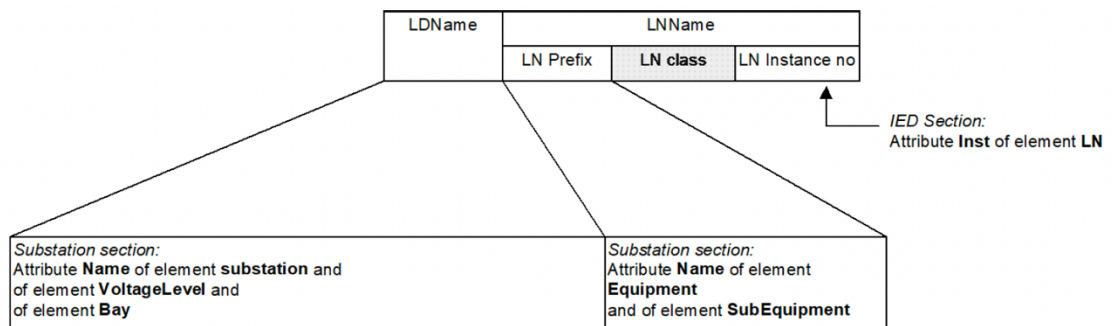
Standardissa on esitetty kaksi tapaa osien yksi ja kaksi määrittämiseen, tuoteliitännäinen nimeäminen (engl. *product-related naming*) ja toimintoliitännäinen nimeäminen (engl. *function-related naming*). IED:n on tuettava vähintään tuoteliitännäistä nimeämistä ja sitä käytetään aina SCL:n sisäisissä viittauksissa. Tuoteliitännäisessä nimeämistavassa loogisen laitteen nimi osassa yksi koostuu SCL-tiedoston IED-osiosta löytyvästä IED-nimestä ja IED:n LD-instanssitunnuksesta. Loogisen solmun etuliite osassa kaksi sekä instanssinumero osassa kolme on määrätty ennalta laitteen tietomallissa. [50, s. 47–48] Tuoteliitännäisen nimeämisen periaate on esitetty kuvassa 17.



Kuva 17. Tuoteliitännäinen nimeäminen. [50, s. 47]

Toimintoliitännäistä nimeämistä ei ole määritetty standardissa yhtä tarkasti. Sen käyttö edellyttää, että IED-laittevalmistaja sallii vähintään loogisen laitteen vapaan nimeämisen sekä vaihtoehtoisesti myös loogisen solmun etuliitteen vapaan nimeämisen. Loogisen laitteen nimi voi standardin mukaan viitata esimerkiksi primääriprosessin toimintoon tai

toiminnon tyyppiin. Ensiölaitteen tapauksessa se voitaisiin muodostaa sähköaseman, jännitetason ja kentän tunnuksista. Loogisen solmun etuliitteenä käytettäisiin tällöin laitteen tunnusta. Ylempien tasojen toimintoja mallinnettaessa loogisen laitteen nimestä voidaan jättää tarpeettomat tunnukset pois ja loogisen solmun etuliite tarvittaessa tyhjäksi. Toimintoliitännäisen nimeämisen periaate on esitetty kuvassa 18. Nimeämisessä on huomioitava, että sama loogisen laitteen nimi saa esiintyä lähiverkon sisällä ainoastaan kerran. Lisäksi standardissa suositellaan, että samaan luokkaan kuuluvilla loogisilla solmuilla on aina eri instanssinumero, jolloin loogisen solmun etuliitettä voidaan tarvittaessa muuttaa jälkikäteen. [50, s. 47–48] LD-nimen, LN-etuliitteen ja LN-instanssinumeron vapaasta muuttamisesta on käytetty myös nimitystä *flexible product naming* (FPN) [77].



Kuva 18. Toimintoliitännäinen nimeäminen. [50, s. 48]

3.10 SCL-tiedostojen validointi

SCL:n validoinnilla tarkoitetaan yhden määritelmän [78] mukaan tekniikoita ja työkaluja, joita tarvitaan IED-laitteen SCL-tiedoston yhdenmukaisuuden tarkistamiseksi IEC 61850:n kanssa. Validointia on kuitenkin käytännössä kyettävä tekemään kaikille SCL-tiedostotyypeille ja samassa lähteessä todetaankin myös, että ihanteelliseen validointityökaluun on pystyttävä helposti tuomaan kaikki SCL-tiedostotyypit.

Validointityökalun avulla voidaan tarkistaa, että SCL-tiedosto on sekä semanttisesti että syntaktisesti yhteensopiva standardin kuudennen osan kanssa. Tarkistuksen suorittamiseksi työkalun on kyettävä lukemaan SCL-skeema ja tietomalli SCL-tiedostosta. [78] Tyypillisesti SCL-tiedostot validoidaan järjestelmän konfigurointivaiheessa ja lisättäessä uusia IED-laitteita järjestelmään [79].

Tiedostojen validoinnin avulla voidaan pyrkiä tunnistamaan ja välttämään mahdolliset ongelmat laitteiden välisessä tietoliikenteessä ennen kuin ne ilmenevät fyysisessä järjestelmässä. Validointia on hyödynnetty esimerkiksi UCAlug:n (*UCA International*

Users Group) vuonna 2019 järjestämässä IEC 61850 IOP-testaustilaisuudessa (*Interoperability*), jota varten eri laitevalmistajien toimittamat ICD-tiedostot validoitiin ennen tilaisuuden alkua useilla eri validointityökaluilla [80]. Tällä menettelyllä pyrittiin ilmeisesti välttämään aiemmissa tilaisuuksissa haittaa aiheuttaneet SCL-tiedostojen laatuongelmat. Tilaisuuden loppuraportista kuitenkin ilmenee, että vain muutama testatuista ICD-tiedostoista läpäisee validoinnin kaikilla käytetyillä työkaluilla. Kaikkia tiedostoja ei myöskään testattu samoilla työkaluilla vaan käytetyt työkalut vaihtelivat tiedostoittain.

4. OHJELMISTOTYÖKALUJEN ARVIOINTI JA VALINTA

Erilaisten ohjelmistojen arviointiin ja valintaan liittyen on tehty useita tutkimuksia. Tässä työssä käytetty ohjelmiston valinnan metodologia muistuttaa lähteessä [81] kuvattua seitsemänvaiheista metodologiaa:

1. Määritellään tarve hankittavalle ohjelmistolle ja selvitetään tarpeeseen soveltuvien ohjelmistojen saatavuus.
2. Listataan ohjelmistot.
3. Poistetaan listasta soveltumattomat ohjelmistot, jonkin kriteerin perusteella.
4. Arvioidaan jäljelle jääneet ohjelmistot arviointimenetelmää käyttäen ja pisteytetään tai asetetaan ohjelmistot paremmuusjärjestykseen.
5. Hankitaan parhaista ohjelmistoista kokeiluversiot ja suoritetaan empiirinen arviointi.
6. Neuvotellaan sopimus, jossa määritellään ehdot ohjelmiston hankinnalle.
7. Hankitaan ja otetaan käyttöön soveltuvin ohjelmisto. [81]

Tässä luvussa käsitelty ohjelmistojen arviointi ja valinta kattaa osittain metodologian neljä ensimmäistä vaihetta, kun taas tarve hankittavalle ohjelmistolle on määritelty osittain jo johdannossa. Viides vaihe on kuvattu luvussa 5 ja vaiheet 6–7 on rajattu työn ulkopuolelle. Ohjelmiston valintaan liittyviä metodologioita ei ole tarkoitettu liian tiukasti noudatettavaksi, vaan niitä voidaan mukauttaa organisaation tarpeen mukaan [81].

4.1 Ohjelmistojen arviointimenetelmät

Ohjelmiston arviointi on monikriteerinen päätöksenteko-ongelma, jonka ratkaisemiseksi on käytetty eri menetelmiä. Yksi laajasti käytetty menetelmä on analyttinen hierarkiaprosessi (engl. *Analytic Hierarchy Process, AHP*), joka perustuu kriteereistä muodostuvaan hierarkkiseen kehykseen. Hierarkian huipulle sijoitetaan tavoite ja pohjalle arvioitavat vaihtoehdot. Näiden väliin asetetaan kriteerit, jotka voidaan jakaa alakriteereihin. Jokaisella tasolla olevia tekijöitä verrataan toisiinsa pareittain suhteessa ylemmällä tasolla oleviin tekijöihin. Parivertailut tehdään kaikille mahdollisille kombinaatioille ja niiden tulosten pohjalta tekijöille muodostetaan paikalliset prioriteetit. Lopuksi jokaiselle vertailun kohteena olevalle vaihtoehdolle lasketaan suhteelliset pisteet päätöksenteon kohteena olevaan tavoitteeseen nähden. [81]

AHP:n vahvuuksiksi ohjelmiston valinnassa on nähty sen mahdollistama päätöksenteko-ongelman jäsentäminen hierarkiaan, joka voi helpottaa ongelman ymmärtämistä

ja yksinkertaistamista. Sen on myös katsottu soveltuvan hyvin sekä määrällisten että laadullisten ongelmien käsittelyyn. Lisäksi sitä voidaan hyödyntää sekä yksilöiden että ryhmien päätöksenteossa. Parivertailujen määrä sekä suoritettavat laskutoimitukset vievät kuitenkin paljon aikaa ja arvioinnit on suoritettava uudestaan kriteerien tai vaihtoehtojen määrän muuttuessa. Tarkasteltavien vaihtoehtojen määrä vaikuttaa vaihtoehtojen sijoitukseen ja määrän muutokset voivat johtaa lopullisen sijoituksen muuttumiseen. [81]

Muita [81] esiin nostamia menetelmiä ovat ominaisuusanalyysi (engl. *feature analysis*), painotettu pisteytys tai painotetun keskiarvon summa (engl. *weighted average sum*) sekä sumeat menetelmät. Ominaisuusanalyysi on artikkelisarjassa [82]–[85] esitelty menetelmä [81]. Ominaisuusanalyysissa luodaan lista erilaisista ominaisuuksista, jotka on mahdollista hajottaa pienempiin osiin. Tällöin niistä voidaan muodostaa hierarkioita analysoinnin helpottamiseksi. Ominaisuudet voidaan jakaa kahteen tyyppiin sen perusteella, arvioidaanko niitä luokittelu- vai järjestysasteikolla. Luokitteluasteikolla arvioitaviin yksinkertaisiin ominaisuuksiin voidaan liittää arvio ominaisuuden tärkeydestä. Ominaisuus voidaan arvioida esimerkiksi pakolliseksi, jolloin sen puuttuminen johtaa suoraan ohjelmiston karsiutumiseen. Monimutkaisempien järjestysasteikolla arvioitavien ominaisuuksien kohdalla on arvioitava ominaisuuden tärkeyden lisäksi, kuinka hyvin ominaisuus on tuettu tai miten se toteutuu. [83] Ominaisuusanalyysi voidaan toteuttaa joko seulontana, tapaustutkimuksena, muodollisena kokeena tai kyselytutkimuksena. Valittu lähestymistapa vaikuttaa arvioinnin tarkkuuteen ja kustannuksiin. [84] Tulosten analysointiin vaikuttaa käytetty arviointimenetelmä ja se, onko ominaisuuksien hyväksyttävyydelle asetettu eksplisiittistä kynnsarvoa. Jos ominaisuuksille on asetettu jokin kynnsarvo, tarkastellaan kynnsarvon ja ohjelmiston kyseiselle ominaisuudelle saaman pisteytyksen erotusta. Muussa tapauksessa voidaan tarkastella suoraan pistemäärää. Etenkin kokeiden ja kyselyiden tapauksessa voidaan samalle ohjelmistolle saada useita pisteytyksiä. Pisteytysten yhdistämiseksi on ehdotettu eri menetelmiä, mutta pisteiden keskiarvoa ei kuitenkaan tulisi käyttää. [85]

Ominaisuusanalyysiin hyvinä puolina on nähty mahdollisuus vaikuttaa arvioinnin tarkkuuteen toteutustavan valinnan avulla. Ominaisuusanalyysiä voidaan käyttää ohjelmistojen lisäksi ohjelmistotoimittajien arviointiin. Tulosten kokoaminen yhdeksi ohjelmisto-kohtaiseksi kokonaispistemääräksi voi kuitenkin johtaa harhaan, koska samaan lopputulokseen voidaan päästä usealla eri pistekombinaatioilla. [81] Tuloksia onkin suositeltu verrattavaksi ominaisuusryhmittäin [85].

Painotetussa pisteytyksessä kriteereille määritetään painoarvot ja yhteinen arviointias- teikko, jota vasten ohjelmistoja verrataan. Arviointias- teikolla määritetään kuinka hyvin ohjelmisto täyttää kriteerin, ja paino-arvo kuvaa kriteerin tärkeyttä suhteessa muihin kri- teereihin. Ohjelmiston kriteerikohtaiset pisteet saadaan kertomalla arviointias- teikolta annetut pisteet niille kuuluvilla painoarvoilla. Kriteerit on voitu jakaa kategorioihin, jolloin kategoriakohtaiset pisteet saadaan laskemalla kategoriaan kuuluvista kriteereistä saa- dut pisteet yhteen. Ohjelmiston kokonaispistemäärä saadaan yhdistämällä katego- riakohtaiset pisteet. [81] Lähteessä [86] kuvatussa menetelmässä, kategorian sisältä- mien kriteerien pisteistä lasketaan keskiarvo, joka kerrotaan kategoriakohtaisella paino- arvolla.

Painotetun pisteytyksen etuna on nähty sen helppokäyttöisyys. Painoarvojen määrittä- mistä on kuitenkin pidetty hankalana, jos kriteereitä on paljon. Lisäksi sitä voi olla vai- kea soveltaa moniulotteisiin monikriteerisiin päätöksenteko-ongelmiin. Myös pisteytyk- sessä käytetty yhteinen numeerinen asteikko on nähty yhtenä menetelmän heikkou- tena. [81]

Sumeita menetelmiä on käytetty ohjelmistojen arvioinnissa, kun pisteytysten ja painoar- vojen tarkka määrittäminen ei ole mahdollista. Sumeat menetelmät hyödyntävät su- meaa joukko-oppia ihmisten tekemien arvioiden epävarmuuksien mallintamisessa. [81] Sumean joukko-opin soveltaminen päätöksentekoon on pääsääntöisesti tarkoittanut klassisten päätöksenteon teorioiden laajentamista tai sumeuttamista [87]. Yhtenä esi- merkkinä sumeasta päätöksentekomenetelmästä on sumea AHP, jonka yhtä variaa- tiota on sovellettu esimerkiksi lähteessä [88].

Sumeiden menetelmien on katsottu helpottavan vaihtoehtojen sanallista arviointia ja parantavan päätöksentekoa huomioimalla inhimilliseen päätöksentekoon liittyvän epä- määräisyyden. Vertailussa käytettyjen arvojen laskemista kaikille vaihtoehdoille on kui- tenkin pidetty hankalana. [81]

4.2 Ohjelmiston valinta

Tavoitteena oli löytää ohjelmisto, joka mahdollistaisi top-down-suunnitteluprosessin so- veltamisen IEC 61850 –toteutuksessa. Tämä edellyttää, että ohjelmiston tulisi sisältää vähintään standardissa määritetyt SST:n ja SCT:n toiminnallisuudet tai vaihtoehtoisesti samalta toimittajalta tulisi löytyä erilliset nämä toiminnot toteuttavat ohjelmistot esimer- kiksi osana samaa ohjelmistopakettia. Erillisten ohjelmistojen tapauksessa ohjelmiston valitsemisen samalta toimittajalta katsottiin helpottavan ohjelmistojen opettelua ja

käyttöä. Lisäksi tämä voisi yksinkertaistaa ohjelmistojen ylläpitoa, koska tuki ja ohjelmistopäivitykset tulisivat yhdeltä toimittajalta. Ohjelmistolla tai ohjelmistopakettilla olisi pystyttävä luomaan järjestelmäspesifikaatio SSD-tiedoston muodossa, siihen olisi pystyttävä tuomaan ICD- ja IID-tiedostot laitevalmistajakohtaisista työkaluista sekä sillä olisi pystyttävä luomaan SCD-tiedosto. Tämän kriteerin täyttäviä ohjelmistoja löydettiin seitsemän kappaletta.

Löydetyistä ohjelmistoista karsittiin aluksi kaksi pienintä ja vähiten tunnettua ohjelmistoa. Jäljelle jääneiltä ohjelmistotoimittajilta pyydettiin lisätietoja sähköpostitse tai täyttämällä toimittajan nettisivuilta löytyvä yhteydenottolomake. Ohjelmistojen eri ominaisuuksia selvitettiin perehtymällä ohjelmistotoimittajilta saatuihin materiaaleihin. Saadut tiedot vaihtelivat paljon eri toimittajien välillä. Osa toimittajista vastasi kysymyksiin tarkasti, mutta osa ei ilmoittanut pyydettyjä tietoja ollenkaan tai vastasi vasta muutaman kuukauden viiveellä, jolloin toimittajalta saatuja tietoja ei ehditty ottaa arviointiin mukaan. Tämän takia arviointikriteerejä ei voitu asettaa kovin tarkasti. Ohjelmistoista pyrittiin arvioimaan seuraavia asioita: dokumentaatio, ominaisuudet, ylläpidettävyys, kustannukset ja tunnettuus.

Dokumentaatiosta pyrittiin arvioimaan sen kattavuutta ja selkeyttä. Dokumentaatioksi luettiin ohjelmiston manuaalit sekä mahdolliset muut materiaalit, kuten tutoriaalivideot. Dokumentaatiota hyödynnettiin myös ohjelmistojen ominaisuuksien selvittämisessä. Ominaisuuksilla tarkoitetaan tässä yhteydessä kaikkia ohjelmistosta SST- ja SCT-työkalujen perustoimintojen lisäksi löytyviä toimintoja. Erityisen tärkeäksi ominaisuudeksi katsottiin ohjelmiston kyky tuottaa valmista dokumentaatiota IEC 61850 –konfiguraatiosta, sillä vaikka top-down-prosessin hyödyntäminen ei osoittautuisi järkeväksi tällä hetkellä, voitaisiin ohjelmistoa silti hyödyntää näiltä osin ja tämä voisi jo itsessään oikeuttaa ohjelmiston hankinnan. Toinen tärkeä ominaisuus olisi ohjelmassa luotujen spesifikaatioiden tallentamisen ja uudelleenkäyttämisen mahdollisuus, jotta spesifikaatiota ei tarvitsisi joka kerta luoda tyhjästä. Tämän lisäksi katsottiin hyödylliseksi mahdollisuus suorittaa validointeja projektin ulkopuolisille SCL-tiedostoille. Yhdessä arvioituista ohjelmistoista oli myös tuki virtuaalisille IED-laitteille.

Ylläpidettävyydellä tarkoitetaan teknisen tuen ja ohjelmistopäivitysten saatavuutta ohjelmistolisenssin hankinnan jälkeen. Kaikissa arvioituissa ohjelmistoissa tuki ja päivitykset olivat erikseen ostettavissa. Ohjelmistopäivitykset ja tuki saattoivat sisältyä ohjelmistolisenssin hintaan enintään vuoden ajaksi. Yhden ohjelmiston osalta ilmoitettiin, että ohjelmistovirheisiin liittyvät päivitykset ovat maksuttomia. Ohjelmistotoimittajista yksi ilmoitti vastaavansa tukipyyntöihin viimeistään yhden työpäivän kuluessa, yksi

ilmoitti vastaavansa kahden työpäivän kuluessa ja kaksi muuta eivät ilmoittaneet tarkkaa takarajaa. Teknisen tuen kannalta yrityksen sijainti voi myös olla oleellinen tekijä. Kahden ohjelmistotoimittajan toimipisteet sijaitsivat Pohjois-Amerikassa, jolloin tuen saatavuus saattaa aikaeron takia olla heikompi eurooppalaisiin ohjelmistotoimittajiin verrattuna.

Ohjelmistoon liittyvät kustannukset olivat kriteereistä helpoiten arvioitavissa. Kustannukset voitiin jakaa ohjelmiston hankintaan ja ylläpitoon liittyviin kustannuksiin. Hankintaan liittyvät kustannukset koostuivat pääasiassa ohjelmistolisensseistä sekä joissain tapauksissa mahdollisista koulutuksista, jotka on tilattava lisenssiä hankittaessa. Lisenssien hinnoitteluun oli erilaisia lähestymistapoja. Yksittäisen lisenssin hinta yleensä pieneni, kun lisenssejä hankittiin isompi määrä. Lisenssit olivat pääasiassa toistaiseksi voimassa olevia. Yksi toimittajista kuitenkin tarjosi myös tilauspohjaista lisenssiä, joka oli voimassa vuoden hankinnasta. Ylläpitoon liittyvät kustannukset syntyivät ohjelmiston päivityksiin ja tekniseen tukeen liittyvistä sopimuksista. Kustannuksia arvioitiin ohjelmistotoimittajilta saatujen tarjousten ja hintalistojen perusteella.

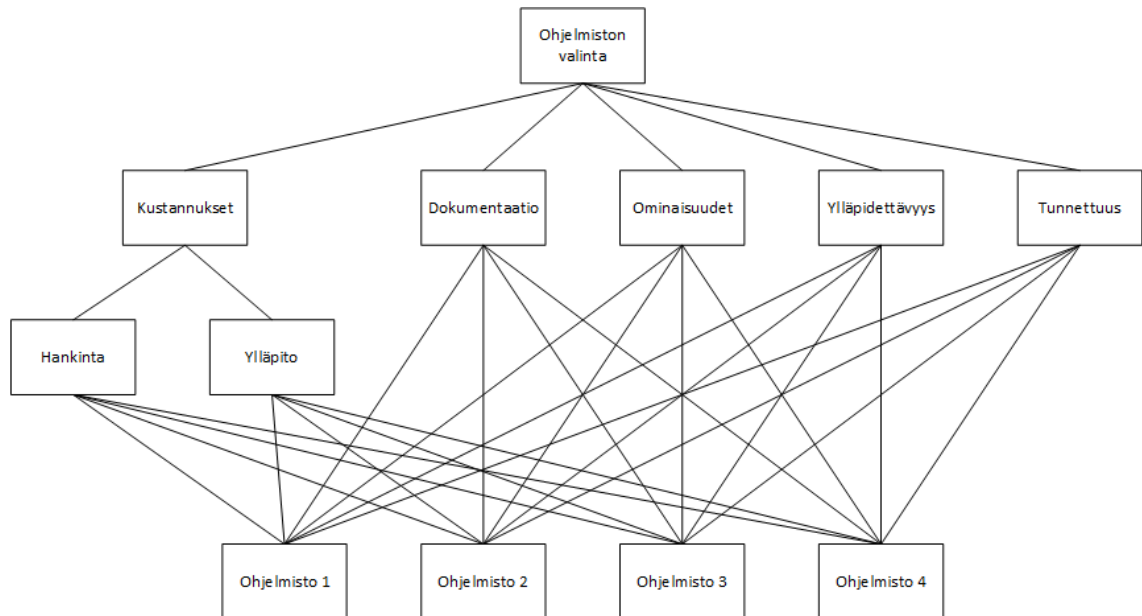
Ohjelmistotoimittajia pyydettiin toimittamaan myös ohjelmistoon liittyviä referenssejä, mutta nämä saatiin aluksi ainoastaan yhdeltä toimittajalta. Referenssit saatiin lopulta toiseltakin toimittajalta, mutta ne eivät ehtineet arviointiin mukaan. Ohjelmiston tunnettuutta pyrittiin arvioimaan tämän lisäksi pitkälti sen pohjalta, kuinka paljon ohjelmisto on ollut esillä alan tapahtumissa ja onko esimerkiksi tiedossa tapauksia, joissa ohjelmisto on ollut käytössä jonkin toisen yhtiön projektissa.

Edellisten lisäksi myös käytettävyyttä pidettiin ohjelmiston lopullisen valinnan kannalta tärkeänä kriteerinä. Käytettävyyttä on kuitenkin hyvin hankala arvioida kokeilematta ohjelmistoa käytännössä, joten sitä ei sisällytetty tässä vaiheessa mukaan vertailuun.

Ohjelmistojen vertailuun päätettiin käyttää analyttistä hierarkiaprosessia. Menetelmän etuna nähtiin erityisesti parivertailu, jonka katsottiin helpottavan arviointia. Menetelmää voitiin pitää myös riittävän tarkkana, koska arvioinnin oli tarkoitus toimia vain suuntaa antavana ja sitä pidettiin sumeisiin menetelmiin nähden vähemmän työläänä. Koska yksi valituista viidestä ohjelmistotoimittajasta ei toimittanut pyydettyjä lisätietoja ollenkaan eikä tarvittavia tietoja ollut julkisesti saatavilla, vertailuun otettiin mukaan neljä ohjelmistoa.

Ohjelmistojen vertailu aloitettiin muodostamalla kriteereistä ja ohjelmistoista hierarkia, joka on esitetty kuvassa 19.. Hierarkian alimman osan muodostavat vertailtavat ohjelmistot ja keskellä ovat kriteerit, joista kustannukset on jaettu kahteen alikriteeriin.

Hierarkian huipulla on tavoite, joka on tässä vertailussa ohjelmiston valinta. Ohjelmiston valintaa ei kuitenkaan tehdä vielä pelkästään tämän vertailun tulosten pohjalta, vaan tavoitteena on asettaa ohjelmistot paremmuusjärjestykseen.



Kuva 19. Ohjelmistojen vertailua varten muodostettu AHP hierarkia.

Hierarkian luomisen jälkeen tehtiin parivertailut kriteereille. Vertailussa käytettiin Saaty'n asteikkoa, joka on esitetty taulukossa 2. Erilaisia vertailussa käytettäviä asteikkoja on kuitenkin ehdotettu useita ja niiden paremmuus jakaa mielipiteitä. [89, s. 15] Suoritettujen parivertailun pohjalta muodostettiin parivertailumatriisi. Parivertailumatriisi **A** on neliömatriisi, jossa jokainen alkio on arvio kahden painon suhteesta $a_{ij} \approx \frac{w_i}{w_j} \forall i, j$ [89, s. 5].

Taulukko 2. Saaty'n asteikko. Perustuu lähteeseen [89, s. 15].

Sanallinen kuvaus	Asteikko
Ei eroa	1
-	2
Kohtalainen preferenssi	3
-	4
Voimakas preferenssi	5
-	6
Erittäin voimakas tai osoitettu preferenssi	7
-	8
Äärimmäinen preferenssi	9

Prioriteettivektorin johtaminen parivertailumatriisista on analyttisessä hierarkiaprozessissa keskeinen vaihe. Prioriteettivektorin sisältämät arvot, joita kutsutaan prioriteeteiksi tai painoiksi, kuvaavat vaihtoehtojen paremmuutta toisiinsa nähden. Suurempi

prioriteetti tarkoittaa, että vaihtoehto on pienemmän prioriteetin omaavia vaihtoehtoja parempi. Jos parivertailu olisi tehty täysin rationaalisesti, vastaisi parivertailumatriisin jokainen alkio täsmälleen kahden painon suhteesta ja prioriteettivektori saataisiin normeeraamalla mikä tahansa matriisin sarake. Tämä ei ole kuitenkaan yleistä ja prioriteettivektori on pystyttävä estimoimaan muilla tavoin. Tähän on esitetty useita menetelmiä, joista jokainen voi johtaa erilaiseen lopputulokseen. Suosituin menetelmä on ominaisarvomenetelmä, jonka mukaan prioriteettivektori \mathbf{w} , saadaan mistä tahansa parivertailumatriisista \mathbf{A} , ratkaisemalla yhtälö 1, jossa λ_{\max} on parivertailumatriisin suurin ominaisarvo. [89, s. 4,17–19]

$$\begin{cases} \mathbf{A}\mathbf{w} = \lambda_{\max}\mathbf{w} \\ \mathbf{w}^T\mathbf{1} = 1 \end{cases} \quad (1)$$

Toinen yleisesti käytetty tapa on geometrisen keskiarvon menetelmä. Sen mukaan jokainen prioriteettivektorin alkio w_i saadaan sitä vastaavan parivertailumatriisin rivin alkioiden geometrisesta keskiarvosta, jaettuna normeeraustermillä, jolloin vektorin alkioiden summa on 1. Normeeraustermillä tarkoitetaan kaavassa 2 esiintyvää jakajaa. [89, s. 19]

$$w_i = (\prod_{j=1}^n a_{ij})^{\frac{1}{n}} / \sum_{i=1}^n (\prod_{j=1}^n a_{ij})^{\frac{1}{n}} \quad (2)$$

Ominaisarvomenetelmän soveltamiseen on esitetty erilaisia tapoja. Esimerkiksi lähteessä [90] on käytetty iteratiivista potenssimenetelmää, jossa parivertailumatriisi kerrotaan itsellään, jonka jälkeen rivit summataan ja normeerataan. Tästä saatu vektori on ensimmäinen arvio prioriteettivektorista. Vaiheita toistetaan, kunnes kahden perättäisen arvion välinen ero alittaa keskeytyskriteerin. Tässä työssä prioriteettivektori muodostettiin ratkaisemalla parivertailumatriisin suurinta ominaisarvoa vastaava ominaisvektori Matlab-ohjelmistolla ja normeeraamalla se alkioidensa summalla.

Epäjohdonmukaisuuden mittaamiseen on ehdotettu johdonmukaisuusindeksiä CI

$$CI(\mathbf{A}) = \frac{\lambda_{\max} - n}{n-1} \quad (3)$$

jossa n on matriisin koko. Johdonmukaisuusindeksi perustuu tulokseen, jonka mukaan parivertailumatriisin $\lambda_{\max} = n$, jos ja vain jos matriisi on yhtenäinen. Numeerisissa tutkimuksissa on kuitenkin havaittu, että satunnaisen $n + 1$ kokoisen matriisin johdonmukaisuusindeksin odotusarvo on suurempi kuin satunnaisen n kokoisen matriisin johdonmukaisuusindeksin odotusarvo. Tämän vuoksi johdonmukaisuusindeksi skaalataan

satunaisindeksillä RI , joka on arvio keskimääräisestä johdonmukaisuusindeksistä n -koisilla satunnaismatriiseilla. Johdonmukaisuussuhde CR saadaan kaavasta

$$CR(A) = \frac{CI(A)}{RI_n} \quad (4)$$

jossa RI_n vastaa taulukossa 3 esitettyä satunaisindeksin arvoa. Vakiintunut tapa on hylätä matriisi, jos johdonmukaisuussuhde on yli 0.1 ja arvioida päätökset uudelleen. [89, s. 24–25]

Taulukko 3. Satunaisindeksin arvot eri kokoisille matriiseille [89, s. 25].

n	3	4	5	6	7	8	9	10
RI_n	0,5247	0,8816	1,1086	1,2479	1,3417	1,4057	1,4499	1,4854

4.3 Arvioinnin tulokset

Arviointi toteutettiin muodostamalla parivertailumatriisi ensin kriteereistä ja määrittämällä tälle painovektori sekä johdonmukaisuusindeksi. Kriteereistä muodostettu parivertailumatriisi ja sille määritetty painovektori on esitetty taulukossa 4. Parivertailumatriisi muodostettiin vertaamalla ensin ensimmäiselle riville asetettua kriteeriä kaikkiin muihin kriteereihin ja tämän jälkeen seuraavan rivin kriteeriä niihin kriteereihin, joihin sitä ei ollut vielä verrattu, kunnes kaikki kombinaatiot oli käyty läpi. Matriisin alkiot muodostuvat siis kyseisen rivin vasemmanpuolimmaisessa sarakkeessa olevan kriteerin ja alkion sarakkeen ensimmäisellä rivillä olevan kriteeriin painojen suhteesta. Esimerkiksi ominaisuuksien tärkeyttä painotettiin Saatyn asteikon mukaisesti voimakkaasti dokumentaation nähden, jolloin taulukon kolmannen rivin toisessa sarakkeessa on luku 5 ja vastaavasti toisen rivin kolmannessa sarakkeessa sen käänteisluku 1/5. Näin ollen parivertailumatriisin päädiagonaali muodostuu aina ykkösistä, koska esimerkiksi tässä tapauksessa kriteeri on aina itseensä nähden saman arvoinen. Matriisille saatu johdonmukaisuusindeksi oli 0,032 ja johdonmukaisuussuhde 0,029. Taulukon oikeanpuolimmaisessa sarakkeessa on esitetty matriisille määritetty prioriteettivektori, jonka alkiot on pyöristetty kolmen desimaalin tarkkuuteen. Prioriteettivektorista nähdään, että ominaisuuksia pidettiin ohjelmistovertailun kannalta selkeästi tärkeimpänä kriteerinä ja tunnettuutta vähiten tärkeänä.

Taulukko 4. Kriteereistä muodostettu parivertailumatriisi ja painovektori.

Kriteeri	Dokumentaatio	Ominaisuudet	Ylläpidettävyys	Kustannukset	Tunnettuus	Prioriteettivektori
Dokumentaatio	1	1/5	1	1/3	3	0,102
Ominaisuudet	5	1	5	3	7	0,504
Ylläpidettävyys	1	1/5	1	1/3	3	0,102
Kustannukset	3	1/3	3	1	5	0,245
Tunnettuus	1/3	1/7	1/3	1/5	1	0,046

Koska kustannukset oli jaettu alakriteereihin, myös niiden vertailua varten muodostettiin parivertailumatriisi. Vertailu olisi voitu kuitenkin tehdä helposti myös ilman parivertailumatriisia, koska suoritettavia parivertailuja oli ainoastaan yksi. Hankintakustannuksia pidettiin Saatyn asteikolla arvioituna erittäin voimakkaasti ylläpitokustannuksia tärkeämpänä kriteerinä.

Kriteerien arvioinnin jälkeen suoritettiin kriteeri- tai alakriteerikohtaiset vertailut jokaiselle ohjelmistolle. Jos kriteeri on jaettu alakriteereihin, ei vertailuja suoriteta erikseen varsinaiselle ylemmän tason kriteerille. Näin ollen ohjelmiston vertailua varten muodostettiin kuusi parivertailumatriisia. Mikäli matriisin johdonmukaisuusindeksi oli liian suuri, päätöksiä arvioitiin tarvittaessa uudestaan. Lopuksi laskettiin kriteerien ja ohjelmistojen kriteerikohtaisten prioriteettien tulot, jotka on esitetty taulukossa 5. Ohjelmistojen kriteerikohtaisen vertailun prioriteettivektori saadaan siis jakamalla taulukon kriteerikohtainen tulos vastaavalla taulukon 4 prioriteettivektorin alkiolla. On kuitenkin huomattava, että kustannuskriteerikohtaiset tulokset on kerrottu myös alakriteerikohtaisilla prioriteeteilla. Ohjelmistojen lopulliset kokonaisprioriteetit on esitetty taulukon oikeanpuolimmaisessa sarakkeessa ja ne saatiin summaamalla rivit yhteen. Tulos on parempi, mitä suurempi luku on. Koska ohjelmiston ominaisuudet arvioitiin muita kriteereitä selkeästi tärkeämmäksi, ohjelmistojen sijoitus tässä kriteerikohtaisessa vertailussa vaikuttaa huomattavasti lopputulokseen. Sen sijaan esimerkiksi ylläpitokustannusten vaikutus on hyvin vähäinen. Tulosten pohjalta kokeiluun valittiin Helinks STS ja Kalkitech ASE SCL Manager.

Taulukko 5. Ohjelmistovertailun tulokset.

Ohjelmisto	Dokumentaatio	Ominaisuudet	Ylläpidettävyys	Hankintakustannukset	Ylläpito-kustannukset	Tunnettuus	Yhteensä
1	0,032	0,214	0,040	0,010	0,002	0,017	0,316
2	0,032	0,214	0,007	0,044	0,005	0,006	0,308
3	0,006	0,028	0,040	0,117	0,018	0,006	0,215
4	0,032	0,047	0,016	0,044	0,006	0,017	0,161

5. SÄHKÖASEMA-AUTOMAATIOJÄRJESTELMÄN KONFIGUROINTI UUDEN SUKUPOLVEN TYÖKALUILLA

Helinks STS:sää ja SCL Manageria päätettiin testata luomalla todelliseen sähköasematoimitukseen pohjautuva testiprojekti. Testiprojektin pohjaksi valittu sähköasematoimitus oli 2PK-järjestelmällä (*kaksoispääkisko*) toteutettu 110 kV:n GIS-kytkinlaitos, jonka kenttien määrää vähennettiin alkuperäisestä. Lisäksi sähköaseman nimi, kenttien nimet ja tunnukset muutettiin. Sähköaseman pääkaavio on esitetty liitteessä A. Kiskoilla AEW1 ja AEW2 on yhteinen kenttäohjausyksikkö. Kisko- ja katkaisijavikasuojaus rajattiin toteutuksen ulkopuolelle. Kentät AE02 ja AE04 ovat identtisiä ja niiden suojaus on toteutettu kahdella distanssireleellä. AE03-kentän suojaus on toteutettu yhdellä distanssireleellä ja johtodifferentiaalireleellä. Johtolähtökenttien ohjaukset on toteutettu ominaisuuksiltaan samanlaisilla kenttäohjausyksiköillä. AE05 on kiskokatkaisijakenttä, johon ei liity suojaustoimintoja. Pääkaaviossa erottimena piirretty ja myöhemmin myös erottimena mallinnettu Q7 ei ole käytännössä erotin vaan eräänlainen erotusväli.

Testiprojektin pohjana toiminut sähköasematoteutus perustui kokonaisuudessaan GE:n IED-laitteisiin. Testiprojektissa jokainen kenttä haluttiin kuitenkin toteuttaa eri laitevalmistajien IED-laitteilla. Tällä tavoin oli mahdollista saada kokemusta siitä, miten eri laitevalmistajien ohjelmistot tukevat top-down-prosessia ja mitä asioita on otettava huomioon eri laitevalmistajien laitteita konfiguroitaessa. Laitteita ei ollut käytettävissä testijärjestelmän rakentamiseksi, joten konfiguraatioiden lataamista laitteeseen ja laitteiden välistä tietoliikennettä ei voitu testata. Käytetyt laitetypit ja ICT-ohjelmistot on esitetty taulukossa 6.

Taulukko 6. Testiprojektiin valitut IED-laitteet ja ohjelmistot

Kenttä	Laitetun- nus	Laittevalmis- taja	Laitetyyppi	ICT
AEW	A1	Siemens	SIPROTEC 6MD85	Digsi 5 V9.00
AE02	A1 F1 F2	Hitachi	RELION REC670 RELION REL670 RELION REL670	PCM 2.10
AE03	A1 F1 F2	Siemens	SIPROTEC 6MD85 SIPROTEC 7SA86 SIPROTEC 7SD86	Digsi 5 V9.00
AE04	A1 F1 F2	GE	MULTILIN C30 MULTILIN D60 MULTILIN D60	UR Setup 8.21
AE05	A1	Hitachi	RELION REC670	PCM 2.10

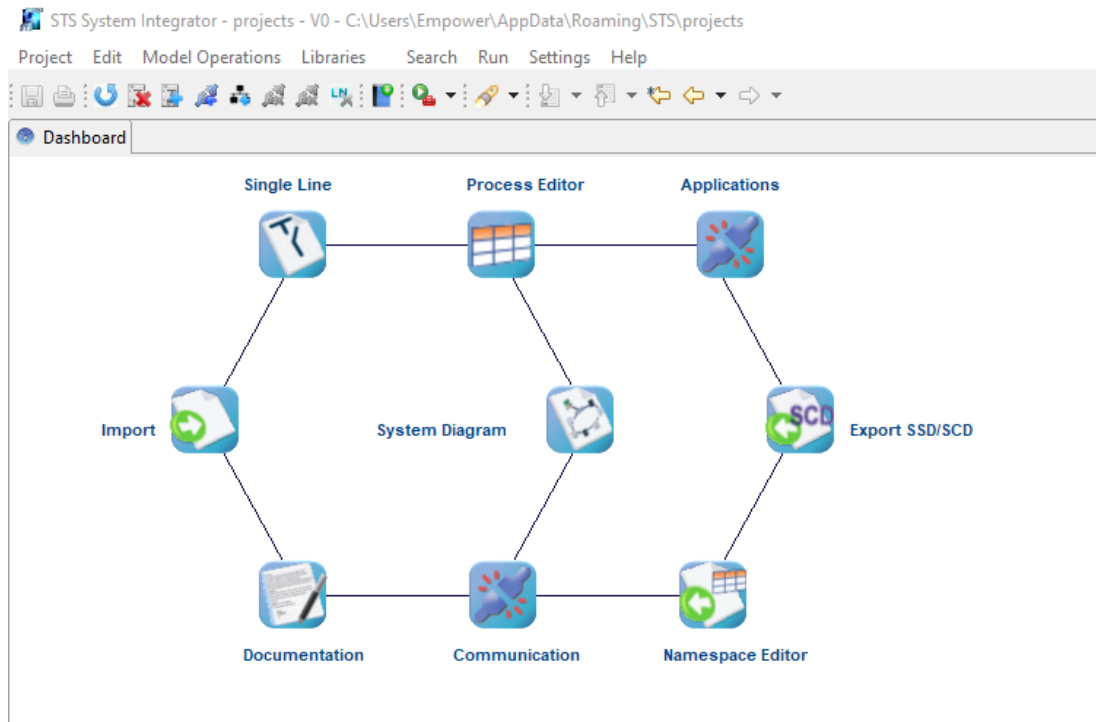
5.1 Tyypikonfiguraatit ja ICD-tiedostot

Konfigurointiprosessissa hyödynnettiin erään asiakkaan olemassa olevia tyypikonfiguraatioita. Tyypikonfiguraatioista luotiin valmistajakohtaiset releprojektit lukuun ottamatta GE:n laitteita, joiden konfiguraatit tallennetaan CID-tiedostoihin. Tyypikonfiguraatit oli tehty 2PK+AK (*kaksoispääkisko-apukisko*) avokytkinlaitokselle, joten ne oli ensin muokattava 2PK GIS-kytkinlaitokseen sopiviksi. Tämä tarkoitti lähinnä tyypikonfiguraatioista löytyvien logiikkaohjelmien ja tietomallien muokkaamista.

Konfigurointiprosessissa käytettiin laitevalmistajien konfigurointiohjelmistoista tallennettuja ICD-tiedostoja ja GE:n laitteiden tapauksessa IID-tiedostoja. Eri laitevalmistajilla oli hieman erilaiset periaatteet ICD-tiedostojen luomisessa. GE:n laitteiden ICD-tiedostojen sisältö pysyy muuttumattomana huolimatta siitä, onko IEC 61850 –tietomallia muutettu konfiguraatiossa ja tiedosto sisältää aina samat tiedot. Tietomallia ei kuitenkaan onnistuttu muuttamaan UR Setupin ulkopuolella ja konfiguraation pohjana jouduttiin käyttämään IID-tiedostoja, jotka sisälsivät tietomalliin tehdyt muutokset. Hitachin ja Siemensin laitteiden ICD-tiedostojen sisältö muuttuu konfiguraation mukana. Esimerkiksi Hitachin tapauksessa ICD-tiedosto sisältää ainoastaan konfiguraatiossa määritetyt loogisten solmujen instanssit, kun taas GE:n ICD-tiedostot sisältävät kaikki konfiguroitavissa olevat loogisten solmujen instanssit. Hitachin ICD-tiedostoihin on mahdollista sisällyttää myös datasetit sekä raportit ja GOOSE-ohjauslohkot. Myös Siemensin IED-laitteiden ICD-tiedostot saattavat sisältää oletuksena joitain raportteja.

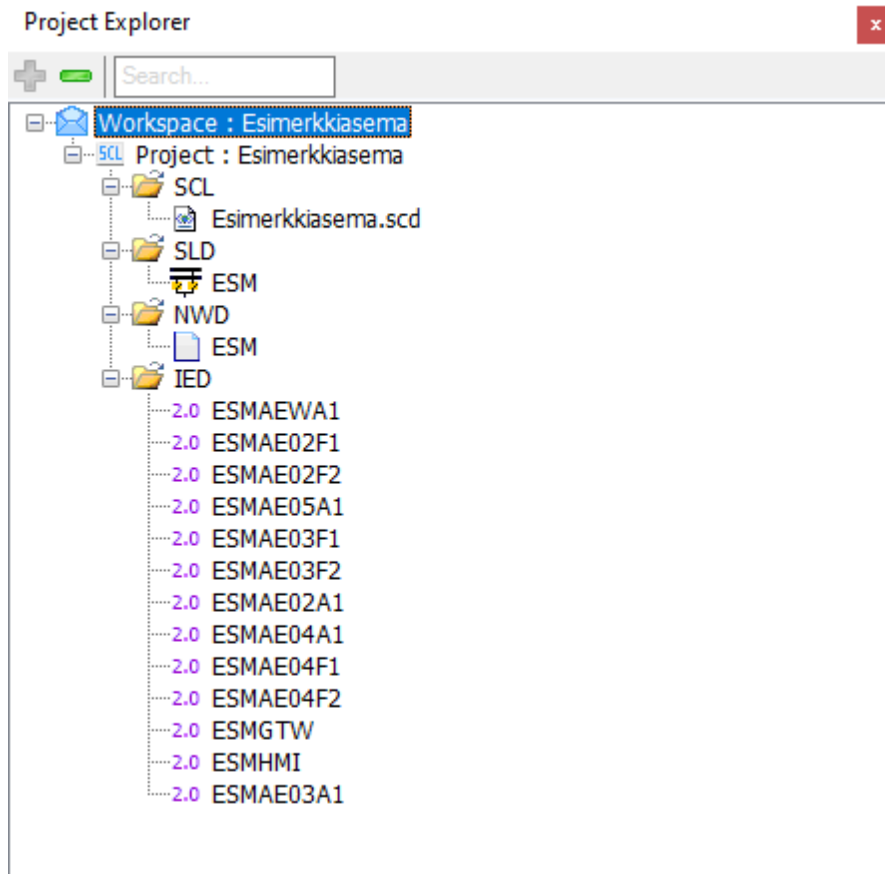
5.2 Konfigurointiprosessi

Molemmilla ohjelmistoilla toteutettuna konfigurointiprosessi noudattaa pitkälti samaa kaavaa. Aluksi luodaan projekti tai työtila, jonka jälkeen voidaan luoda järjestelmäspesifikaatio ja järjestelmäkonfiguraatio. Kuvassa 20 on esitetty Helinks STS työtilan aloitusvalikko. Helinks STS:ssä konfigurointi aloitetaan sähköaseman tapauksessa luomalla pääkaavio *Single Line* –näkyvästä ja liittämällä siihen halutut toiminnot tai luomalla standardissa määritetty prosessiosio *Process Editor* –näkyvästä. Prosessieditori on aktivoitava ohjelmaan erikseen lisäämällä ohjelman pikakuvakkeen käynnistyskomenttoon ylimääräinen parametri. Helinks STS mahdollistaa myös esimerkiksi GOOSE- ja raportointisovellusten määrittelyn jo tässä vaiheessa *Applications*-näkyvässä. Varsinainen järjestelmäkonfiguraation luominen voidaan aloittaa lisäämällä työtilaan IED-laitteiden ICD-tiedostot *Import*-valikosta. Import-valikon kautta voidaan tuoda SCL-tiedostojen lisäksi myös XLS-tiedostoja projektiin. Tämän jälkeen *System Diagram* –näkyvässä voidaan luoda IED-instanssit ja liittää ne lähiverkkoon. *Communication*-näkyvässä on mahdollista tarkastella ja muokata IED-laitteiden tietomalleja, datasettejä ja ohjauslohkoja. *Export SSD/SCD* –valikossa voidaan luoda erilaisia SCL- ja XLS-tiedostoja. *Namespace Editor* –mahdollistaa omien nimiavaruuksien luomisen. *Documentation*-valikosta voidaan luoda PDF-muotoinen projektiraportti.



Kuva 20. Helinks STS –työtilan aloitusvalikko.

SCL Manager –projekti koostuu aluksi pelkän otsakkeen (*header*) sisältävästä SCD-tiedostosta. Projektiin on mahdollista lisätä sisältöä kuvassa 21 esitetyn *Project Explorer* –näkymän kautta joko suoraan tai erilaisten ohjattujen toimintojen avulla. Konfigurointi voidaan aloittaa lisäämällä projektiin pääkaavio, johon lisätty sisältö näkyy suoraan projektin SCD-tiedoston substation-osiossa. Järjestelmän konfigurointi voidaan aloittaa lisäämällä projektiin IED-laitteita esimerkiksi ICD-tiedostoista ja luomalla niiden tietoliikennemäärittelyt. Helinksistä poiketen SCL Managerissa ei ole mahdollista luoda sovellusspesifikaatioita vaan esimerkiksi GOOSE-sovellukset on konfiguroitava suoraan IED-laitteelle. *Project Explorer* –näkymästä voidaan lisätä projektiin myös väyläkaavio sekä mitä tahansa muita yleisiä tiedostoja. IED-laitteiden alle voidaan lisätä myös osaluetteloita tai laitteen liitintietoja. Projektiin on mahdollista luoda myös esitettyä tarkempi kansiorakenne, jolloin esimerkiksi IED-laitteet voidaan järjestää esimerkiksi jännitetason ja kentän mukaan.

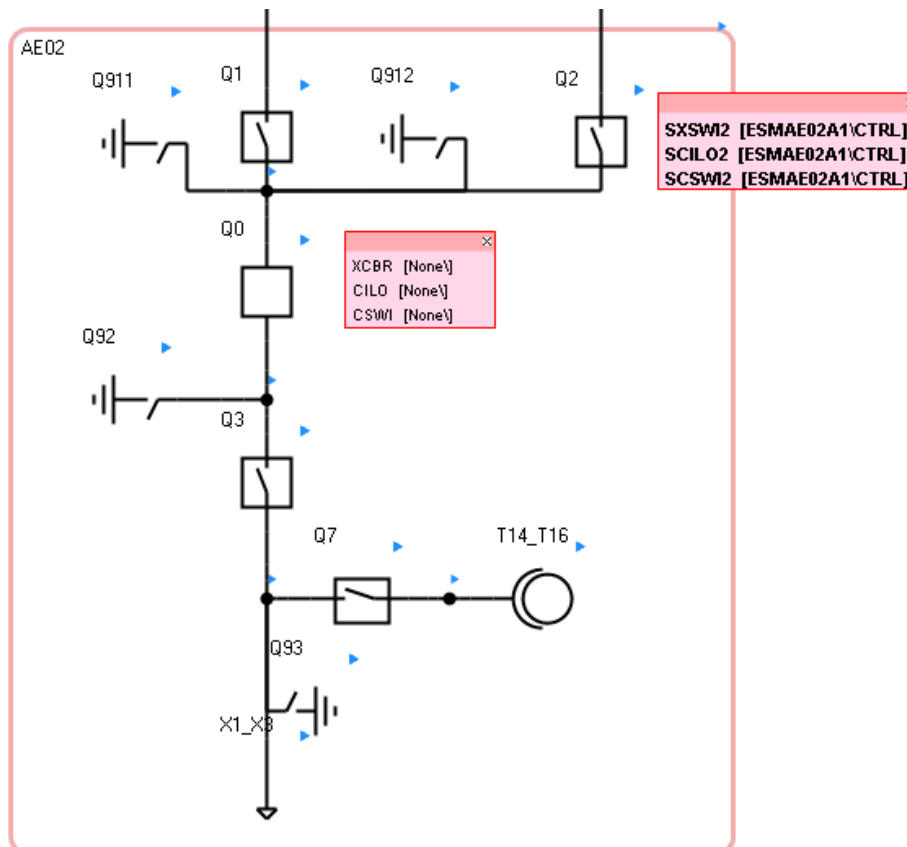


Kuva 21. SCL Managerin Project Explorer –näkyvä.

5.3 Järjestelmäspesifikaatio

Helinks STS:ssä pääkaavio rakennetaan *Single Line* –näkyvässä, jossa luodaan ensin sähköasemarakenne ja tämän alle tarvittavat jännitetasot standardissa määritetyn hierarkian mukaisesti. Jännitetasojen alle voidaan luoda kiskot ja tarvittavat kentät. Kiskoa luotaessa ohjelma luo automaattisesti kiskon nimeä kantavan kentän sekä *connectivity noden*. Kenttien alle voidaan rakentaa sähköaseman pääkaaviota vastaava kuvaus ensiöjärjestelmästä käyttämällä ohjelmiston kirjastosta löytyviä ensiökomponentteja. Muuntajat voidaan tästä poiketen sijoittaa myös hierarkiassa suoraan sähköaseman alle. Komponentit yhdistetään connectivity nodejen avulla. Kuvassa 22 on esitetty osa malliprojektin pääkaaviosta piirrettynä Helinks STS:llä.

SCL Managerissa pääkaavio rakennetaan samaan tapaan kuin Helinksissä eli ensin luodaan sähköasemahierarkia, jonka alle voidaan lisätä jännitetasot ja kentät. Kuvassa 23 on esitetty AE02-kenttä SCL Managerilla piirrettynä. Helinksistä poiketen ensiökomponentteihin liittyvät loogiset solmut saadaan näkyviin suoraan pääkaavioonäkymään. Kuvassa Q2-erottimeen on linkitetty siihen liittyvät loogiset solmut ESMAE02A1-IED:stä, mutta katkaisijaan Q0 ei ole tehty linkityksiä. Muita kuin suoraan ensiölaiteeseen liittyviä toimintoja ei saa näkyviin pääkaavioon, vaan niitä voidaan tarkastella erillisessä puunäkymässä.

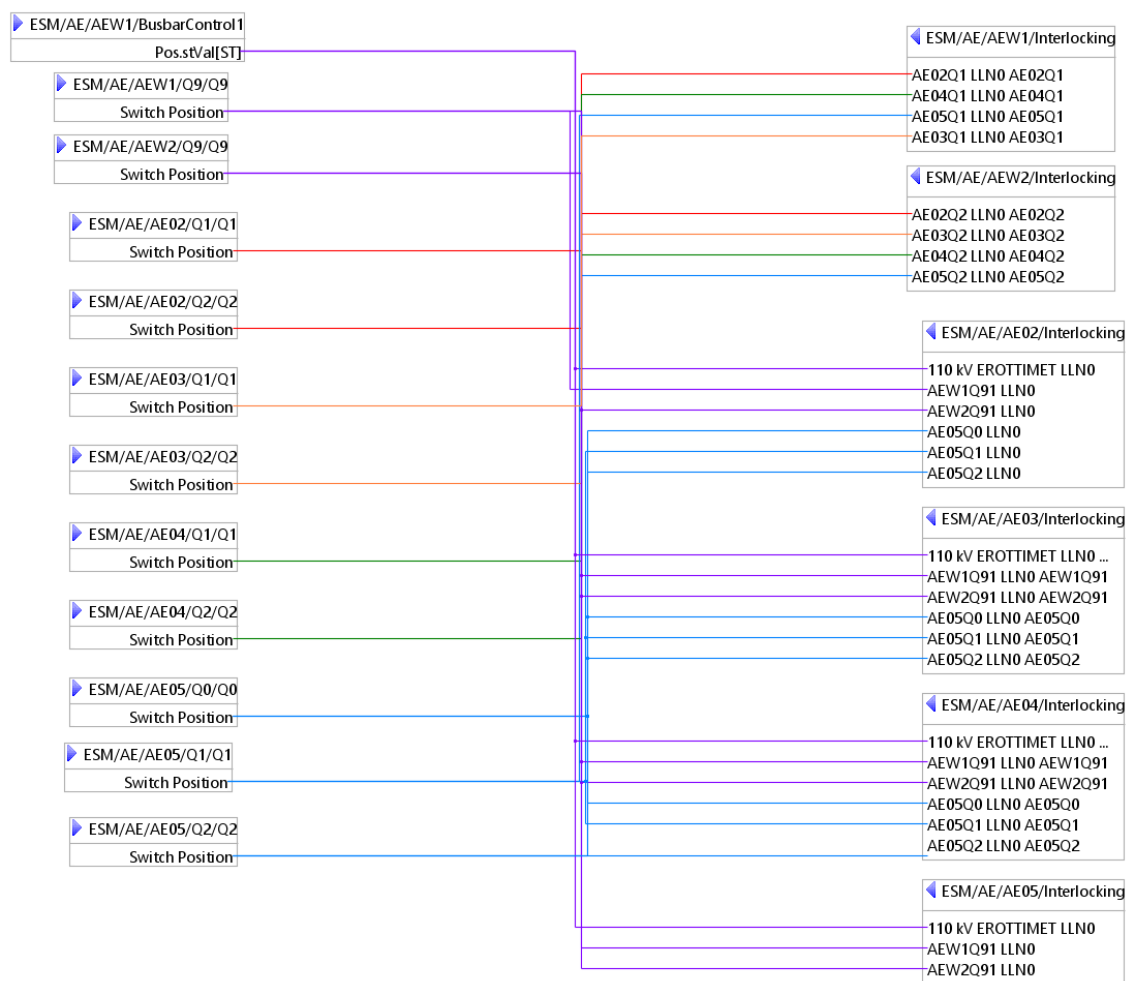


Kuva 23. Kenttä AE02 SCL Managerilla piirrettynä.

Myös SCL Managerissa kenttiä, jännitetasoja tai kokonaisia sähköasemia on mahdollista tallentaa mallipohjaksi uudelleenkäyttöä varten. Mallipohjiin tallentuu kuitenkin ainoastaan kyseisen hierarkian alle määritetyt ensiökomponentit ja niiden liitännät. Komponenttien nimet, kuvaukset, loogiset solmut tai esimerkiksi kentän alle määritetyt toiminnallisuudet eivät tallennu mallipohjaan, joten mallipohjien hyödyntäminen ei ole yhtä tehokasta kuin Helinksissä. Esimerkiksi kenttiä on kuitenkin mahdollista monistaa myös kopioidulla tai kloonamalla, jolloin kentän alle määritettyjen komponenttien nimet, loogiset solmut ja toiminnot säilyvät. Kuvaukset sekä loogisten solmujen prefixit, instanssi-numerot sekä loogisten laitteiden nimet on kuitenkin määritettävä uudestaan. Toinen

huomionarvoinen puute käyttöliittymässä on, että se ei sisällä lainkaan *undo- tai redo-*toimintoja.

Helinksissä luodut toiminnot on mahdollista liittää virtuaalisiin IED-laitteisiin, jotka voidaan korvata varsinaisilla IED-laitteilla järjestelmää konfiguroitaessa. Myös GOOSE-, SV- ja raportointisovellukset voidaan määrittää jo spesifointivaiheessa ja myös ne voidaan tallentaa mallipohjiksi myöhempää käyttöä varten. Kuvassa 24 on esitetty Helinkillä luotu GOOSE-lukitussovellus. Kuvassa vasemmalla on GOOSE:ja lähettävät toiminnot ja oikealla vastaanottavat. Eri ohjauslohkoihin liittyvät GOOSE-sanomat on värikoodattu, jotta ne on helpompi erottaa toisistaan.



Kuva 24. GOOSE-sovellus Helinks STS:ssä.

5.4 Järjestelmäkonfiguraatio

SCL Managerin työtilaan on mahdollista tuoda IED-laitteita joko suoraan SCL-tiedostosta tai ohjelmaan sisältyvästä IED-tietokannasta. IED-laitteita on mahdollista myös luoda itse. IED-tietokanta sisältää valmiiksi useita eri laitevalmistajien IED-malleja,

mutta siihen voidaan myös lisätä uusia laitteita SCL-tiedostoista. Tietokantaan on mahdollista määrittää laitteisiin liittyviä tietoja ja niitä pystytään suodattamaan niiden sisältämien toimintojen avulla. IED:n lisäämistä tietokannasta työtilaan ei kuitenkaan saatu työn aikana toimimaan.

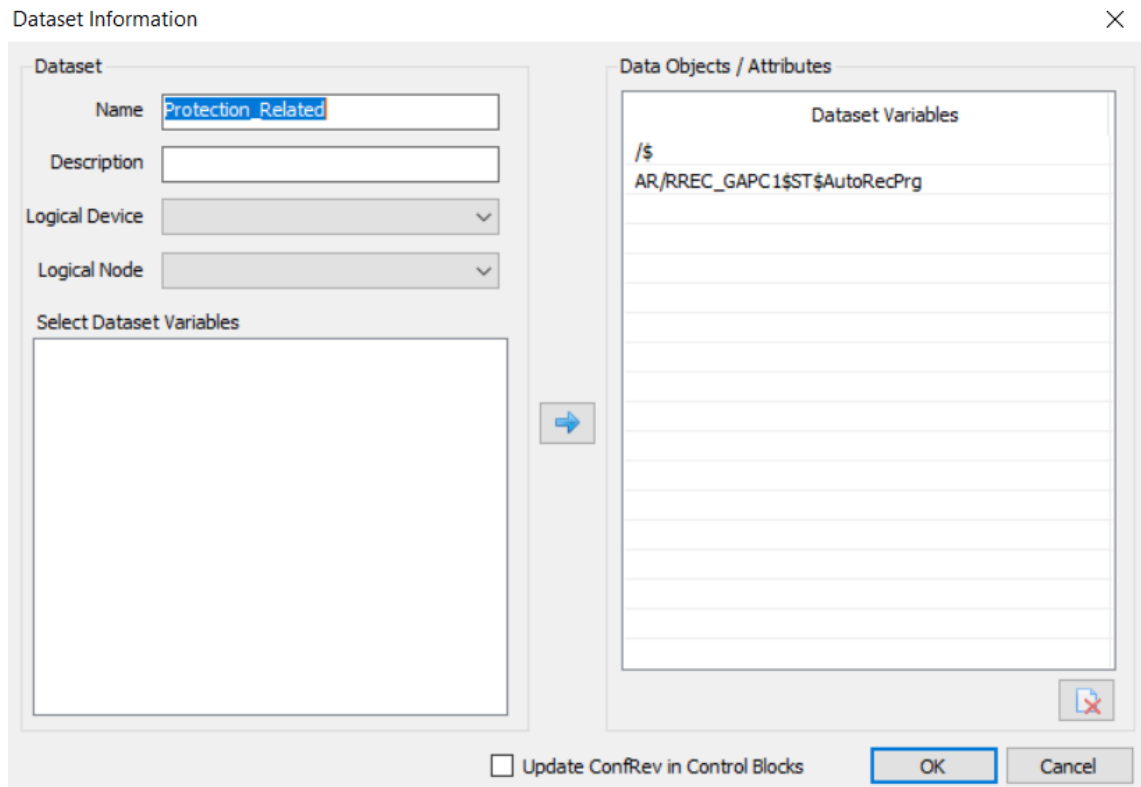
Kun IED-laitteet on lisätty projektiin, niille voidaan tehdä tarvittavat tietoliikennemäärittelyt sekä luoda datasetit, raportit ja GOOSE:t. Datasettien ja ohjauslohkojen määrittäminen yksitellen voi viedä paljon aikaa, eikä niitä voi suoraan kopioida esimerkiksi samanlaiselta laitteelta toiselle. Määrittelyitä on kuitenkin mahdollista viedä IED:stä toiseen SCL Managerin *Replace/Update* –toiminnolla. Uuden laitteen voi tuoda työtilaan myös IID-tiedostona, mikä helpottaa laitemäärittelyiden monistamista kentästä toiseen.

Helinks mahdollistaa oletusarvojen määrittämisen esimerkiksi raporteille, mikä nopeuttaa konfigurointia, jos raporttien asetelut eivät poikkea kovin paljon toisistaan. SCL Managerissa tämä ei ole mahdollista, jolloin jokaiselle raportille on aseteltava erikseen esimerkiksi samat TrgOps- ja puskurointi-aikamäärittelyt.

SCL Managerin *Replace/Update* –toimintoa kokeiltiin tuomalla Hitachin ja Siemensin IED:t uuteen työtilaan ensin IID-tiedostona ja tämän jälkeen päivittämällä niiden tietomallit viemällä datasetit päivitetystä ICD-tiedostoista luotuihin laitteisiin. Siemensin suojarkeitä päivitettäessä huomattiin, että yksittäisiä dataobjekteja saattoi jäädä puuttumaan datasetistä, vaikka vastaava tieto löytyi myös uudesta tietomallista. Kenttäohjausyksikön tapauksessa kahteen datasettiin jäi puuttuvien tietojen paikalle tyhjiä viittauksia:

```
<FCDA IdInst="" lnClass="" fc="" />
```

Puuttuvien tietojen paikalla oli SCL Managerin *Dataset Information* –ikkunassa merkintä /\$, kuten kuvasta 25 nähdään. Merkinnät tulivat näkyviin toisessa tyhjiä viittauksia sisältäneessä datasetissä vasta sen jälkeen, kun toinen dataset oli ensin poistettu ja lisätty uudelleen. Aluksi toisen datasetin tyhjä viittaus olivat nähtävissä vain SCL-koodista ja ne tulivat esille tiedostoa validoitaessa. Merkintöjen poistaminen datasetistä ei poistanut tyhjää viittausta SCL-koodista. Sen sijaan toisessa datasetissä merkintöjen poistaminen aiheutti sen, että tyhjä viittaus siirtyivät SCL-koodissa datasetin alkuun. Hitachin IED-laitteiden kanssa vastaavaa ongelmaa ei tullut esille.



Kuva 25. Tyhjä FCDA-viittaus SCL Managerin Dataset information –ikkunassa.

Uudesta konfiguraatiosta oli poistettu kokonaan yksi erotin ja siihen liittyvät tiedot, mikä selittää osan puuttuvista tiedoista. Vertaamalla kenttäohjausyksikön ICD-tiedostoja toisiinsa Notepad++:n vertailutoiminnolla havaittiin, että joidenkin lukituksiin liittyvien loogisten solmujen InType oli muuttunut. Tämä oli luultavasti seurausta erottimen poistosta, koska poistetun CILO7-solmun InType oli uudessa tiedostossa CILO11-solmulla. Kaikilla CSWI-solmuilla oli tietomallissa sama InType, joten näihin erottimen poistaminen ei aiheuttanut muutoksia. Kuitenkin myös osa erottimien tilatiedoista oli kadonnut yhdestä datasetistä. Lisäksi kuvassa 25 esitetyn Protection_Related datasetistä puuttuvassa SYNC/RSYN_GAPC1\$ST\$/SynPrg-tiedossa ei huomattu eroavaisuuksia ICD-tiedostojen välillä. Kun ICD-tiedostoja verrattiin toisiinsa SCL Managerin *Compare IED Models* –toiminnolla tietomalleista ei löydetty poikkeavuuksia. Myös Hitachin kenttäohjausyksikön konfiguraatiosta oli poistettu aiemmin yksi erotin, mutta uudessa ja vanhassa ICD-tiedostossa ei ollut tätä lukuun ottamatta suuria eroavaisuuksia. Esimerkiksi kaikkien CILO-solmujen InType oli sama, eikä erottimen poistaminen vaikuttanut olemassa olevia InType-tietoja. Oli tietojen puuttumisen syy mikä tahansa, datasettiin ei pitäisi jäädä tyhjiä FCDA-viittauksia.

SCL Manager mahdollistaa projektin SCD-tiedoston muokkaamisen suoraan käsin *Edit Master Project File* –näkyvästä, jonka kautta tyhjtä viittaukset olisi pystytty poistamaan

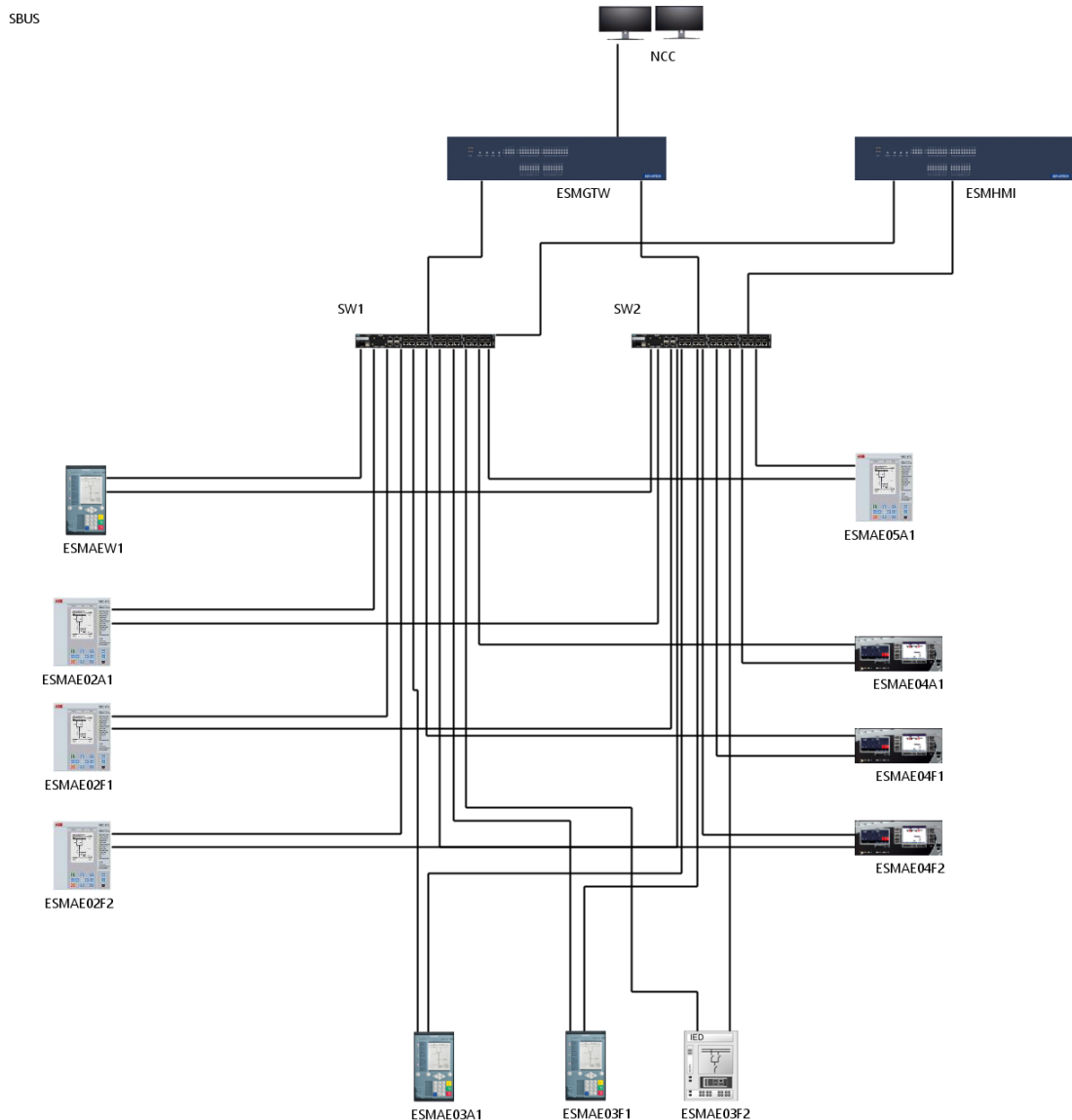
helposti. Tyhjästä viittauksista oli mahdollista päästä eroon myös poistamalla ja lisäämällä koko dataset uudestaan tai ottamalla kyseisen IED-laitteen IID-tiedosto ulos SCL Managerista ja muokkaamalla sitä käsin tekstieditorissa, jonka jälkeen IID-tiedosto vietiin takaisin SCL-manageriin. Vaikka vanha IED-laite korvattiin työtilassa korjatulla IID-tiedostolla Replace/Update –toiminnolla, osa kyseisestä IED:stä pääkaavioon tehdyistä viittauksista hävisi. Viittaukset vaikuttivat lisäksi häviävän täysin satunnaisesti. Tämä todettiin tekemällä linkitykset kenttäohjausyksiköstä pääkaavion kytkinlaitteisiin ja korvaamalla IED Replace/Update –toiminnon avulla samasta IID-tiedostosta luodulla laitteella useita kertoja. Joka kerta pääkaaviosta hävisi osa viittauksista hieman poikkeavalla tavalla.

Kun tiedosto oli lähestulkoon valmis, löytyi SCL Managerin validoinnissa vielä muutama ongelma. AE03A1-kenttäohjausyksikön tietomallissa oli kaksi samannimistä RxPos-dataobjektia samassa loogisessa solmussa, mutta tämä ei ollut nähtävissä tarkasteltaessa laitteen tietomallia SCL Managerin kautta. Ylimääräinen dataobjekti poistettiin muokkaamalla SCL Managerista saatua IID-tiedostoa tekstieditorilla ja viemällä se takaisin Replace/Update -toiminnon avulla. Jäi epäselväksi missä välissä ylimääräinen dataobjekti oli SCD-tiedostoon päätenyt, mutta laitteen ICD-tiedostosta sitä ei löytynyt. Lisäksi Siemensin kenttäohjausyksiköiltä RTU:lle ja HMI:lle raportteina vietävät signaalien värähtelynsuodatustiedot oli mallinnettu LPHD:lla, jonka instanssinumero oli 0. Kun raportit määritettiin RTU:lle ja HMI:lle, SCL Manager jätti näihin liittyvien ExtRef-elementtien LN instanssinumerot tyhjäksi. Tämä korjattiin ensin manuaalisesti viettäessä SCD ulos ohjelmistosta siitä löytyvällä *Export wizard* –toiminnolla. Export wizard mahdollistaa validointivirheiden korjaamisen joko käsin samaan tapaan kuin Edit Master Project File tai automaattisesti. Automaattinen korjaus kuitenkin poisti virheelliset rivit kokonaan, joten virheet oli korjattava käsin. Export wizardin kautta tehdyt muutokset eivät kuitenkaan tallentuneet projektitiedostoon, joten pysyvä korjaus tehtiin myöhemmin Edit Master Project File –näköisestä. Hieman vastaava validointivirhe aiheutui datasetistä, jossa viitattiin LLN0:aan:

```
<FCDA IdInst="LD0" InClass="LLN0" fc="ST" InInst="" doName="Mod" />
```

Muissa dataseiteissä, joissa esiintyi vastaavia viittauksia, InInst-attribuutti oli jätetty kokonaan pois ja luultavasti sitä ei olisi kuulunut olla tässäkään tapauksessa. Virhe korjaantui poistamalla dataobjekti datasetistä ja lisäämällä se uudestaan. Kun kaikki ongelmat oli ratkaistu, linkitettiin loput laitteiden tiedot pääkaavioon.

Helinksissä järjestelmäkonfiguraatio aloitetaan tuomalla tarvittavat ICD-tiedostot työtilaan, jonka jälkeen niiden avulla voidaan luoda tarvittavat IED-instanssit System Diagram –näkyssä. Näkyssä määritetään tarvittavat lähiverkot ja laitteiden verkkoasetukset. Laitteiden verkkoliitynnät määritetään piirtämällä väyläkaavio. Ohjelmassa luotu väyläkaavio on esitetty kuvassa 26. Laitteiden IP-osoitteet on muokattu kuvasta pois. SCL Managerissa väyläkaavio toimii ainoastaan dokumentaationa, eikä sen kautta ole mahdollista määrittää tietoliikenteeseen liittyviä asetteluita.



Kuva 26. Helinks STS:llä piirretty väyläkaavio.

Kun kaikki laitteet on määritetty väyläkaavioon, virtuaaliset IED:t voidaan korvata varsinaisilla IED-laitteilla. Tässä yhteydessä spesifikaatiossa määritetyt loogiset solmut linkitetään IED:n tietomallista löytyviin loogisiin solmuihin. Tämä on mahdollista tehdä joko automaattisesti tai manuaalisesti. Automaattinen linkitys ei välttämättä kuitenkaan toimi

kovin hyvin, jos spesifikaatiossa määritetty tietomalli poikkeaa käytetyn IED-laitteen tietomallista. Joidenkin laitteiden tietomalleissa osa signaaleista oli toteutettu samalla loogisella solmulla, kun taas spesifikaatiossa näille oli määritetty omat loogiset solmut. Saman loogisen solmun käyttäminen monta kertaa saman funktion sisällä ei kuitenkaan ollut loogisia solmuja linkitettäessä mahdollista. Kun tietomallin loogiset solmut on linkitetty spesifikaation loogisiin solmuihin ohjelma yrittää löytää niistä spesifioituja signaaleja vastaavat data-attribuutit automaattisesti. Jos vastaavuutta ei suoraan löydy täytyy signaalit liittää laitteen tietomalliin manuaalisesti. Mikäli konfiguraatio rakennettaisiin täysin ylhäältä alas siten, että IED-laitteiden tietomalli määritettäisiin spesifikaation pohjalta, määrittäykset voitaisiin tehdä täysin automaattisesti. Kaikki laitteet eivät kuitenkaan välttämättä mahdollista tietomallin muokkaamista näin vapaasti. Helinks mahdollistaa myös muiden kuin spesifikaatiossa määritettyjen loogisten solmujen liittämisen spesifikaatioon, mikä ei ole mahdollista SCL Managerissa. Tietomallien liittämisen jälkeen myös spesifikaatiossa määritellyt tiedot ovat edelleen nähtävissä, eivätkä ne pyyhkiydy pois samaan tapaan kuin SCL Managerissa. Spesifioitujen signaalien liittäminen manuaalisesti tietomalliin onnistui pääsääntöisesti hyvin. Prosessin aikana kuitenkin saattoi helposti poistaa spesifioituja signaaleja, jolloin ne oli luotava uudestaan.

Jos raportit ja GOOSE:t on määritelty jo spesifikaatiossa Helinks luo automaattisesti spesifikaation pohjalta tarvittavat datasetit, ohjauslohkot ja sisääntulot *Synchronize SCL Com* –toiminnolla. Kun toiminto ajettiin ensimmäisen kerran, saatiin virheilmoitus yhden kentän erottimen loogisten solmujen linkitysten puuttumisesta. Loogisten solmujen liittäminen kyseiseen erottimeen ei jostain syystä onnistunut ennen kuin koko erotin poistettiin pääkaaviosta ja lisättiin uudelleen. Tämän jälkeen erottimeen liittyvät raportit ja GOOSE-toiminnot oli määritettävä uudestaan. Poistetusta erottimesta jäi kuitenkin GOOSE-sovellukseen jälki, jonka poistaminen onnistui vasta muutaman yrityksen jälkeen.

Raporttien ohjauslohkojen indeksointitoiminto on mahdollista ottaa pois käytöstä, jolloin Helinks luo määritetystä ohjauslohkosta asiakkaiden määrästä riippuen yhden tai useamman kopion, jonka nimen perään lisätään indeksinnumero- tai kirjain. Indeksoinnin ollessa käytössä samasta ohjauslohkosta luodaan useita instansseja IED:ssä ja indeksinnumero lisätään ohjauslohkon nimeen vasta myöhemmin eikä se ole suoraan nähtävissä SCL:ssä. SCL:ssä esiintyvät ohjauslohkot tulisikin käsittää tyypeinä, joita käytetään ohjauslohkoinstanssien luomiseen [50, s. 98]. Toiminto on määritelty standardissa [50], mutta selkeämpi kuvaus löytyy lähteestä [91].

Dataseiteissa voidaan määrittää käytettäväksi, joko FCDA:ta tai FCD:tä. Käytettäessä FCD:tä datasettiin sisällytetään koko dataobjekti toiminnallisella rajoitteella suodatetuna, vaikka signaalit olisi valittu raporttiin data-attribuuttitasolla.

Helinksin sovelluskonfiguraatiossa ei ollut mahdollista määrittää ConfRev-attribuuttia raporttien ohjauslohkoille samaan tapaan kuin GOOSE-ohjauslohkoille, vaan Helinks määrittä ne automaattisesti. Tämän vuoksi raporttien ConfRev-attribuutit oli muokattava käsin Helinksin *Communication Editorissa*. Käsin asetellut arvot säilyivät konfiguraatiossa myös Synchronise SCL Comin ajamisen jälkeen, jos attribuutteja ei halunnut erikseen resetoida.

Siemensin IED-laitteille määritetyt ohjauslohkot ja datasetit oli automaattisesti luotu laitteiden tietomallissa ensimmäisenä olevan loogisen laitteen alle. Ne olisi ollut mahdollista siirtää toisen loogisen laitteen alle leikkaamalla ja liittämällä Communication Editorissa. Muokkausten jälkeen tilaajien asetellut olisi luotava uudestaan Synchronise SCL Com –toiminnolla, jotta uudet sijainnit päivittyisivät myös tilaajille [92]. Synchronise SCL Comin ajaminen manuaalisten muutosten jälkeen on kuitenkin ohjelmistotoimittajan mukaan riskialtista ja rikkoo kaikki linkit spesifikaatioon [93]. Helinksiltä on erikseen tilattavissa räätälöityjä *Utility Package* –lisäosia, joiden avulla voidaan tehdä tämän tyyppisiä muutoksia ohjelman toimintoihin [92] ja vastaavia ratkaisuja on toteutettu aikaisemminkin [93].

Kun projektin tietoliikennemääritykset on tehty, voidaan luoda SCD-tiedosto. Helinksissä on mahdollista luoda joko geneerinen SCD-tiedosto tai laitevalmistajakohtaiset tiedostot. Projektissa käytetyt ICD-tiedostot on mahdollista päivittää väyläkaaviossa *Reload ICD* –toiminnolla. Toimintoa kokeiltiin päivittämällä jälkikäteen Siemensin IED-laitteiden ICD-tiedostot. Päivitys ei vertailun perusteella aiheuttanut merkittäviä muutoksia SCD-tiedoston sisältöön.

5.5 SCD-tiedostojen vienti Siemensin ICT-työkaluun

Siemensin Siprotec 5 –sarjaan kuuluvien laitteiden konfigurointiin käytetään Digsii 5 –ohjelmistoa. Tyypillisesti IEC 61850 –kommunikointi määritellään näille laitteille Siemensin IEC 61850 System Configurator –ohjelmistolla. Digsii 5:n uudemmissa versioissa on kuitenkin mahdollista tuoda myös kolmannen osapuolen ohjelmistolla luotu SCD-tiedosto suoraan releprojektiin.

Sekä SCL Managerilla että Helinks STS:llä luotuja SCD-tiedostoja yritettiin tuoda Digsiiin. SCL Managerilla luotu SCD-tiedosto aiheutti joitain virheilmoituksia Digsii:ssä.

Tämän jälkeen SCL Managerilla luotiin SCD-tiedosto, josta oli poistettu osa laitteista tai niiden sisällöstä. Muokattu SCD-tiedosto tuotiin onnistuneesti Digsiin, mutta GOOSE-signaaleja ei kuitenkaan luotu releprojektiin. Helinksillä luotu SCD-tiedosto taas aiheutti ohjelmiston kaatumisen ilman selvää virheilmoitusta, josta olisi voinut saada selville mikä kaatumisen aiheutti. Tämän jälkeen Digsin päivitettiin versiosta 9.00 uusimpaan versioon 9.30 ja SCD-tiedostojen tuontia kokeiltiin uudestaan. Tällä kertaa sekä SCL Managerilla että Helinksillä luodut tiedostot aiheuttivat ohjelmiston kaatumisen ilman selkeää virheilmoitusta. Ohjelma vaikutti kaatuvan aina, kun yksittäistä GOOSE-sisääntuloa yritettiin luoda *Import specification* –näköymästä. Myös IEC 61850 System Configuratorilla luotu SCD-tiedosto johti Digsin kaatumiseen vastaavalla tavalla tuotuna. Ohjelmiston kaatuminen ei siis johtunut kolmannen osapuolen ohjelmistoilla luotujen SCD-tiedostojen sisällöstä vaan virheestä ICT-ohjelmistossa.

5.6 SCD-tiedostojen vienti GE:n ICT-työkaluun

GE:n IED-laitteisiin IEC 61850 –asetteluiden tuominen SCD-tiedostosta on mahdollista joko suoraan tai UR Setup –ohjelmiston Merge SCL –toiminnolla. Toiminnon avulla voidaan käytännössä yhdistää kahden eri SCL-tiedoston sisällöt, jotka tallennetaan kolmanteen tiedostoon. Uuteen tiedostoon valitaan IEC 61850 sekä verkkoasetukset SCD-tiedostosta ja IED-laitteen muut asetukset CID-tiedostosta. Kaikki GE:n IED-laitteiden asettelut on tallennettu CID-tiedostojen *private*-elementtien alle.

IEC 61850 –asetteluiden tuomista kokeiltiin ensin Helinksillä luodusta SCD-tiedostosta. Uudet SCL-tiedostot luotiin onnistuneesti, mutta jokaisen laitteen osalta saatiin jotain virheilmoituksia. Standardissa [50] on määritetty kaksi mahdollista tapaa määrittää IED-laitteen GOOSE- ja SV-sisääntulot. Sisääntulo voidaan luoda joko SCT:ssä siten, että siihen sidotaan ensin julkaistavan datasetin tiedot ja tilaajan sisäinen osoite määritetään jälkepäin ICT:ssä. Sisääntulo voidaan myös luoda ja sitoa sisäiseen osoitteeseen etukäteen ICT:ssä ja vasta tämän jälkeen sitoa se julkaistavan datasetin tietoihin SCT:ssä. [48] Näistä jälkimmäiseen viitataan standardissa ja eri ohjelmistoissa nimellä *later binding*. Helinks luo laitteille oletuksena uudet GOOSE-sisääntulot, mutta GE:n IED-laitteissa voidaan käyttää ainoastaan ennalta määrättyjä GOOSE-sisääntuloja. Kun GOOSE-signaalit sidottiin käsin olemassa oleviin sisääntuloihin Helinksin GOOSE-sovelluskonfiguraatiossa, virheilmoitukset poistuivat, mutta ulkoiset GOOSE-signaalit eivät kuitenkaan tulleet näkyviin UR Setupissa. Tämä mahdollisesti edellyttäisi sitä, että myös GOOSE-signaaleja lähettävät laitteet olisivat GE:n laitteita ja ne olisi myös tuotu UR Setupiin.

UR Setupilla konfiguroiduissa CID-tiedostoissa kaikki sisältö löytyy S1-liityntäpisteen alta, johon viitataan muista liityntäpisteistä *ServerAt*-elementillä. Jos SCD-tiedoston Communication-osiassa määritetty liityntäpiste on jokin muu kuin S1, tästä jää virheilmoitus UR Setupin tiedostojen yhdistämisen ja tuonnin yhteydessä luotuihin lokitiedostoihin. On kuitenkin epäselvää, onko tällä mitään käytännön merkitystä. Jos liityntäpisteeksi määrittää SCD-tiedostossa S1, vaikka haluttaisiin käyttää jotain toista liityntäpistettä, on verkkoasetukset määritettävä UR Setupissa käsin.

GE:n IED-laitteissa datasetit on jaettu nopeisiin ja hitaisiin standardissa määriteltyihin siirtoaikaluokkiin perustuen. Molempia datasettejä on käytössä rajattu määrä. Projektissa käytetyssä C60-kenttäohjausyksikössä on mahdollista luoda kuusi nopeaa ja 12 hidasta datasettiä. Nopeista dataseteistä korkeintaan neljää voidaan käyttää GOOSE-sovelluksissa ja kaikkia kuutta raporteissa. UR Setupin lokeihin jäi virheilmoitus nopeiden datasettien määrän ylittymisestä. Datasettejä ei kuitenkaan jaeta samalla tavalla nopeisiin ja hitaisiin SCT-ohjelmistoissa tai SCD-tiedostossa ja ainoat tähän viittaavat parametrit löytyvät ohjauslohkoista. UR Setup vaikuttaa määrittävän datasetit siten, että ensimmäiset kuusi SCD-tiedostossa esiintyvää datasettiä määritetään nopeiksi ja loput hitaiksi, samassa järjestyksessä kuin datasetit on ohjelmistossa listattu. Hitaiden datasettien sisältö jäi kuitenkin puuttumaan kokonaan. Ongelma korjaantui, kun datasetin nimen alkuun lisättiin teksti: TT3_, jolloin UR Setup osasi määrittää nämä datasetit hitaiksi ja dataseteistä löytyi SCD-tiedostossa määritelty sisältö. SCL Managerissa datasettien nimet oli muodostettu tällä tavoin jo aiemmin, joten samaa ongelmaa ei tullut enää eteen SCL Managerilla luodun SCD-tiedoston kanssa.

Helinks luo GE:n IED-laitteisiin ohjauslohkot UR Setupissa käytetyillä oletusnimillä, toisin kuin Hitachin ja Siemensin laitteisiin, jotka käyttävät spesifioituja nimiä. Tämä on mahdollisesti jokin laitevalmistajakohtainen yhteensopivuuteen liittyvä ominaisuus, joka ei ole välttämättä enää tarpeellinen käytössä olleessa UR Setupin versiossa. Jos yksittäisen raportin nimeä ja ID:tä, muokkasi käsin Helinksissä ja SCD-tiedoston vei UR Setupiin, ei tästä vaikuttanut aiheutuvan ongelmia.

Osassa raporttien ohjauslohkoissa käytetty indeksointitoiminto aiheutti virheilmoituksen UR Setupin lokitiedostoihin, koska käytetyissä laitteissa raporttien tilaajien maksimimäärä oli rajoitettu yhteen. Ongelma korjaantui, kun ohjauslohkojen *indexed*-attribuutti muutettiin epätodeksi ja ohjauslohkoista luotiin useampi kopio SCD-tiedostoon. Sama ongelma tuli eteen myös SCL Managerilla luodun SCD-tiedoston kanssa, koska tilaajien maksimimäärä oli asetettu ohjauslohkoille kahteen. Ongelman korjaamiseksi ohjauslohkoista olisi pitänyt luoda toiset kopiot käsin ja asettaa tilaajien määrä

molemmille kopioille yhteen. SCL Managerissa ohjauslohkojen *indexed*-attribuuttia ei ole mahdollista muuttaa ja standardin [50] mukaan SCT:ssä luoduissa ohjauslohkoissa attribuutin on oltava aina tosi.

Sekä SCL Managerilla että Helinksillä luotuja SCD-tiedostoja käytettäessä UR Setupin lokitiedostoihin tallentui virheilmoituksia, jotka liittyivät CID-tiedostojen *private dataan*. Import vaiheen lokeissa samoja virheilmoituksia ei kuitenkaan enää esiintynyt. Kun suojaroleisiin ohjelmoidut logiikat avattiin UR Setupissa, huomattiin että ne olivat käytännössä korvautuneet kenttäohjausyksiköiden logiikalla. Sama ongelma esiintyi riippumatta siitä, kummalla SCT-työkalulla SCD-tiedosto oli tehty ja tuotiinko tiedosto suoraan UR Setupiin vai käytettiin Merge SCL –toimintoa. Kun yhden releen sisältämää *private dataa* verrattiin tekstieditorilla alkuperäisen CID-tiedoston ja Helinksillä luodun SCD-tiedoston välillä, ei huomattu mitään eroavaisuuksia sisennyksiä lukuun ottamatta. Kun UR Setupin Merge SCL tai SCD Import –toimintojen kautta luotuja CID-tiedostojen *private dataa* verrattiin alkuperäiseen, löydettiin joitain poikkeavuuksia. Koska SCL:n *private-elementtien* alle määritelty tieto on standardoimatonta laitevalmistajakohdaista tietoa, on vaikea päätellä mitä poikkeavuudet tarkalleen ottaen merkitsevät. GE:n UR-sarjan laitteiden tietoliikenneoppaan [94] mukaan ensimmäinen *private-elementti* sisältää kaikki asetellut, joita ei ole voitu määrittää standardin mukaisissa elementeissä. Toinen *private-elementti* taas sisältää tarkistuskoodin, jonka avulla voidaan tarkistaa, onko tiedostoa muokattu muilla kuin GE:n työkaluilla. Oppaan mukaan IID-tiedostoissa voi olla myös kolmas *private-elementti*, joka sisältää laitteen tilaamiin GOOSE-sanomiin liittyviä tietoja, jotka on muissa tiedostoissa esitetty standardoidussa muodossa. Kuitenkin kaikissa tarkastelluissa tiedostoissa oli myös kolmas *private-elementti* ja IID-tiedostoja oli käytetty ainoastaan GE:n SCD-tiedoston rakentamiseen Helinksissä. Kolmannen *private-elementin* alle oli sisällöstä päätellen määritetty laitteiden näyttöpaneeliin liittyviä tietoja eikä näissä ollut poikkeavuuksia tiedostojen välillä. Poikkeavuuksia esiintyi ensimmäisen *private-elementin* tiedoissa, josta muutokset releiden logiikassa ovat todennäköisesti peräisin sekä toisen elementin tarkistuskoodissa. Koska alkuperäisen CID-tiedoston ja Helinksin SCD-tiedoston sisältämissä tiedoissa ei havaittu eroja, muutokset ovat luultavasti peräisin UR Setupista.

5.7 SCD-tiedostojen vienti Hitachin ICT-työkaluun

Hitachin PCM600-ohjelmistoon voidaan tuoda IEC 61850 –konfiguraatio joko SCL-tiedostosta tai se on mahdollista tehdä ohjelmiston sisällä. Jos esimerkiksi datasettejä halutaan muokata PCM:ssä on otettava *IEC 61850 configuration engineering* –tila käyttöön asetuksista, jonka ollessa päällä ohjelmaan ei voi tuoda ulkoisia SCL-tiedostoja.

SCL-tiedostojen tuontia varten on mahdollista ottaa käyttöön *Flexible Product Naming* -tila, jolloin releprojekti muunnetaan SCD-tiedoston tuonnin yhteydessä FPN-projektiksi. Tällöin releen tietomalli voidaan liittää spesifioituun tietomalliin hieman vastaavalla tavalla kuin Helinksin järjestelmäkonfiguraatiossa, mutta releen lähettämät tiedot vastaavat spesifioituja signaalinimiä. Koska SCD-tiedostot luotiin PCM:stä vietyjä ICD-tiedostoja käyttäen, ei FPN-toimintoa kokeiltu tässä työssä. Toimintoa on esitelty tarkemmin lähteessä [95].

PCM:ssä on mahdollista valita SCD-tiedoston tuonnin yhteydessä, tuodaanko tuntemattomat IED-tyypit generisinä IEC 61850 -objekteina vai jätetäänkö ne tuomatta. Samassa yhteydessä on mahdollista tehdä muutamia muita valintoja, kuten esimerkiksi huomioidaanko SCD-tiedoston substation-osiota. Helinksillä luotu SCD-tiedosto tuotiin PCM:ään ensin ilman PCM:lle tuntemattomia IED-tyyppejä. SCD-tiedoston tuominen ei aiheuttanut mitään virheilmoituksia, mutta PCM:n IEC 61850 konfiguraatio -näkyvää avattaessa saatiin varoituksia GE:n IED-laitteisiin liittyvistä GOOSE sisääntuloista, jotka viittasivat PCM:n mukaan olemattomiin dataobjekteihin. Jäi kuitenkin epäselväksi mistä varoitukset johtuivat. PCM:n IEC 61850 -näkyvästä löytyi kaikki IED-laitteille määritetyt GOOSE sisääntulot, datasetit ja ohjauslohkot. Kaikkia SCD:ssä määritettyjä laitteita, kuten varoituksissa esiintyneitä GE:n laitteita, ei kuitenkaan ollut merkitty vastaanottajiksi PCM:n GOOSE-matriisissa. Myös SCD:ssä määritetyt raportit ja niihin liittyvät datasetit olivat nähtävissä PCM:ssä, mutta SCD:ssä määritettyä RTU:ta ei ollut merkitty yhdenkään raportin vastaanottajaksi. Kaikki SCD-tiedostoon määritetyt GOOSE-sanomien ja raporttien vastaanottajat olivat silti näkyvissä PCM:n *Client editor* -näkyvässä. Jos PCM:lle tuntemattomat laitteet luotiin SCD:n tuonnin yhteydessä generisinä laitteina, myös niille kuuluvia datasettejä ja ohjauslohkoja pystyi tarkastelemaan PCM:ssä. Geneeristen laitteiden tuonnilla ei kuitenkaan ollut vaikutusta raporttien ja GOOSE-sanomien vastaanottajiin.

Kun GOOSE-sovellusta tarkasteltiin uudelleen Helinksissä havaittiin, että AE04-kentän kenttäohjausyksikköä ei ollut määritetty AE05-kentän Q2-erottimen tilatiedon tilaajaksi. PCM-konfiguraatiosta huomattiin, että Q2-erottimen tilatieto puuttui kokonaan datasetistä. Tämä johtui siitä, että Q2-erottimeen oli Helinksissä virheellisesti liitetty IED:n Q1-erottimeen liittyvät loogiset solmut. Kun tämä korjattiin ja Synchronise SCL Com -ajettiin uudelleen, näkyi AE04A1-IED AE05Q2-tilatiedon tilaajana.

Tuotaessa SCL Managerilla luotu SCD-tiedosto PCM:ään saatiin virheilmoitus, joka liittyi luultavasti siihen, että GE:n IED-laitteiden GOOSE-ohjauslohkot oli SCD:n tietoliikenneosiossa määritetty jokaiselle laitteen liityntäpisteelle. Myös tässä SCD:ssä

määritetyt datasetit ja ohjauslohkot olivat nähtävissä PCM:ssä, mutta vastaanottajiin liittyvät merkinnät puuttuivat lähestulkoon kokonaan. PCM antoi tähän liittyen varoituksia, joiden mukaan raporttien vastaanottajien loogisia solmuja ei ole olemassa.

5.8 Dokumentointiominaisuudet

Sekä Helinksillä että SCL Managerilla voidaan tulostaa projektiraportti, joka sisältää tietoja ohjelmistoilla luodusta IEC 61850 –konfiguraatiosta. Helinks tulostaa projektiraportin pdf-tiedostoon, jonka sisältöä on mahdollista muokata ohjelmassa. Projektiraportin ensimmäinen sivu on etulehti, johon voidaan lisätä yhteystietoja sekä käyttäjän valitsema kuva. Etulehden jälkeen tulee sisällysluettelo, joka sisältää yläotsikkotasolla seuraavat asiat:

1. SCD History
2. Introduction
3. General Specification
4. projects Single Line and Functions
5. System Communication Diagram

Raportin kolmannella sivulla on kuvaluettelo. Varsinainen sisältö alkaa ensimmäisestä luvusta, joka sisältää SCD-tiedoston historiaosiosta koostetun taulukon. Toinen luku on johdanto, johon on mahdollista lisätä esimerkiksi tiivistelmä. Kolmanteen lukuun voidaan kirjoittaa sanallisia kuvauksia projektin sovellusspesifikaatioista. Raporttiin tallentui ainoastaan GOOSE-sovellukselle määritetyt kuvaukset, mutta ei raportointisovellusten kuvauksia, vaikka niitä yritettiin ohjelmassa määrittää. Neljännessä luvussa on projektin pääkaavio, sovellukset ja kenttäkohtaiset toiminnot. GOOSE-sovellus oli kuvattu kaavioina ja taulukoina, mutta raportointisovellukset jäivät jälleen puuttumaan.

GOOSE-sovellus oli kuvattu taulukoissa ensin osana prosessia eli taulukoissa oli eritelty lähetettävät ja vastaanottavat funktiot ja tämän jälkeen erillisissä taulukoissa oli esitetty julkaisijat ja vastaanottajat. GOOSE-sanomissa käytettyjä tietoja ei määritetty Helinksiin signaaleina samaan tapaan kuin raporteissa käytettyjä tietoja, jolloin taulukossa signaalinimen paikalla oli tässä tapauksessa vain generinen *Pos.stVal[ST]* tai *Switch Position* –teksti. Raportissa käytetystä pääkaaviosta puuttui joitain viivoja, mutta se muistutti muuten ulkoasultaan Helinksistä kuvatiedostonakin tallennettavissa olevaa pääkaaviota. Neljännen luvun lopussa oli paikka myös muuntajiin liittyville toiminnoille, mutta koska projektissa ei ollut muuntajia, oli raportissa ainoastaan näihin liittyvät otsikot. Viides luku sisältää väyläkaavion sekä muita tietoliikennekonfiguraatioon liittyviä tietoja kuten aliverkot ja niihin kytkettyjen laitteiden IP-osoitteet. Viidennessä luvussa on lisäksi yhteenvetotaulukko IED-laitteista, jossa on listattu laitteiden valmistajat, tyytit

ja missä kentissä niitä on käytetty. Luvun lopussa on lueteltu ohjauslohkoihin liittyvät tiedot. Projektiraportti on mahdollista tulostaa myös ilman neljännen luvun kenttäkohtaisia signaalilistoja, jolloin raportin pituus voi lyhentyä huomattavasti. Projektiraportti voidaan tulostaa myös kustomoituna, mutta ominaisuus ei kuitenkaan toiminut testatuilla ohjelmistoversioilla käytetyssä testikokoonpanossa. Projektiraportti oli pääosin selkeästi muotoiltu, mutta joidenkin taulukoiden soluissa olleet tekstit jatkuivat toisten solujen päälle.

SCL Manager tulostaa projektiraportin RTF-tiedostona (*Rich Text Format*), joten sitä pystyy muokkaamaan suoraan tekstieditorilla. Myös SCL Managerin projektiraportti alkaa etulehdestä, jonka jälkeen tulee sisällysluettelo, jossa ei ole sivunumeroita. Sisällysluettelon rakenne on ylätasolla seuraava:

- Introduction
- Project Details
 - 1. Project: Esimerkkiasema
 - 1.1 Single Line Diagrams
 - 1.2 Network Diagrams
 - 1.3 Substation Details
 - 1.4 Logical Functions of Equipments
 - 1.5 Logical Functions Mapped to IEDs
 - 1.6 IED Details

Raportin varsinainen sisältö alkaa sisällysluettelon jälkeen geneerisestä johdantotekstistä. Raportin sisältö on jaettu yhden luvun alle, jossa se on jaettu alalukuihin ja alalukujen alalukuihin. Kaksi ensimmäistä alalukua *Single Line Diagrams* ja *Network Diagrams* sisältävät projektin pää- ja väyläkaaviot. Pääkaaviosta on mahdollista tallentaa tarkempi versio ohjelmasta suoraan kuvatiedostona, mutta väyläkaaviota ei pysty erikseen tallentamaan. Loput raportista koostuu erilaisista taulukoista. Ensimmäinen taulukko alaluvussa *Substation Details* listaa spesifioidun sähköaseman ensiökomponentit, toinen taulukko alaluvussa *Logical Functions of Equipments* listaa loogiset solmut ja kolmas alaluvussa *Logical Functions Mapped to IEDs* loogisten solmujen toteutuksen IED-laitteissa. Näiden jälkeen tulee *IED Details* –alaluku, joka kattaa valtaosan raportin sisällöstä. Alaluku on jaettu laitekohtaisiin alalukuihin, joka sisältää IED-laitteesta riippuen erilaisia tietoja laitteen konfiguraatiosta kuten esimerkiksi tietoliikenteen liityntäpisteiden parametrit, laitteen sisältämät loogiset solmut, GOOSE-ohjauslohkojen ja raporttien parametrit sekä laitteen lähettämien GOOSE-sanomien liitynnät muihin IED-laitteisiin. IED-laitteet on esitetty samassa järjestyksessä kuin SCL Managerin Project Explorer –näkyvässä.

Helinksillä on mahdollista luoda myös erilaisia Excel-taulukoita, kuten signaalilista. Helinksin luomassa signaalilistassa on esimerkiksi esitetty signaalille spesifioitu osoite sekä toteutunut osoite ja status, joka kertoo muun muassa onko signaali toteutettu kuten spesifioitu vai ei. Signaalilistan lisäksi Helinks pystyy luomaan ainakin luettelot raporttien sekä sähköasemaosion sisällöstä, projektin IED-laitteista ja GOOSE-matriisin. GOOSE-matriisissa on esitetty GOOSE-sanomien sisältö, sanomien julkaisijat ja vastaanottajat sekä niihin liitetyt LGOS LN:t. LGOS on looginen solmu, jota käytetään GOOSE-sanomien valvontaan vastaanottopäässä. Myös SCL Managerilla on mahdollista luoda vastaavanlainen GOOSE-matriisi CSV-tiedostona (*Comma-Separated Values*).

6. YHTEENVETO

Tässä luvussa tarkastellaan diplomityön tuloksia sille asetettujen käytännön tavoitteiden ja tutkimusongelmien kautta. Yhtenä käytännön tavoitteena oli arvioida standardin kuudennessa osassa esitetyn suunnitteluprosessin soveltuvuutta nykyaikaisiin laitteistoriippumattomiin sähköasematoteutuksiin. Suunnitteluprosessin hyödyntämisestä oli saatavilla hyvin vähän kirjoitettua tietoa ja sen soveltamisesta käytäntöön tietoa ei ollut saatavilla juuri yhtään. Kiinnostus aiheeseen on kuitenkin ehkä ollut jonkin verran kasvussa viime vuosina, mikä on näkynyt esimerkiksi alan konferensseissa pidettyjen esityksien aiheissa.

Suunnitteluprosessin soveltuvuuden arvioinnin kannalta oli keskeistä löytää käyttötarkoitukseen soveltuva työkalu ja ensimmäinen tutkimuksen pääongelmista oli: *mitä ovat uuden sukupolven työkalut ja työtavat IEC 61850 –sähköasematoimitusten suunnittelussa?* Sekä SST:n että SCT:n toiminnallisuudet sisältäviä ohjelmistoja löydettiin yhteensä seitsemän kappaletta. Löydetyistä ohjelmistoista neljää verrattiin ohjelmistomittajilta saatujen tietojen pohjalta ja näistä kaksi valittiin yksityiskohtaisempaan tarkasteluun. Tarkasteluun valikoituneet ohjelmistot: Kalkitech ASE SCL Manager ja Helinks STS edustivat hieman erilaisia lähestymistapoja IEC 61850 –järjestelmän konfiguroinnissa. SCL Managerissa konfigurointi on huomattavasti manuaalisempaa, mikä mahdollistaa toisaalta suuremman valinnan vapauden konfiguraation luonnissa, mutta samalla hidastaa työskentelyä. Helinks STS:ssä monet toiminnot oli automatisoitu mikä tekee työskentelystä tehokkaampaa, mutta toisaalta joitain määrittelyitä voi olla hankalampi tehdä. Tästä esimerkkinä datasettien ja ohjauslohkojen automaattinen sijoittelu loogisiin laitteisiin, josta poikkeaminen vaatii Helinksin tapauksessa hieman vaivannäköä tai ohjelmiston räätälöintiä. Kuitenkin työnkulun automatisointi tekee Helinksistä järjestelmäintegraattorin näkökulmasta huomattavasti houkuttelevamman vaihtoehdon. Ohjelmaan voidaan luoda tyyppikirjasto, johon voidaan tallentaa esimerkiksi laitevalmistaja- ja asiakaskohtaisia tyyppikenttiä, joiden avulla IEC 61850 –konfiguraatio olisi mahdollista rakentaa hyvin tehokkaasti. Pitkälle vietynä prosessin lopputuloksena voitaisiin saada SCD-tiedoston lisäksi jopa valmis signaalilista IEC 60870-5-104 –osoitteinen.

Toinen merkittävä ero työkalujen välillä oli niiden järjestelmän spesifiointiin liittyvissä ominaisuuksissa. SCL Manager toteuttaa näiltä osin ainoastaan standardissa määritellyt ominaisuudet ja spesifikaatio sisältää ainoastaan pääkaavion sekä siihen liittyvät

loogiset solmut. Sen sijaan Helinks tukee myös virtuaalisia IED-laitteita, joka mahdollistaa sekä laitteiden että myös erilaisten sovellusten spesifioinnin. Sovellusspesifikaatioita ei kuitenkaan pysty tallentamaan SCL-muotoon, vaan ne ovat olemassa ainoastaan Helinksin sisällä. Koska näitä ominaisuuksia ei ole vielä standardoitu, tietojen vaihtaminen eri valmistajien työkalujen välillä voi olla hankalaa.

Molemmissa ohjelmistoissa esiintyi jonkin verran käyttöä haittaavia ohjelmistovirheitä. Virheistä raportoiminen tapahtui Helinksin tapauksessa verkkosivuilta löytyvän lomakkeen kautta ja SCL Managerin tapauksessa sähköpostitse. Kaikkia työn aikana kohdattuja ongelmia ei kuitenkaan raportoitu eteenpäin. Molemmat ohjelmistotoimittajat vastasivat vikailmoitukseen nopeasti.

Ensimmäinen tutkimusongelma jaettiin kahteen alaongelmaan: *millaisia rajoitteita erilaiset projektit ja asiakkaat asettavat työkaluille ja suunnittelumenetelmille ja millaisia rajoitteita käytetyt laitteet asettavat työkaluille ja suunnittelumenetelmille*. Työssä ei juurikaan selvitetty projektien ja asiakkaiden työkaluille ja suunnittelumenetelmille asettamia rajoitteita. On yleisesti tiedossa, että verkkoyhtiöt – joitain poikkeuksia lukuun ottamatta – eivät juurikaan rajoita konfigurointi- ja suunnittelutyökalujen käyttöä, eikä IEC 61850 –toteutuksia ole spesifioitu kovin yksityiskohtaisella tasolla. Suurimmat rajoitteet kolmannen osapuolten työkalujen ja uusien suunnittelumenetelmien käytöstä tulevatkin laitevalmistajien ohjelmistoista. Tässä työssä käytetyistä IED-konfigurointiohjelmistoista ainoastaan yhteen pystyttiin tuomaan kolmannen osapuolen suunnittelutyökalulla luotu SCD-tiedosto ilman merkittäviä ongelmia. IEC 61850 –konfiguraatiota luodessa on tunnettava myös laitekohtaiset rajoitteet ja ominaisuudet, jollaisia tuli esille ainakin GE:n IED-laitteiden tapauksessa. Tuloksia ei voida pitää täysin kattavana, koska SCD vietiin ainoastaan IED-konfigurointiohjelmiin eikä konfiguraatioita ladattu laitteisiin. Menetelmän ja työkalujen hyödyntämisen tarkempi arviointi edellyttäisi myös testijärjestelmän rakentamista. IED-konfigurointiohjelmissä esiintyneiden ongelmien takia tällaisen kokeen toteuttaminen olisi tällä hetkellä järkevää ainoastaan Hitachin laitteilla. GE:n UR Setupista ei käytetty työssä viimeisintä versiota, joten on mahdollista, että käytetyssä versiossa esiintyneet ongelmat on korjattu. Siemensin IED-konfigurointiohjelmistossa havaitusta viasta ilmoitettiin Siemensin tukeen, eikä siihen ole kirjoitushetkellä olemassa korjausta. On kuitenkin odotettavissa, että vika korjataan lähitulevaisuudessa.

Toinen tutkimusongelma *millä tavoin uuden sukupolven työkalut ja työtavat ovat hyödynnettävissä laitteistoriippumattomassa IEC 61850 –toteutuksessa* jaettiin kolmeen alaongelmaan:

- a. Miten uudet suunnittelumenetelmät ja työkalut ovat hyödynnettävissä nykyaikaisissa toteutuksissa?
- b. Miten uusia suunnittelumenetelmiä ja -työkaluja voidaan hyödyntää tulevaisuudessa?
- c. Mitä ovat uusien menetelmien edut ja haitat vanhoihin nähden?

Uusien suunnittelumenetelmien ja työkalujen hyödynnettävyyttä nykyaikaisissa toteutuksissa rajoittaa pitkälti edellä kuvatut ongelmat laitevalmistajien IED-konfigurointiohjelmistoissa. Jos kolmannen osapuolen työkaluja voidaan hyödyntää ainoastaan yhden laitevalmistajan laitteilla rakennetussa järjestelmässä, ei päästä eroon laitevalmistaja-kohtaisista työkaluista. Joissain tapauksissa voi olla tarpeen rakentaa SCD-tiedosto ainoastaan IEC 61850 –järjestelmän toimintaa kuvaavaksi dokumentiksi. Tällaisessa tapauksissa olisi suotavaa, että käytetty työkalu on mahdollisimman automatisoitu, jolloin Helinks STS voisi olla sopivin vaihtoehto. Helinksin Communication Editor voisi olla myös hyödyllinen olemassa olevien SCD-tiedostojen sisällön tarkastelussa. Kummankaan tarkastellun työkalun kyky tuottaa erillistä dokumentaatiota SCL-tiedostojen sisällöstä ei ehkä silti ole sellainen, että se itsessään oikeuttaisi ohjelmiston hankintaa. Standardissa määritetyn suunnitteluprosessin spesifointivaihe on lähtökohtaisesti suunnattu verkonhaltijoiden työkaluksi ja sen tarkastelu jäi vähemmälle huomiolle. Nykyisellään standardissa määritetyn SSD-tiedoston sisältö on sen verran suppea, että siitä ei ehkä ole merkittävää hyötyä kenellekään. Standardiin tulossa olevat laajennukset saattavat kuitenkin muuttaa tilannetta huomattavasti. SCL-muotoisten spesifikaatioiden määrittäminen ja hyödyntäminen vaatisi myös verkkoyhtiöiltä panostusta IEC 61850 –järjestelmäosaamiseen. Uusi suunnitteluprosessi voisi mahdollistaa paremman hallinnan järjestelmien toteutuksesta. SCL-muotoiset spesifikaatiot vähentäisivät myös järjestelmäintegraattorin työtä ja niillä voisi olla lyhentävä vaikutus sähköasema-automaatioprojektien läpimenoaikoihin. Ei ole kuitenkaan tietoa missä vaiheessa standardin kehitys näiltä osin on ja voi mennä vielä vuosia ennen kuin uudet ominaisuudet ovat kunnolla hyödynnettävissä.

Kohdeyhtiön ja järjestelmäintegraattorin näkökulmasta realistinen lähitulevaisuuden tavoite voisi olla pyrkiä hyödyntämään kolmannen osapuolen työkaluja IEC 61850 –järjestelmien konfiguroinnissa. Suurimmat hyödyt saadaan konfigurointityön nopeutumisesta, jonka kannalta keskeistä on oikean työkalun valinta ja tähän tarkoitukseen soveltuvin vaihtoehto on vertailun perusteella Helinks STS. Lisäksi SCD-tiedoston rakentaminen heti konfigurointitöiden alussa mahdollistaa RTU- ja HMI-konfiguraatiotöiden sekä IED-konfiguraatiotöiden välisen riippuvuuden poistamisen, joka mahdollistaa töiden vapaamman aikataulutuksen. Myöhemmässä vaiheessa toteutusten yhtenäistyminen helpottaa järjestelmien käyttöönottoa ja ylläpitoa. IEC 61850 –toteutusten

spesifiointi voisi olla järkevää ottaa myös osaksi olemassa olevia sähköasematoimitusten kehityshankkeita. Kolmannen osapuolen työkalujen hyödyntäminen tulee kuitenkin rajoittamaan toisiolaitevalintoja ja työkaluja olisi tarpeellista vielä pilotoida oikeassa sähköasematoteutuksessa tai testijärjestelmässä. IEC 61850 –toteutusten yhtenäistäminen oli myös yksi työn käytännön tavoitteista, johon ei tämän työn puitteissa päästy. Diplomityön aikana saatiin silti hyvä näkemys saatavilla olevista kolmannen osapuolen konfigurointityökaluista sekä niiden soveltamismahdollisuuksista, joka antaa hyvät lähtökohdat toteutusten yhtenäistämiseen tulevaisuudessa.

LÄHTEET

- [1] Working Group B5.12, Engineering Guidelines for IEC 61850 Based Digital SAS, CIGRE, 2011, 55 p.
- [2] IEC/TR 61850-1, Communication networks and systems for power utility automation – Part 1: Introduction and overview, Edition 2.0, IEC, Geneve, Sveitsi, 2013. 73 p.
- [3] Network Protection & Automation Guide, Alstom Grid, 2011, 508 p.
- [4] R. E. Mackiewicz, Overview of IEC 61850 and Benefits, 2006 IEEE PES Power Systems Conference and Exposition, Atlanta, Georgia, Yhdysvallat, 2006, pp. 623–630.
- [5] Working Group B5.50, IEC 61850 based substation automation systems – Users expectations and stakeholders interactions, CIGRE, 2020, 73 p.
- [6] Working Group B5.36, Applications of IEC 61850 Standard to Protection Schemes, CIGRE, 2013, 209 p.
- [7] E. Carvalheira, A. Klien, How the Engineering Design Process Can Simplify the Testing of Automation and Control Systems, 2019 72nd Conference for Protective Relay Engineers (CPRE), College Station, Texas, Yhdysvallat, 2019, pp. 1–7.
- [8] A. Keskinen, Sähköasemastandardin IEC 61850 soveltaminen toimitusprojektissa, diplomityö, Tampereen teknillinen yliopisto, 2008, 53 s.
- [9] S. Turunen, Protection of Microgrids and Distributed Energy Resources based on IEC 61850, diplomityö, Tampereen teknillinen yliopisto, 2016, 99 p.
- [10] J. Hurta, Digitaalisen sähköaseman testaus- ja valvontaratkaisut, Diplomityö, Aalto-yliopisto, Sähkötekniikan korkeakoulu, Espoo, 2020, 91 s. + liitt. 8 s.
- [11] B. Bauernschmitt, M. Kaliwoda, B. Keune, D. Hilbrich, C. Rehtanz, An integral engineering process for centralised protection & control systems according to IEC 61850-6, 2015 Modern Electric Power Systems (MEPS), Wroclaw, Puola, 2015.
- [12] M. A. Aftab, S. M. S. Hussain, I. Ali, T. S. Ustun, A Novel SCL Configuration Method for Modeling Microgrids With IEC 61850, IEEE Systems Journal, Vol 14, No. 2, 2020, pp. 2676–2683.
- [13] C. Brunner, T. Sterckx, IEC61850 ENTSO-E Profile introduction and Engineering Process Refinement, OSMOSE, 2020, 24 p. + liitt. 6 p. Saatavissa (viitattu 12.12.2021): <https://www.osmose-h2020.eu/download/d7-1-iec61850-entso-e-profile-introduction-and-engineering-process-refinement/>
- [14] L. Pohjoisvirta, Choosing a tool for improved software test management, diplomityö, Tampereen teknillinen yliopisto, 2013, 113 p. + liitt. 9 p.

- [15] P. Koski, Software calculation tools in electric power system substation design, diplomityö, Teknillinen korkeakoulu, 2007, 83 p.
- [16] Working Group B3.01, The Impact of New Functionalities on Substation Design, CIGRE, 2009, 180 p.
- [17] Working Group B5.11, The introduction of IEC 61850 and its impact on protection and automation within substations, CIGRE, 2007, 102 p.
- [18] IEC 61850-5, Communication networks and systems for power utility automation – Part 5: Communication requirements for functions and device models, Edition 2.0, IEC, Geneve, Sveitsi, 2013, 306 p.
- [19] J. Elovaara, L. Haarla, Sähköverkot II, Verkon suunnittelu, järjestelmät ja laitteet, Otatieto, Turenki, 2011, 551 s.
- [20] SFS-IEC 60050-448, Sähkötekniillinen sanasto, Osa 448: Sähkövoimajärjestelmän relesuojaus, Suomen standardoimisliitto SFS, 2002, 82 s.
- [21] J. Mörsky, Relesuojaustekniikka, 2. painos, Otatieto, Hämeenlinna, 1993, 459 s.
- [22] S. H. Horowitz, A. G. Phadke, Power System Relaying, 4. painos, John Wiley and Sons Ltd, 2014, 381 p.
- [23] P. M. Anderson, Power System Protection, IEEE Press, Piscataway, New Jersey, Yhdysvallat, 1999, 1307 p.
- [24] Working Group I-01, Understanding microprocessor-based technology applied to relaying, IEEE, 86 p. Saatavissa (viitattu 18.4.2021): <https://www.pes-psrc.org/kb/published/reports/UNTAR-Ed2.pdf>
- [25] T. Krieg, J. Finn, Substations, Springer International Publishing AG, 2019, 1079 p.
- [26] J. D. McDonald, Electric Power Substations Engineering, 3. painos, CRC Press, Boca Raton, Florida, Yhdysvallat, 2012, 524 p.
- [27] T. Cegrell, Power system control technology, Prentice-Hall International, Hemel Hempstead, Hertfordshire, Yhdistynyt kuningaskunta, 1986, 342 p.
- [28] WG H9, Communications Technology for Protection Systems, IEEE, 2019, 229 p. Saatavissa (viitattu 9.8.2021): <https://www.pes-psrc.org/kb/published/reports/IEEE%20PSRC%20Subcommittee%20H9%20Understanding%20Comm%20Tech%20for%20Protection%20-20130113%20D8F.pdf>
- [29] B. M. Wilamowski, J. D. Irwin, Industrial Communication Systems, 2. painos, CRC Press, Boca Raton, Florida, Yhdysvallat, 2011, 944 p.
- [30] ABB:n TTT-käsikirja 2000-07, Luku 15: Sähköjakeluverkon automaatio, ABB, 2000. 16 s. Saatavissa (viitattu 2.10.2021): http://www.oamk.fi/~kurki/automaatiolabrat/TTT/15_S%84hk%94njakeluverkon%20automaatio.pdf
- [31] S. Repo, Distribution automation, Network monitoring, luentokalvot, Tampereen teknillinen yliopisto, Tampere, 2019.

- [32] B. Nikolić, S. Khan, N. Gabdullin, Development of non-conventional instrument transformers (NCIT) using smart materials, *Journal of Physics: Conference Series* 772, 2016.
- [33] J. Močnik, J. Humar, A. Žemva, A non-conventional instrument transformer, *Measurement*, Vol. 46, Iss. 10, 2013, pp. 4114–4120.
- [34] R. Thomas, A. Vujanic, D. Z. Xu, J. - E. Sjödin, H. R. M. Salazar, M. Yang, N. Powers, Non-conventional instrument transformers enabling digital substations for future grid, 2016 IEEE/PES Transmission and Distribution Conference and Exposition (T&D), Dallas, Texas, Yhdysvallat, 2016.
- [35] A. Mingotti, F. Costa, L. Peretto, R. Tinarelli, Characterization Procedure for Stand-Alone Merging Units Based on Hardware-in-the-Loop Technology. *Energies*, Vol. 14, Iss. 7. 2021.
- [36] A. Apostolov, B. Vandiver, IEC 61850 process bus - principles, applications and benefits, 2010 63rd Annual Conference for Protective Relay Engineers, College Station, Texas, Yhdysvallat, 2010.
- [37] H. Vardhan, R. Ramlachan, W. Szela, E. Gdowik, Deploying digital substations: Experience with a digital substation pilot in North America, 2018 71st Annual Conference for Protective Relay Engineers (CPRE), College Station, Texas, Yhdysvallat, 2018.
- [38] Future Intelligent Transmission Network Substation (FITNESS) Project Progress Report, SP Energy Networks, 2018, 21 p. Saatavissa (viitattu 14.2.2021): https://www.spenergynetworks.co.uk/userfiles/file/FITNESS_PPR_081618.pdf
- [39] Fingrid toteuttaa digitaalisen sähköaseman pilotin, Fingrid Oyj, verkkosivu, 2019, Saatavissa (viitattu 24.7.2021): <https://www.fingrid.fi/ajankohtaista/tiedotteet/2019/fingrid-toteuttaa-digitaalisen-sahkoaseman-pilotin/>
- [40] Working Group K15, Centralized Substation Protection and Control, IEEE, 2015, 76 p. Saatavissa (viitattu 25.4.2021): https://www.pes-psrc.org/kb/published/reports/IEEE_PES_PSRC_WG%20K15_Report_CPC_Dec_2015.pdf
- [41] Z. Bo, L. Wang, F. Zhou, K. Luo, M. Han, W. Yin, J. Liu, Substation cloud computing for secondary auxiliary equipment, 2014 International Conference on Power System Technology, Chengdu, Kiina, 2014, pp. 1853-1858
- [42] D. Proudfoot, UCA and 61850 for dummies, 2002. Saatavissa (viitattu 25.9.2021): <https://www.nettedautomation.com/download/UCA%20and%2061850%20for%20dummies%20V12.pdf>
- [43] IEEE-SA TR 1550, IEEE-SA Technical Report on Utility Communications Architecture (UCA™) Version 2.0, The Institute of Electrical and Electronics Engineers Inc., New York, New York, Yhdysvallat, 1999, 418 p. Saatavissa (viitattu 27.9.2021): <https://www.epri.com/research/products/TP-114398>
- [44] Working Group H6, Application Considerations of IEC 61850/UCA 2 for Substation Ethernet Local Area Network Communication for Protection and Control, IEEE, 2005, 89 p. Saatavissa (viitattu 28.10.2021): https://www.pes-psrc.org/kb/published/reports/H6Paper-App%20Consider%20of%20IEC61850&UCA_072205_083105.pdf

- [45] M. Mekkanen, On reliability and performance analyses of IEC 61850 for digital SAS, väitöskirja, Vaasan yliopisto, Acta Wasaensia 336, 2015, 202 p.
- [46] C. R. Ozansoy, A. Zayegh, A. Kalam, Object Modeling of Data and DataSets in the International Standard IEC 61850, IEEE Transactions on Power Delivery, vol. 24, no. 3, 2009, pp. 1140-1147.
- [47] Communication networks and systems for power utility automation – ALL PARTS, IEC, verkkosivu. Saatavissa (viitattu 20.8.2021): <https://web-store.iec.ch/publication/6028#additionalinfo>
- [48] IEC 61850-7-1, Communication networks and systems for power utility automation – Part 7-1: Basic communication structure – Principles and models, Edition 2.1, IEC, Geneve, Sveitsi, 2020, 195 p.
- [49] IEC 61850-4, Communication networks and systems for power utility automation – Part 4: System and project management, Edition 2.1, IEC, Geneve, Sveitsi, 2020, 47 p.
- [50] IEC 61850-6, Communication networks and systems for power utility automation – Part 6: Configuration description language for communication in power utility automation systems related to IEDs, Edition 2.1, IEC, Geneve, Sveitsi, 2018, 269 p.
- [51] IEC 61850-8-1, Communication networks and systems for power utility automation – Part 8-1: Specific communication service mapping (SCSM) – Mappings to MMS (ISO 9506-1 and ISO 9506-2) and to ISO/IEC 8802-3, Edition 2.0, IEC, Geneve, Sveitsi, 2011, 386 p.
- [52] IEC 61850-9-2, Communication networks and systems for power utility automation – Part 9-2: Specific communication service mapping (SCSM) – Sampled values over ISO/IEC 8802-3, Edition 2.1, IEC, Geneve, Sveitsi, 2020, 34 p.
- [53] Y. M. Bourien, C. Brunner, T. Sterckx, C. Bloch, IEC 61850 standard: What for, which benefits, what pending challenges? How is the Osmose project contributing, OSMOSE, 2021, Saatavissa (viitattu 10.6.2022): https://www.osmose-h2020.eu/wp-content/uploads/2021/05/Webinar-IEC61850-full-presentation_FINAL-1.pdf
- [54] IEC 61850-7-4, Communication networks and systems for power utility automation – Part 7-4: Basic communication structure – Compatible logical node classes and data object classes, Edition 2.0, IEC, Geneve, Sveitsi, 2010, 179 p.
- [55] T. Kostic, O. Preiss, C. Frei, Understanding and using the IEC 61850: a case for meta-modelling, Computer Standards & Interfaces, Vol. 27, Iss. 6, 2005, pp. 679-695.
- [56] IEC 61850-7-2, Communication networks and systems for power utility automation – Part 7-2: Basic information and communication structure – Abstract communication service interface (ACSI), Edition 2.0, IEC, Geneve, Sveitsi, 2010, 213 p.
- [57] M. Rintala, J. Jokinen, Olioiden ohjelmointi C++:lla, 2013, 466 s.
- [58] B. Dathan, S. Ramnath, Object-Oriented Analysis and Design, Springer, 2011, 470 p.

- [59] W. Vogels, World Wide Failures, Proceedings of the 7th workshop on ACM SIGOPS European workshop: Systems support for worldwide applications, Con-nemara, Irlanti, 1996, pp. 115–120.
- [60] IEC TR 61850-90-4, Communication networks and systems for power utility au-tomation – Part 90-4: Network engineering guidelines, Edition 2.0, IEC, Geneve, Sveitsi, 2020, 358 p.
- [61] IEC/IEEE 61850-9-3, Communication networks and systems for power utility au-tomation – Part 9-3: Precision time protocol profile for power utility automation, Edition 1.0, IEC, Geneve, Sveitsi, 2016, 14 p.
- [62] T. S. Ustun, S. M. Farooq, S. M. S. Hussain, Implementing Secure Routable GOOSE and SV Messages Based on IEC 61850-90-5, IEEE Access, Vol. 8, 2020, pp. 26162-26171.
- [63] IEEE Std 802.1Q™-2018, IEEE Standard for Local and Metropolitan Area Net-works – Bridges and Bridged Networks, IEEE, New York, New York, Yhdys-vallat, 2018, 1991 p.
- [64] Overview and Introduction to the Manufacturing Message Specification (MMS), Revision 2, Sterling Heights, Michigan, Yhdysvallat, SISCO Inc., 1995, 47 p. Saatavissa (viitattu 26.10.2021): [https://www.sisconet.com/wp-content/up-loads/2016/03/mmsovrlg.pdf](https://www.sisconet.com/wp-content/uploads/2016/03/mmsovrlg.pdf)
- [65] What is Helinks STS?, Helinks LLC, verkkosivu, Saatavissa (viitattu 5.11.2021): <https://www.helinks.com/products-services/>
- [66] IEC 61850 Substation Configuration Tool, Kalki Communication Technologies Private Limited, verkkosivu, Saatavissa (viitattu 5.11.2021): <https://www.ase-systems.com/ase61850-configuration-software/>
- [67] SCL Matrix, Grid Software Inc., verkkosivu, Saatavissa (viitattu 5.11.2021): <https://www.gridsoftware.com/products/sclmatrix.html>
- [68] IET600 - Integrated engineering tool, Hitachi Energy Ltd., verkkosivu, Saata- vissa (viitattu 5.11.2021): <https://www.hitachienergy.com/offering/product-and-system/substation-automation--protection---control/products/tools/iet600>
- [69] Engineering software for IEC 61850 systems, Siemens, verkkosivu, Saatavissa (viitattu 5.11.2021): <https://new.siemens.com/global/en/products/energy/energy-automation-and-smart-grid/protection-relays-and-control/engineering-tools-for-protection/engineering-software-iec-61850-system-configurator.html>
- [70] EcoStruxure™ Power Automation System Engineering – EPAS-E, Schneider Electric, verkkosivu, Saatavissa (viitattu 5.11.2021): <https://www.se.com/ww/en/product-range/62039-ecostruxure™-power-automata-tion-system-engineering---epas-e/?parent-subcategory-id=5440&filter=busi-ness-6-medium-voltage-distribution-and-grid-automation#overview>
- [71] PCM600 - Protection and control IED manager, Hitachi Energy Ltd., verkkosivu, Saatavissa (viitattu 5.11.2021): <https://www.hitachienergy.com/offering/product-and-system/substation-automation-protection-and-control/prod-ucts/tools/pcm600>

- [72] DIGSI 5 - Engineering software, Siemens, verkkosivu, Saatavissa (viitattu 5.11.2021): <https://new.siemens.com/global/en/products/energy/energy-automation-and-smart-grid/protection-relays-and-control/engineering-tools-for-protection/engineering-software-digsi-5.html>
- [73] J. Reuter, TR IEC 61850 90-30 – Guidelines for IEC 61850 Function Modeling in SCL, PAC World, Iss. 061, 2022. Saatavissa (viitattu 5.2.2023): <https://www.pacw.org/tr-iec-61850-90-30-guidelines-for-iec-61850-function-modeling-in-scl>
- [74] J. Reuter, Understanding the Engineering Process with Enhanced Specification, IEC 61850 Week 2021.
- [75] S. T. Jose, IEC 61850 SCL – Top Down / Bottom-Up Engineering and Ed.1.0 to Ed.2.1 changes, Saatavissa (viitattu 8.1.2022): <http://iec61850.ucaiug.org/2019IOP-Charlotte/IOP%20Test%20Documents/Boot%20Camp%20Presentations/SCL%20Engineering%20-%20BottomUp-ASE-Kalkitech.pdf>
- [76] J. Reuter, IEC 61850 top-down design, IEC 61850 Global 2019. Saatavissa (viitattu 8.1.2022): <https://www.helinks.com/wp-content/uploads/2019/10/Top-Down-Design-JReuter-Helinks-IEC61850Global.pdf>
- [77] M. Haecker, C. Bennauer, Unified IED modelling using IEC 61850 Flexible Product Naming, PAC World, Iss. 051, 2020. Saatavissa (viitattu 6.11.2021): <https://www.pacw.org/unified-ied-modelling-using-iec-61850-flexible-product-naming>
- [78] B. Jang, A. Abubakari, N. Kim, IEC 61850 SCL Validation Using UML Model in Modern Digital Substation, Smart grid and renewable energy, Vol. 9, 2018, pp. 127 – 149.
- [79] B. T. Jang, A. Alidu, N. D. Kim, Design of an Algorithm for the Validation of SCL in Digital Substations, KEPCO Journal on Electric Power and Energy, Vol. 3, No. 2, 2017, pp. 89–97.
- [80] IEC 61850 Interoperability Testing, Oct 2019, Charlotte, NC USA, Final Test Report, UCAIug, 318 p. Saatavissa (viitattu 15.2.2022): http://www.ucaiug.org/IOP_Registration/IOP%20Reports/IEC%2061850%202019%20IOP%20Final%20Report%2020200122.pdf
- [81] A. S. Jadhav, R. M. Sonar, Evaluating and selecting software packages: A review, Information and Software Technology, Vol. 51, Iss. 3, 2009, pp. 555–563.
- [82] B. A. Kitchenham, L. Jones, Evaluating software engineering methods and tool part 5: the influence of human factors, Software Engineering Notes, Vol. 22, No. 1, 1997, pp. 13–15.
- [83] B. A. Kitchenham, L. Jones, Evaluating software engineering methods and tool part 6: identifying and scoring features, Software Engineering Notes, Vol. 22, No. 2, 1997, pp. 16–18.
- [84] B. A. Kitchenham, L. Jones, Evaluating software engineering methods and tool part 7: planning feature analysis evaluation, Software Engineering Notes, Vol. 22, No. 4, 1997, pp. 21–24.

- [85] B. A. Kitchenham, L. Jones, Evaluating SW Eng. methods and tools part 8: analysing a feature analysis evaluation, *Software Engineering Notes*, Vol. 22, No. 5, 1997, pp. 10–12.
- [86] M. Pérez, T. Rojas, Evaluation of Workflow-type software products: a case study, *Information and Software Technology*, Vol. 42, Iss. 7, 2000, pp. 489–503.
- [87] C. E. Bozdağ, C. Kahraman, D. Ruan, Fuzzy group decision making for selection among computer integrated manufacturing systems, *Computers in Industry*, Vol. 51, Iss. 1, 2003, pp. 13–29.
- [88] F. Karayazi, I. Bereketli, Criteria Weighting for Blockchain Software Selection Using Fuzzy AHP, Teoksessa: C. Kahraman, S. C. Onar, B. Oztaysi, I. U. Sari, S. Cebi, A. C. Tolga (toim.), *Intelligent and Fuzzy Techniques: Smart and Innovative Solutions, INFUS 2020, Advances in Intelligent Systems and Computing*, Vol. 1197, Springer, Cham, Sveitsi, 2021, pp. 608–615.
- [89] M. Brunelli, *Introduction to the Analytic Hierarchy Process*, Springer, Cham, Sveitsi, 2015, 83 p.
- [90] A. Ishizaka, M. Lusti, How to derive priorities in AHP: a comparative study, *Central European Journal of Operations Research*, Vol. 14, Iss. 4, 2006, pp. 387–400.
- [91] K. Mahoney, K. Coyne, B. Waldron, Lessons Learned in Implementing IEC 61850 Communications Solutions, *Power and Energy Automation Conference*, Spokane, Washington, Yhdysvallat, 2019, 12 p. Saatavissa (viitattu 21.11.2022): <https://selinc.com/api/download/125605/>
- [92] J. Reuter, Helinks LLC, keskustelu, 15.11.2022.
- [93] J. Reuter, Helinks LLC, sähköposti, 23.12.2022.
- [94] UR Family, *Communications Guide*, Product version: 8.2x, GE publication code: 1601-0401-AL1, GE Multilin, 2021 p. 668.
- [95] J. Menezes, L. Frisk, K. Hagman, Flexible product naming, ABB AB, 3 p. Saatavissa (viitattu 2.10.2022): https://library.e.abb.com/public/aacbd56e70c54541b80ee2b754982d70/4CAE000255_en_Flexible_product_naming.pdf

LIITE A: TESTIPROJEKTIN PÄÄKAAVIO

