



Vaasan yliopisto
UNIVERSITY OF VAASA

Emil Uski

Sähköasema-automaation liityntä käytönvalvontajärjestelmään

Käyttöliittymän toteutus Zenon-ohjelmistolla

Tekniikan ja innovaatiojohtamisen yksikkö
Diplomityö
Sähkötekniikka

Vaasa 2025

VAASAN YLIOPISTO
Tekniikan ja innovaatiojohtamisen yksikkö

Tekijä:	Emil Uski		
Tutkielman nimi:	Sähköasema-automaation liityntä käytönvalvontajärjestelmään: Käyttöliittymän toteutus Zenon-ohjelmistolla		
Tutkinto:	Diplomi-insinööri		
Oppiaine:	Sähkötekniikka		
Valvoja:	Kimmo Kauhaniemi		
Tarkastaja:	Hannu Laaksonen		
Ohjaaja:	Antti Vainionpää, Omexom		
Valmistumisvuosi:	2025	Sivumäärä:	71

TIIVISTELMÄ:

Tämän diplomityön tavoitteena on perehtyä sähköasema-automaatioon ja käytönvalvontajärjestelmään sekä käyttöliittymän konfigurointiin ja testaukseen. Käyttöliittymä on osa käytönvalvontajärjestelmää ja toimii rajapintana järjestelmän ja ihmisen välillä. Käytönvalvontajärjestelmä mahdollistaa sähköasemien valvonta- ja ohjaustoimien keskittämisen käyttökeskukseen ja sen avulla voidaan kasvattaa verkon käyttövarmuutta sekä lyhentää häiriöiden kestoaikaa. Sähköasemien automaatio lisääntyy jatkuvasti, minkä takia myös koestajien on tarpeen kehittää osaamistaan keisillä osa-alueella.

Teoriaosuudessa käsitellään ensin käytönvalvontajärjestelmän toimintaa ja rakennetta sekä muutoksia, jotka ovat kasvattaneet käytönvalvontajärjestelmän merkitystä. Tämän jälkeen käsitellään asema-automaatiota prosessitasolta asematasolle, sekä siihen kuuluvia laitteita ja tiedonsiirtoväyliä. Työssä käsitellään myös käytönvalvontajärjestelmässä siirrettävää tietoa sekä erilaisia verkkotopologioita ja -protokollia, joiden avulla tiedonsiirron toimintavarmuutta voidaan lisätä. Työssä käydään läpi myös keskeisiä sähköaseman tiedonsiirto-protokollia, kuten Modbus, IEC 60870-5-101, -103 ja -104 sekä IEC 61850. IEC 61850 mahdollistaa laitevalmistajasta riippumattoman kommunikoinnin laitteiden välillä ja teoriaosuuden loppupuolella käsitellään sen tärkeimpiä ominaisuuksia, joihin kuuluvat ACSI:n laite- ja palvelumalli sekä sovellustason viestintäprotokollat GOOSE, MMS ja SV. Lisäksi käsitellään aikatahdistusta sekä IEC 61850 määrittämää SCL-kieltä ja sen eri tiedostomuotoja.

Teoriaosuuden jälkeen työssä keskitytään käyttöliittymän konfigurointiin Copa-Datan kehittämällä Zenon-ohjelmistolla. Osiossa käydään läpi ohjelmiston keskeisimmät ja tärkeimmät ominaisuudet sekä työvaiheet. Konfigurointiin kuuluu muun muassa yksiviivakaavion, ohjaustoimintojen sekä hälytys- ja tapahtumalistojen toteuttaminen. Lisäksi työn loppupuolella perehdytään asema-automaation testaamiseen Omicron IEDScout-ohjelman ja MBX1-laitteen avulla.

Työn tuloksena saatiin hyvä kokonaiskuva asema-automaatiosta sekä käyttöliittymän konfiguroinnista Zenon-ohjelmistolla. Lisäksi saatiin hyvä käsitys automaation testaamisesta IEDScout-ohjelmalla ja todettiin sen oleva toimiva ratkaisu vastaaviin tilanteisiin, joissa kaikkia IED-laitteita ei ole saatavilla.

AVAINSANAT: Sähköasema-automaatio, HMI, IEC 61850, käytönvalvontajärjestelmä, RTU, SCADA

UNIVERSITY OF VAASA**School of technology and innovations**

Athor:	Emil Uski
Title of the thesis:	Integration of substation automation into the SCADA system: Implementation of a human machine interface using Zenon software
Degree:	Master of Science
Programme:	Electrical Engineering
Supervisor:	Kimmo Kauhaniemi
Evaluator:	Hannu Laaksonen
Instructors:	Antti Vainionpää, Omexom
Year:	2025 Number of pages: 71

ABSTRACT:

The aim of this thesis is to explore substation automation and supervisory control and data acquisition (SCADA) systems, as well as the configuration and testing of a human-machine interface (HMI). The HMI is an integral part of the SCADA system and serves the interface between the operator and the system. Centralizing substation monitoring and control in a control center enhances grid reliability and helps reduce the duration of disturbances. As automation becomes increasingly prevalent in substations, it is essential for commissioning engineers to develop their competence in this area.

The theoretical part begins by examining the operation and structure of SCADA systems, along with changes in the power grid that have increased their importance. This is followed by a discussion on substation automation from the station level down to process level, including related devices and communication buses. The thesis also addresses the types of data transmitted within SCADA systems, and various network topologies and protocols used to enhance communication reliability. Key substation communication protocols and standards such as Modbus, IEC 60870-5-101, -103, -104 and IEC 61850 are also covered. IEC 61850 enables vendor-independent communication between devices and its key features including the ACSI device and service model and the application layer protocols GOOSE, MMS, and sampled values are discussed in detail. The section also covers time synchronization and the SCL language with its associated file types as defined by the standard.

The empirical part focuses on the configuration of the SCADA system using the Zenon software developed by Copa-Data. The key features and configuration steps of the software are presented, such as creating single-line diagram (SLD), implementing command processing, and setting up alarm and event lists. The final part of the work describes the testing of substation automation using Omicron IEDScout software and MBX1 device.

This thesis provided a solid understanding of substation automation and the configuration of the HMI using Zenon. It also offered valuable insight into automation testing with IEDScout software, which proved to be an effective solution in situations where not all IEDs are available.

KEYWORDS: HMI, IEC 61850, SCADA, Substation automation, RTU

Alkusanat

Tämä työ tarjosi erinomaisen mahdollisuuden perehtyä sähköasema-automaatioon ja sen liityntään käytönvalvontajärjestelmään. Haluan kiittää Omexomia mielenkiintoisesta aiheesta sekä kaikkia, joiden kanssa olen saanut työskennellä kuluneen vuoden aikana. Kiitän myös Vaasan yliopiston henkilökuntaa hyvästä opetuksesta ja erityisesti työn valvojaa Kimmo Kauhaniemeä, jonka kurssien myötä kiinnostukseni nykyisiin työtehtäviin heräsi. Lopuksi haluan kiittää opiskelukavereita yhteisistä opiskeluvuosista sekä läheisiäni jatkuvasta tuesta läpi opintojen.

Kokkolassa 23.5.2025

Emil Uski

Sisällys

1	Johdanto	10
1.1	Työn tausta	10
1.2	Työn tavoite ja tutkimuskysymykset	10
1.3	Kohdeyritys	11
2	Sähköasema-automaatio	12
2.1	Käytönvalvontajärjestelmä (SCADA)	12
2.1.1	Toiminta ja rakenne	12
2.1.2	SCADAN roolia kasvattaneet muutokset sähköverkossa	13
2.2	Asema-automaation tasot ja komponentit	14
2.2.1	Primäärilaitteet	16
2.2.2	Intelligent electronic device (IED)	17
2.2.3	Ala-asema (RTU)	17
2.2.4	Käyttöliittymä (HMI)	18
3	Tiedonsiirto sähköverkossa	19
3.1	Siirrettävä informaatio	19
3.2	Verkkotopologiat ja -protokollat	20
3.2.1	Rapid spanning tree protocol (RSTP)	21
3.2.2	High seamless redundancy (HSR)	22
3.2.3	Parallel redundancy protocol (PRP)	24
3.3	Perinteiset sähköaseman tiedonsiirtoprotokollat	26
3.3.1	Modbus RTU ja Modbus TCP	26
3.3.2	IEC 60870-5-101 / 103 / 104	27
3.4	IEC 61850	28
3.4.1	Abstract communication service interface (ACSI)	29
3.4.2	Manufacturing message specification (MMS)	33
3.4.3	Generic object oriented substation event (GOOSE)	34
3.4.4	Prosessiväylä ja sampled values	37
3.4.5	Konfigurointikieli ja tiedostotyypit	38
3.5	Aikatahdistus	39
3.5.1	Precision time protocol (PTP)	39

3.5.2	Network time protocol (NTP)	41
3.5.3	Simple network time protocol (SNTP)	42
4	Konfigurointi ja testaus	43
4.1	Työkalut ja työn tavoitteet	43
4.2	Konfigurointiprosessi	44
4.2.1	Projektin luominen	44
4.2.2	Näyttöjen rakenteet	46
4.2.3	Ajurit	47
4.2.4	Muuttujat	48
4.2.5	Funktiot	49
4.2.6	Visualisointi symboleilla ja yhdistetyillä elementeillä	51
4.2.7	SLD:n luominen	53
4.2.8	Reaktiomatriisi	54
4.2.9	Komennot	55
4.2.10	Tapahtumalista	56
4.2.11	Hälytyslista	57
5	Testaus ja tulokset	58
5.1	IEDScout-ohjelmiston valmistelu ja käyttö	58
5.2	Testaus	60
5.2.1	SLD:n testaus	60
5.2.2	Tapahtumalistan testaus	61
5.2.3	Hälytyslistan testaus	62
5.3	Tulokset	63
6	Yhteenveto	65
	Lähteet	68

Kuvat

Kuva 1. Käytönvalvontajärjestelmän rakenne (Thomas & McDonald, 2015, s. 6).	13
Kuva 2. Sähköasema-automaation tasot ja erilaiset väylätoteutukset (Huotari, 2022).	16
Kuva 3. Verkkotopologiat (Heavy.ai, n.d).	21
Kuva 4. RSTP protokollan toimintaperiaate (Cisco, 2017).	22
Kuva 5. HSR protokollan toimintaperiaate (Cisco, 2024).	24
Kuva 6. PRP protokollan toimintaperiaate (Thiriet, 2021).	26
Kuva 7. IEC 61850 datamalli (Tebekaemi, 2016).	30
Kuva 8. Katkaisijan looginen solmu XCBR (Mackiewicz, 2006).	31
Kuva 9. Single point status ominaisuudet (Mackiewicz, 2006).	32
Kuva 10. IEC 61850 standardin mukainen viittaus datapisteeseen (Mackiewicz, 2006).	32
Kuva 11. Sovellustason protokollien yhteys alemman tason protokoliin (Liang, 2008).	34
Kuva 12. Goose:n uudelleenlähetystoiminto (Lopes, 2015).	35
Kuva 13. Goose-viestin ethernet-kehysten rakenne (Lopes, 2015).	36
Kuva 14. Goose protocol data unit sisältö (Hussain, 2023).	37
Kuva 15. Konfiguroinnin periaate (Peng, 2016).	39
Kuva 16. PTP-protokollan toimintaperiaate (Networklessons.com, n.d.).	40
Kuva 17. Precision time protocol aikaleimat (Wang, 2010).	41
Kuva 18. SNTP protokollan kommunikointi (Ussoli, 2013).	42
Kuva 19. Omicron MBX1-laite (Omicron, 2021).	44
Kuva 20. Zenon Engineering Studio yleisnäkymä.	45
Kuva 21. Service enginen muutosten lataaminen ja käynnistäminen.	46
Kuva 22. Kehysten luominen.	47
Kuva 23. IED-laitteen lisääminen.	48
Kuva 24. Muuttujien tuominen CID-tiedostosta.	49
Kuva 25. Näytönvaihtofunktion luominen.	50
Kuva 26. Skriptin luominen.	50
Kuva 27. Yhdistetyn elementin luominen.	52
Kuva 28. Muuttujien linkittäminen yhdistettyjen elementtien kopioinnissa	53

Kuva 29. Yhdistetyn symbolin luominen.	54
Kuva 30. Reaktiomatriisin luonti.	55
Kuva 31. Komentojen luonti.	56
Kuva 32. IEDScoutin laitteenvaihto	58
Kuva 33. IEDScout aloitusnäyttö.	59
Kuva 34. IED-laitteen simulointi IEDScoutilla.	60
Kuva 35. SLD:n testaus.	61
Kuva 36. Tapahtumalistan testaus.	62
Kuva 37. Hälytyslistan testaus.	63

Taulukot

Taulukko 1. IEC 61850 standardin rakenne (Mackiewicz, 2006).	28
--	----

Lyhenteet

ACSI	Abstract communication service interface
CDC	Common data class
CID	Configured IED Description
DA	Data-attribuutti
DAN	Dual attach node
DO	Dataobjekti
FC	Function constraint
GOOSE	Generic object oriented substation event
HMI	Human-machine interface
HSR	High seamless redundancy protocol
ICD	IED capability description
IED	Intelligent electronic device
LD	Logical device
LN	Logical node
LPIT	Low power instrument transformer
MMS	Manufacturing message specification

MU	Merging unit
NTP	Network time protocol
PRP	Parallel redundancy protocol
PTP	Precision time protocol
RCT	Redundancy control trailer
RSTP	Rapid spanning tree protocol
RTU	Remote terminal unit
SAMU	Stand alone merging unit
SAN	Single attach node
SCADA	Supervisory control and data acquisition
SCD	Substation configuration description
SCL	Substation configuration language
SLD	Single line diagram
SNTP	Simple network time protocol
SSD	System specification description
SV	Sampled value
VMD	Virtual manufacturing device

1 Johdanto

1.1 Työn tausta

Sähköverkkojen automaation ja tiedonsiirron merkitys on kasvanut samaan aikaan kun yhteiskunnan riippuvuus sähköstä on lisääntynyt. Kehittyneet automaatio- ja viestintä-ratkaisut mahdollistavat käytönvalvontajärjestelmän, jonka avulla sähköaseman valvonta- ja ohjaustoimet voidaan keskittää valvomoon. Verkon valvonta- ja ohjaustoimia keskittämällä voidaan muun muassa parantaa verkon toimintavarmuutta sekä lyhentää häiriöiden kestoaikaa. Samaan aikaan sähköaseman suojaus- ja automaatiolaitteiden testaamisesta ja käyttöönotosta vastaavien käyttöönottoinsinöörin eli koestajien tehtävät ovat muuttuneet siten, että niissä vaaditaan entistä laajempaa osaamista ja parempaa kokonaiskuvan ymmärtämistä. Automaation ja tietoliikenteen ollessa yhä keskeisemmässä roolissa sähköaseman toiminnassa on tärkeää kehittää koestajien osaamista myös näillä osa-alueilla.

1.2 Työn tavoite ja tutkimuskysymykset

Tämän diplomityön tavoitteena on selvittää, kuinka Copa-Datan kehittämällä Zenon-ohjelmistolla voidaan toteuttaa sähköaseman käytönvalvontajärjestelmän käyttöliittymä eli human machine interface (HMI). Teoriaosuudessa perehdytään monipuolisesti sähköasema-automaatioon koestajan näkökulmasta. Työn toisena tavoitteena onkin tarjota nykyisille ja tuleville koestajille parempi ymmärrys asema-automaatiosta.

Tutkimuskysymykset ovat:

1. Mistä sähköasema-automaatio koostuu?
2. Miten toteutetaan HMI Zenon-ohjelmistolla?
 - a. Mitä lähtötietoja sähköasemasta tarvitaan konfigurointia varten?
 - b. Mitkä ovat keskeisimmät komponentit ja työvaiheet konfiguroinnissa?
3. Miten asema-automaation testauksessa voidaan hyödyntää Omicron IEDScoutia?

1.3 Kohdeyritys

Infratek Finland Oy, joka toimii nykyisin Omexom-nimen alla, on sähkönjakelu- ja sähkön-siirtoinfrastruktuuriin erikoistunut yritys, joka toimii ympäri Suomen. Omexom-nimen alla toimii myös muita yhtiöitä yhteensä 40 eri maassa työllistäen noin 24 000 työntekijää (Omexom, 2025). Omexom on osa ranskalaista VINCI Energies -konsernia, joka puolestaan on osa ranskalaista VINCI-konsernia.

Omexomin visiona on edistää globaalia energian saatavuutta sekä tehdä energiamurros mahdolliseksi kestäväen kehityksen periaatteiden mukaisesti. Seuraavien viiden vuoden aikana globaalin energiasektorin uskotaan muuttuvan enemmän kuin viimeisen 50 vuoden aikana. Omexom pyrkii toteuttamaan visiotaan tarjoamalla ratkaisuja kaikille niille, jotka tuottavat, muuntavat tai kuljettavat sähköä. Siten Omexom auttaa muun muassa energiantuottajia ja verkko-operaattoreita täyttämään tehtävänsä navigoidessaan kehityvässä maailmassa.

2 Sähköasema-automaatio

Riippuvuus sähköstä kasvaa nopealla vauhdilla maailmanlaajuisesti, mikä vääjäämättä asettaa uusia vaatimuksia sähkönsiirron ja -jakelun luotettavuudelle, energiatehokkuudelle sekä ympäristövaikutuksille. McDonaladin ja Thomasin (2015, s. 1) mukaan energiakulutuksen uskotaan, jopa kolminkertaistuvan vuoteen 2050 mennessä. Sähköverkoissa muutos on näkynyt erityisesti automaation ja tiedonsiirron lisääntymisenä. Muutosajureina automaatiojärjestelmien yleistymisessä sähköverkoissa ovat olleet pienentyneet automaatiolaitteiden kustannukset, ethernet-pohjaisen tiedonsiirron kehittyminen sekä tiedonsiirtoprotokollien standardointi. Sähköaseman automaatio on nykyisin tärkeässä roolissa mahdollistaessaan pääsyn voimajärjestelmään sekä paikallisesti, että valvomosta kaukokäytön avulla. Asema-automaation avulla järjestelmää saadaan ohjattua sekä kerättyä tietoa sen tapahtumista ja tilasta.

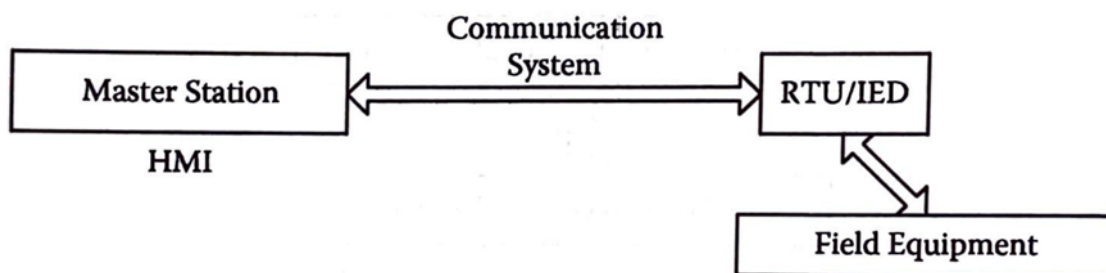
2.1 Käytönvalvontajärjestelmä (SCADA)

2.1.1 Toiminta ja rakenne

Käytönvalvontajärjestelmä, eli supervisory control and data acquisition (SCADA) on tietokonepohjainen valvonta- ja ohjausjärjestelmä. Sen avulla voidaan valvoa ja ohjata laajoja hajautettuja järjestelmiä, kuten sähköverkkoja. Sen tehtävänä on kerätä ja välittää tietoa sekä ohjauksia prosessilaitteilta valvomoon ja toisinpäin. Tämä mahdollistaa kaukokäytön, jonka myötä laajankin järjestelmän valvonta ja ohjaustoiminnot voidaan keskittää yhteen paikkaan.

Clarcken (2004, s. 3) mukaan käytönvalvontajärjestelmän pääkomponentteja ovat keskusasema, HMI, tiedonsiirtojärjestelmä, remote terminal unit (RTU) sekä asema- ja kenttätason laitteet. Kuvassa 1 on havainnollistettu järjestelmän rakennetta yksinkertaisimmillaan. HMI on keskeinen osa käytönvalvontajärjestelmää, sillä se toimii rajapintana ihmisen ja järjestelmän välillä. RTU eli ala-asema toimii puolestaan rajapintana sähköaseman sisäisen ja ulkopuolisen liikenteen välillä. Se välittää kenttä- ja prosessitason laitteiden

lähettämät tiedot valvomon keskusasemalle ja toisinpäin. RTU:hun kytketty kenttälaite voi olla esimerkiksi sähköverkon mittausdataa ja kytkinlaitteiden tilatietoja välittävä suo-
jarele. Luvussa 3 perehdytään yksityiskohtaisemmin aseman sisäiseen tietoliikentee-
seen.



Kuva 1. Käytönvalvontajärjestelmän rakenne (Thomas & McDonald, 2015, s. 6).

2.1.2 SCADAN roolia kasvattaneet muutokset sähköverkossa

Yksi tekijä sähköverkon muutoksen taustalla on kasvava uusiutuvan energian osuus sähkötuotannossa. Kuusiston (2023) mukaan perinteisten generaattoreiden liike-energiaa eli inertiaa on voitu hyödyntää hetkellisesti vikatilanteissa ja nopeissa tuotantovaihteluissa. Generaattoreiden inertia on antanut aikaa reagoida vikoihin sekä tuotantovaihteluihin ja palauttaa verkko normaalitilaan. Nykyään esimerkiksi tuulivoimalat liitetään verkkoon taajuusmuuttajien välityksellä, eivätkä ne tarjoa inertiaa samoin kuin perinteiset voimalaitokset. Ne reagoivat verkon taajuuden ja jännitteen muutoksiin eri tavoin kuin perinteiset voimalaitokset ja lisäävät sähköjärjestelmän alttiutta häiriöille. Käytönvalvontajärjestelmän avulla verkon tilassa tapahtuviin muutoksiin voidaan reagoida nopeasti. Erilaiset häiriöt ovat entistä merkittävämpiä, koska sähköstä ollaan yhä enemmän riippuvaisia esimerkiksi lämmöntuotannon ja liikenteen sähköistyessä.

Elovaaran (2011) mukaan verkon käytön tehtävänä on vastata energian siirtoprosessin hallinnasta. Se tarkoittaa, että tuotettu ja kulutettu sähkö pyritään pitämään tasapainossa, sekä siirto pyritään toteuttamaan mahdollisimman taloudellisesti ja käyttövarmasti. Verkon ja sen käytön teknistä laatutasoa voidaan parantaa keskittämällä verkon

ohjaus- ja valvontatoimia. Se tarkoittaa, että esimerkiksi toimitusvarmuus paranee, häiriöiden kesto aika lyhenee.

Uusiutuvan energian lisääntymisen ohella on lisääntynyt myös hajautettu sähköntuotanto, mikä on luonut uusia haasteita sähköverkkoon. Lakervin ja Partasen (2008, s. 209) mukaan hajautettu tuotanto vaikuttaa, niin jännitteeseen kuin järjestelmien suojausseenkin. Vaikutuksia verkkoon voi aiheutua esimerkiksi aurinko- ja tuulivoimaloista tai sähköenergiavarastosta. Perinteisesti sähköntuotanto on ollut hyvin keskitettyä ja sähkö on liikkunut pääosin yksisuuntaisesti sähköasemilta johtojen ja jakelumuuntamoiden kautta kulutuskohteisiin. Se tarkoittaa, että käyttöpaikan jännite on saatu selville vähentämällä sähköaseman keskijännitekiskon jännitteestä muuntajan ja johtojen jännitteenalenemat. Nyt kuitenkin hajautetun sähkön tuotannon myötä on riskinä muun muassa jännitteen nousu haitallisesti. Jännitteenousu voi muodostua haitalliseksi, jos verkkoon liitetty hajautettu tuotanto on paljon suurempi kuin alueen kulutus. Samalla uudet tuotantokohteet ja energiavarastot voivat vaikuttaa myös verkon vikavirtoihin esimerkiksi siten, että suojausalueen näkemät vikavirrat eivät välttämättä ole riittävän suuria, jolloin suojaus ei toimi odotetusti.

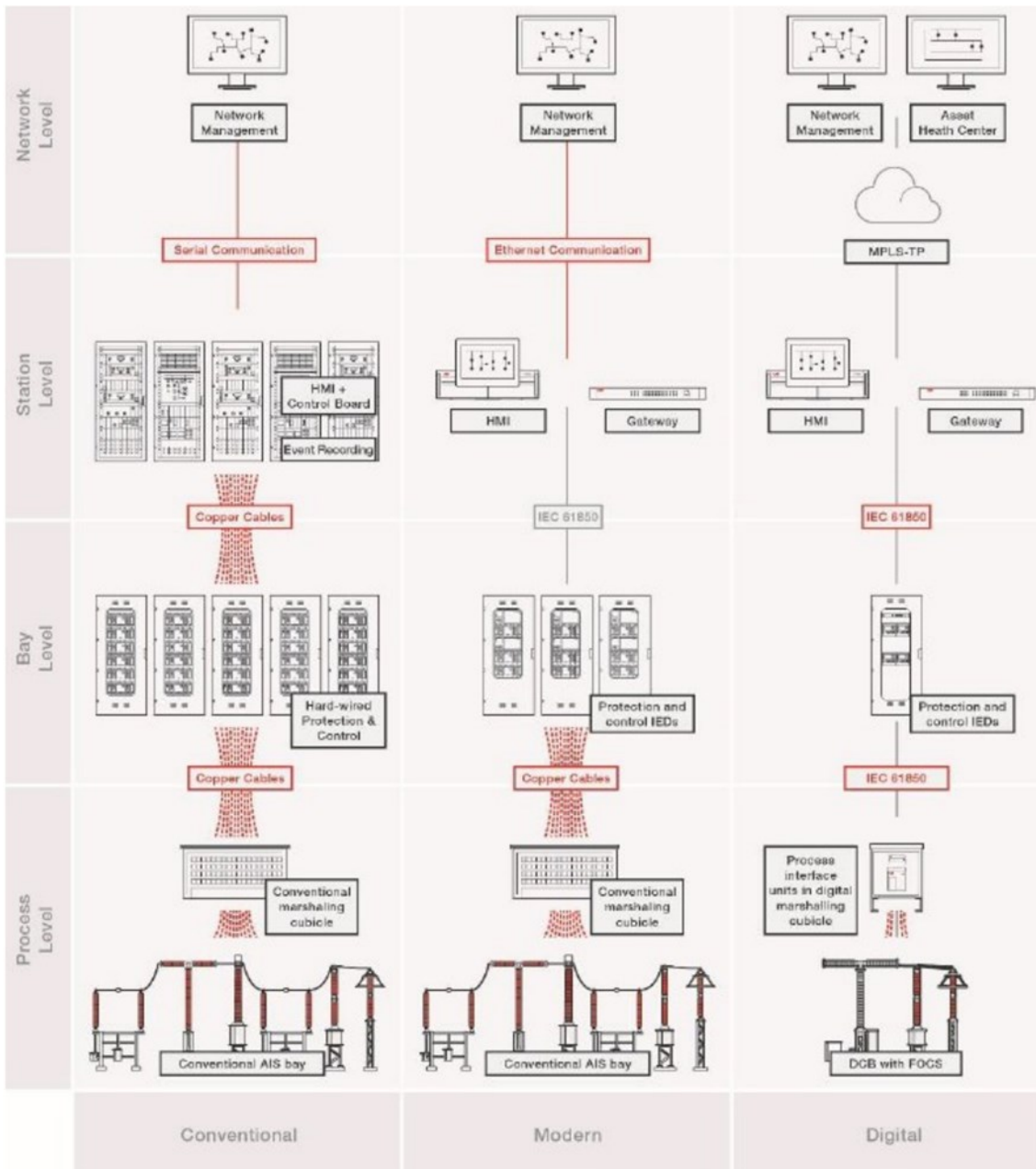
Tulevaisuudessa saarekekäytöt saattavat olla mahdollisia vikatilanteissa ja pahimmillaan ne voivat olla hallitsemattomia ja myös maadoitus voi olla puutteellinen. Saarekekäyttö vaatii muun muassa jatkuvaa jännitteen sekä taajuuden valvontaa muiden toimintojen lisäksi. Näiden tekijöiden myötä verkon rakenne monimutkaistuu, mikä tarkoittaa, että verkosta on kerättävä yhä enemmän tietoa, jotta se toimisi suunnitellusti.

2.2 Asema-automaation tasot ja komponentit

Elovaaran ja Haarlan (2011, s. 386) mukaan sähköaseman automaatiojärjestelmä tarkoittaa järjestelmää, joka vastaa paikallisten manuaalisten ja automaattisten toimintojen lisäksi tiedonsiirron toimivuudesta. Se huolehtii tiedonsiirtoliitynnöistä ja tiedon muotoilemisesta siten, että eri laitteet kykenevät kommunikoimaan keskenään.

Thomasin ja McDonaldin (2015, s. 148) mukaan asema-automaatio ja sen laitteet voidaan jakaa eri tasoihin, joita on havainnollistettu kuvassa 2. Kuvasta voidaan havaita myös, miten tasojen välinen tiedonsiirto on toteutettu. Suomessa nykyiset sähköasemat muistuttavat keskellä olevaa modernia mallia. Mallissa alimmaisena on prosessitaso, jossa sijaitsevat primääripiirien laitteet. Prosessitason laitteet kommunikoivat ylempänä olevan kenttätason kanssa, jota kutsutaan joskus myös suojaus- ja ohjaustasoksi. Kenttätason laitteet kuten suojarieleet ja kenttäohjauksyksiköt kommunikoivat prosessitason lisäksi keskenään sekä yllä olevan asematason laitteiden kanssa, joihin kuuluvat muun muassa HMI ja RTU. Asematason RTU kommunikoi aseman ulkopuolella olevan verkkotason kanssa, jota pitkin yhteys ulottuu valvomoon saakka.

Kenttä- ja asematason välinen tiedonsiirto tapahtuu asemaväylän avulla sekä modernilla, että digitaalisella sähköasemalla, mutta kenttä- ja prosessitason väliset toteutukset poikkeavat toisistaan. Kuvassa oikealla on digitaalisen sähköaseman malli, jossa prosessi- ja kenttätason välinen tiedonsiirto toteutetaan perinteisen kuparikaapeloinnin sijaan ethernetiin perustuvalla prosessiväylällä. Prosessiväylää hyödyntävät digitaaliset sähköasemat ovat kuitenkin vielä harvinaisia Suomessa. Prosessiväylää käsitellään tarkemmin IEC 61850 standardin yhteydessä luvussa 3.4. Seuraavissa kappaleissa käsitellään tarkemmin eri tasojen laitteita ja niiden rooleja.



Kuva 2. Sähkösäätö- ja automaatio-asteet ja erilaiset välitoteutukset (Huotari, 2022).

2.2.1 Primäärilaitteet

Useat käytönvalvontajärjestelmässä siirrettävät tiedot ovat peräisin primäärilaitteilta tai osoitettu niille. Sähkösäätö- ja automaatio-asteisiin kuuluvat muun muassa mittamuuntajat, katkaisijat ja erottimet. Katkaisijoissa ja erottimissa olevien koskettimien avulla saadaan selville niiden asentotiedot käytönvalvontajärjestelmään. Vastaavasti mittamuuntajien

avulla saadaan virta- ja jännitetiedot ja niiden myötä tehojen arvot käytönvalvontajärjestelmään. Perinteisesti primäärilaitteilta on toteutettu kullekin signaalille oma kaapeliyhteys suojarielelle tai kenttäohjausyksikölle. Digitaalisella sähköasemalla primäärilaitteiden yhteydessä hyödynnetään liittymisyksikköjä (engl. merging unit), joilla tiedot voidaan muuntaa digitaaliseen muotoon, jolloin ne voidaan siirtää prosessiväylää pitkin suojarieleille ja kenttäohjausyksiköille. Tällöin kaapelointitarve vähenee huomattavasti, kun useat eri signaalit voidaan siirtää yhdellä väyläkaapelilla relekaapille. Toimintavarmuuden ja kaistanleveyden kasvattamiseksi väyläkaapeleita on kuitenkin useampia.

2.2.2 Intelligent electronic device (IED)

IED tulee sanoista Intelligent electronic device ja se tarkoittaa älykästä elektronista laitetta kuten suojarielettä tai kenttäohjausyksikköä. Nämä laitteet ovat avainasemassa sähköaseman automaatiojärjestelmässä, sillä ne mahdollistavat suojaus- mittaus-, ohjaus- ja automaatiotoimintoja (Thomas, McDonald, 2015, s. 137). Ne ovat yhteydessä primäärilaitteisiin sekä asemaväylään, jota pitkin ne voivat kommunikoida keskenään ja RTU:n kanssa.

2.2.3 Ala-asema (RTU)

RTU, jota kutsutaan myös ala-asemaksiksi, toimii tiedonkeruu- ja ohjausyksikkönä käytönvalvontajärjestelmässä. Sen avulla sähköasema liittyy aseman ulkopuolisiin tiedonsiirtojärjestelmiin. McDonaldin ja Thomasin (2015, s. 25) mukaan ala-asema kerää järjestelmän dataa kenttälaitteista, prosessoi sen ja lähettää sen keskusasemalle. Samaan aikaan ala-asema myös ottaa vastaan käskyjä keskusasemalta ja välittää niitä kenttälaitteille. Ala-aseamalla on keskeinen rooli käytönvalvontajärjestelmässä, sillä se kykenee kommunikoiamaan useilla eri tiedonsiirtoprotokollilla ja muuntamaan tiedot sopivaan muotoon seuraavalle laitteelle. Ala-aseman yksi tärkeistä tehtävistä onkin toimia protokollamuunnimena. Eri tiedonsiirtoprotokollia käsitellään tarkemmin luvussa 3.

Clarcken ja muiden (2004) mukaan ala-asema koostuu tyypillisesti ohjausprosessorista, siihen liittyvästä muistista, teholähteestä, erilaisista tiedonsiirtorajapinnoista, kuten analogisista tuloista ja lähdöistä sekä digitaalisista tuloista ja lähdöistä. Nykyisin kuitenkin kommunikointi ala-aseman kanssa perustuu useimmiten valokuitua tai kierrettyä paria hyödyntäviin ethernet-pohjaisiin ratkaisuihin, jolloin muille tuloille tai lähdöille ole tarvetta.

2.2.4 Käyttöliittymä (HMI)

HMI tarkoittaa tässä tapauksessa käytönvalvontajärjestelmän ja ihmisen välistä rajapintaa, joka koostuu nykyisin usein tietokoneesta ja näyttöpäätteestä. McDonaladin ja Thomasin (2015, s. 54) mukaan tavoitteena on, että HMI mahdollistaa käyttäjäystävällisen järjestelmän operoinnin. Se tarkoittaa, että käyttäjä saa hyvän kuvan järjestelmän tilasta ja rakenteesta sekä kykenee muun muassa tarvittaessa helposti ohjaamaan kytkinlaitteita kuten esimerkiksi erottimia. Sähköaseman HMI-näytöltä nähdään yleensä vähintään hälytyslista, tapahtumalista sekä yksiviivakaavio SLD, jonka kautta voidaan myös toteuttaa ohjauskäskyjä.

3 Tiedonsiirto sähköverkossa

Nykyisin tiedonsiirto on erittäin keskeisessä roolissa sähköverkossa, sillä sen avulla mahdollistetaan verkon reaaliaikaiset valvonta- ja ohjaustoiminnot. Lakervin ja Partasen (2008, s. 245) mukaan valvomon ja sähköasemien välillä liikkuu jatkuvasti paljon tärkeää tietoa ja osa siitä on hyvin aikakriittistä. Sen vuoksi eri toiminnoille on määritelty erilaisia tiedonsiirron luotettavuus-, kapasiteetti-, ja aikakriittisyysvaatimuksia. Edellä mainittujen vaatimusten lisäksi otetaan huomioon hinta, kun mietitään kuhunkin tarkoitukseen sopivinta tiedonsiirtomenetelmää.

3.1 Siirrettävä informaatio

Elovaaran ja Haarlan (2011, s. 394) mukaan siirrettävä informaatio voidaan jakaa sen muodon ja tärkeyden perusteella ohjauksiin, mittauksiin, hälytyksiin ja ilmoituksiin. Ohjauksiin kuuluvat käskyt, jotka aiheuttavat ohjaustoimenpiteitä asemalla, kuten katkaisijan aukaisu tai erottimen yhdistäminen. Ohjauksiin kuuluu myös esimerkiksi käämikytinten asetusarvojen muuttaminen. Yleensä ohjauksiin liittyvältä tiedonsiirrolta vaaditaan nopeutta ja varmuutta vahinkojen välttämiseksi, minkä takia ohjauksiin liittyvälle informaatiolle annetaan yleensä korkein prioriteetti. Näiden korkean prioriteetin ohjauksen perillemeno varmistetaan yleensä ilmoituksella. Ilmoitukset tulevat yleensä näkyviin HMI:n tapahtumalistaan. Katkaisijoiden ja erottimien tilatietojen muuttuminen voidaan havaita usein myös HMI:n yksiviivakaaviosta.

Mittausinformaatio voi olla analogista ja jatkuvaa tai se voi olla diskreetti kuten edellä käsitellyt katkaisijoiden ja erottimien tilatiedot. Mittaustietojen avulla suojaareleet tulkitsevat oman verkonosansa tilannetta jatkuvasti ja tarvittaessa toteuttavat suojaustoimintoja. Mittaustietojen avulla voidaan myös tarkastella eri verkonosien kuormituksia valvomosta.

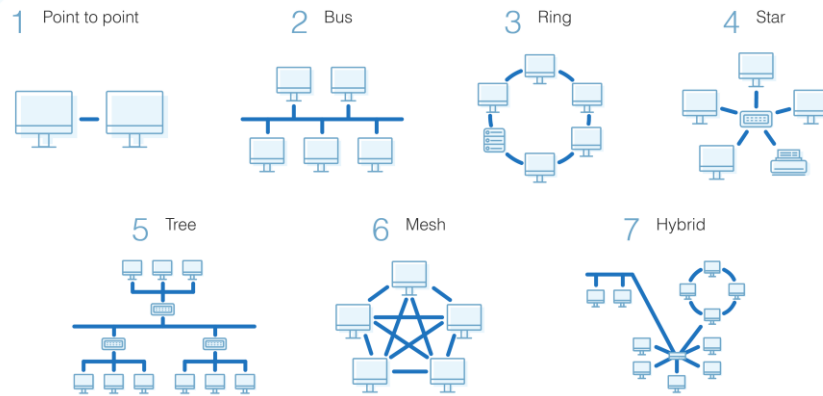
Hälytystietojen välittäminen valvomoon on tärkeä toiminto käytönvalvontajärjestelmässä, sillä niiden avulla voidaan nopeuttaa vian paikantamista, sen korjauksen

aloittamista tai parhaimmassa tapauksessa estää vikojen ja vahinkojen syntyminen. Hälytyksiä varten käytönvalvontajärjestelmään luodaan yleensä oma hälytyslista, josta hälytyksiä voidaan tarkastella niiden saapumisjärjestyksessä. Hälytyksiä voi olla tarpeen luokitella niiden vakavuuden perusteella esimerkiksi asettamalla hälytyksen perään teksti ”ei kiireellinen”.

3.2 Verkkotopologiat ja -protokollat

Verkkotopologia määrittää miten tietoliikenneverkon laitteet ovat yhdistetty keskenään. Topologialla on merkitystä esimerkiksi verkon suorituskyvyn ja luotettavuuden kannalta. McDonaldin ja Thomasin (2015, s. 79) mukaan voidaan puhua fyysisestä ja loogisesta topologiasta. Fyysinen topologia tarkoittaa miten laitteet ovat fyysisesti yhdistetty toisiinsa. Looginen topologia puolestaan tarkoittaa, miten tieto kykenee liikkumaan näitä yhteyksiä pitkin. Vaikka fyysinen yhteys olisi olemassa, mutta laitteiden portit ovat esto-tilassa, niin loogista yhteyttä ei ole. Yleisimpiä topologioita ovat väylä (bus), rengas (ring), tähti (star), puu (tree), mesh ja point to point. Kuvasta 3 voidaan havaita kunkin topologian tyypillinen rakenne. Näihin topologioihin perustuen on kehitetty erilaisia verkkoprotokollia muun muassa toimintavarmuuden parantamiseksi. Asema-automaatiossa hyödynnetään ainakin Rapid Spanning Tree Protocol (RSTP), High Seamless Redundancy (HSR) ja Parallel Redundancy Protocol (PRP) protokollia, joita käsitellään seuraavaksi tarkemmin.

Network Topology Types



Kuva 3. Verkkotopologiat (Heavy.ai, n.d).

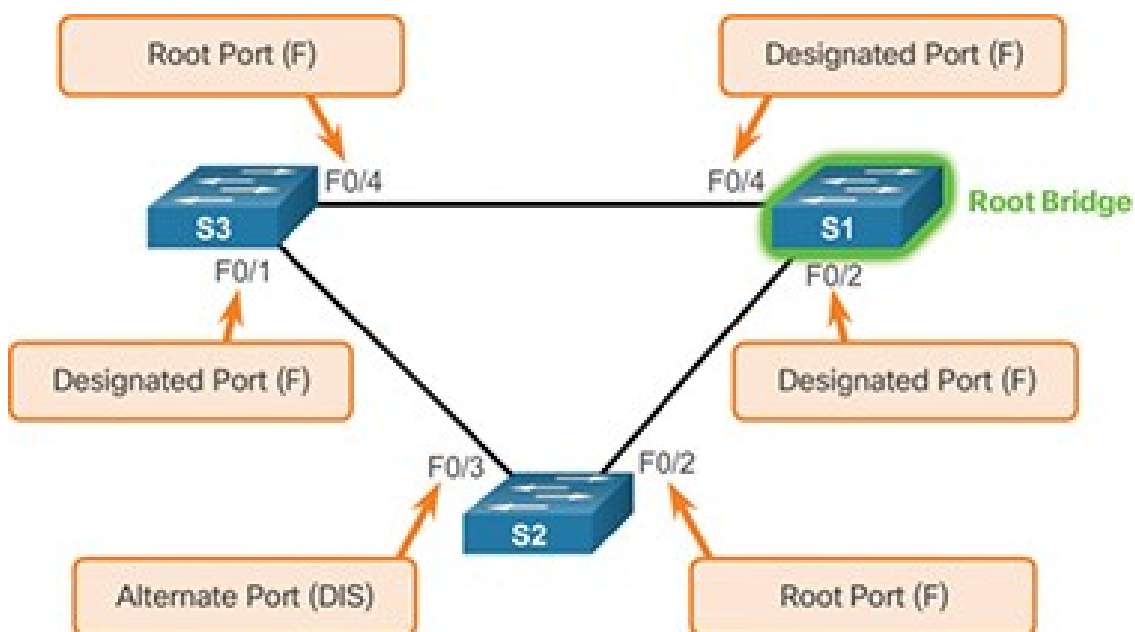
3.2.1 Rapid spanning tree protocol (RSTP)

RSTP on verkon toimintavarmuuden parantamiseksi kehitetty protokolla. Wajdok (2003) mukaan RSTP perustuu loogiseen puutopologiaan. Fyysinen topologia voi kuitenkin olla esimerkiksi tähti, rengas tai silmukka, mikä lisää redundanttisuutta ja toimintavarmuutta. Kytkimien ristiin kytkennän riskinä on kuitenkin se, että kytkimet alkavat lähettämään silmukassa viestejä uudelleen loputtomasti, jolloin yhteys ylikuormittuu ja lakkaa toimimasta. Loogisten silmukoiden estämiseksi RSTP estää osan porteistaan siten, että kullekin laitteelle on vain yksi reitti, jolloin muodostuu looginen puutopologia. Tyypillisesti asemaväylän kytkimet on yhdistetty toisiinsa RSTP protokollaa hyödyntävän rengastopologian avulla.

Kuvassa 4 havainnollistetaan RSTP protokollan periaatetta. Yksi silloista on root bridge, joka toimii keskussolmuna. Porttien tilat pyrkivät asettumaan siten, että kullakin solmulla on mahdollisimman lyhyt reitti keskussolmulle. Designated port tarkoittaa porttia, joka välittää viestejä keskussolmusta poispäin ja root port tarkoittaa porttia, joka välittää viestejä keskussolmuun päin. Alternate port on varaportti, joka voidaan ottaa käyttöön, mikäli liikenne katkeaa ensisijaisen portin kautta. Porteilla on 3 tilaa, jotka ovat

discarding (DIS), forwarding (F) ja learning. Discarding tilassa portti ei osallistu verkon toimintaan eikä sen kautta ole yhteyttä. Learning tilassa portti ei välitä liikennettä, vaan opettelee verkon laitteiden MAC-osoitteita sekä minkä portin kautta ne tavoitetaan. Forwarding tilassa portti toimii kuten learning tilassa, jonka lisäksi se välittää viestejä eteenpäin.

Kuvan 4 avulla voidaan havainnollistaa protokollan toimintaa. Porttien tilojen perusteella nähdään, että kyseessä on looginen puutopologia ja fyysinen rengastopologia. Tiedonsiirto toimii suoraan laitteiden S1 ja S3 välillä sekä S1 ja S2 välillä. Mikäli viian seurauksena laitteiden S1 ja S2 välinen yhteys lakkaisi toimimasta niin laitteen S2 varaportti (Alternate Port) tulisi käyttöön automaattisesti ja laitteiden S1 ja S2 välinen tiedonsiirto tapahtuisi laitteen S3 kautta.



Kuva 4. RSTP protokollan toimintaperiaate (Cisco, 2017).

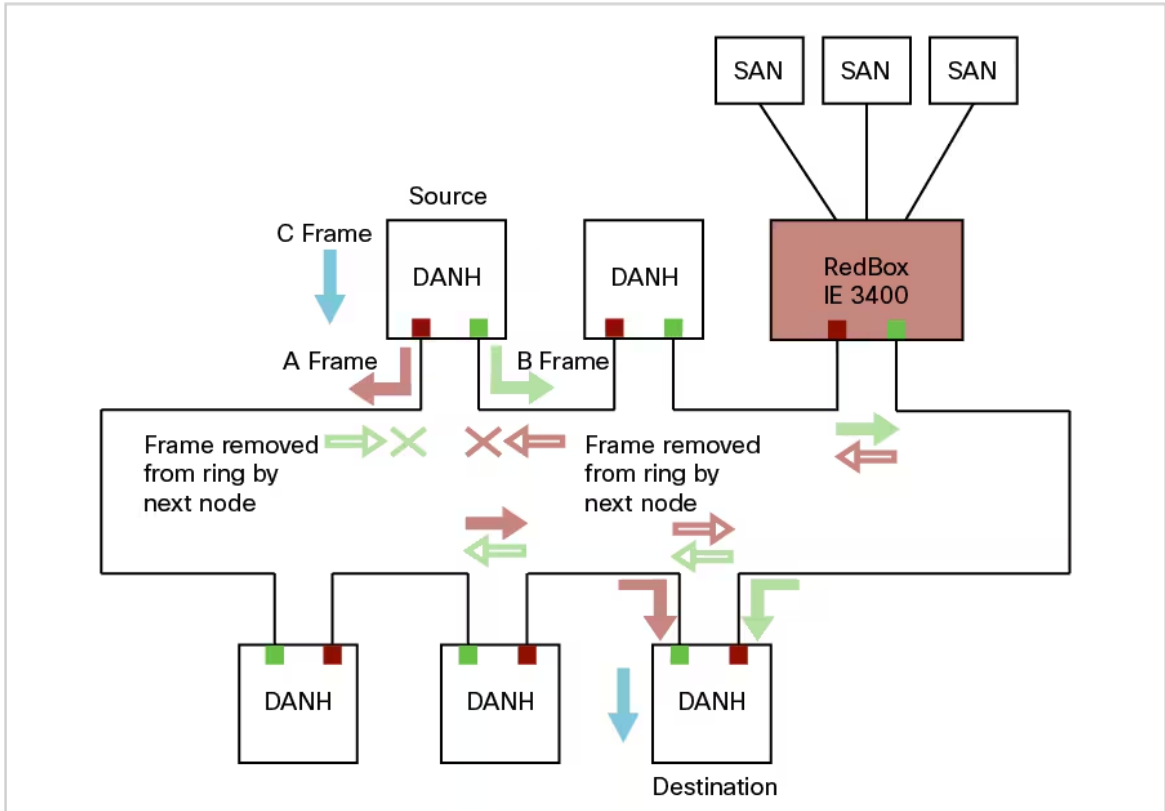
3.2.2 High seamless redundancy (HSR)

Nguyen (2016) mukaan HSR protokolla perustuu rengastopologiaan, jossa viesti lähetetään renkaassa samanaikaisesti kahteen eri suuntaan, mikä voidaan havaita kuvasta 5. Tähän renkaaseen kytkettyjä laitteita kutsutaan solmuiksi. Solmut voivat olla tavallisia

kahdella portilla varustettuja HSR-solmuja eli Dual attached Node HSR (DANH), joilla on myös yhteinen MAC-osoite. Kun solmu vastaanottaa viestin eikä se ole viestin lähde tai ainoa kohde niin viesti lähetetään eteenpäin toisen portin kautta, kuin mistä se vastaanotettiin. Solmun ollessa viestin ainoa kohde alkaa se prosessoimaan ensimmäiseksi saapunutta viestiä. Kun viestin toinen kopio saavuttaa solmun niin se poistetaan eikä sitä välitetä eteenpäin. Lähettämällä kaksi kopiota viestistä samanaikaisesti renkaassa eri suuntiin voidaan tiedonsiirrosta tehdä luotettavampaa, sillä yhden yhteyden katkeaminen ei vielä estä kommunikointia.

Kaikilla laitteilla ei kuitenkaan ole HSR- välitystoimintoa eivätkä ne kykene liittymään suoraan HSR-renkaaseen. Tätä varten renkaaseen kytketään redundancy box (Redbox), jonka kautta kukin yksittäisellä liitännällä varustettu solmu eli single attached node (SAN) voi liittyä renkaaseen. Viestinvälittäjänä HSR-renkaassa redbox toimii muuten täysin DANH tavoin.

Mikäli halutaan yhdistää useita HSR-renkaita, niin tarvitaan Quadruple port device (Quadbox), joka toimii kahden renkaan välisolmuna. Quadbox koostuu neljästä portista, joista toinen pari toimii solmuna toisessa renkaassa ja toinen toisessa. Nämä kaksi paria välittävät viestit toisilleen, jolloin tiedonsiirto toimii renkaasta toiseen.



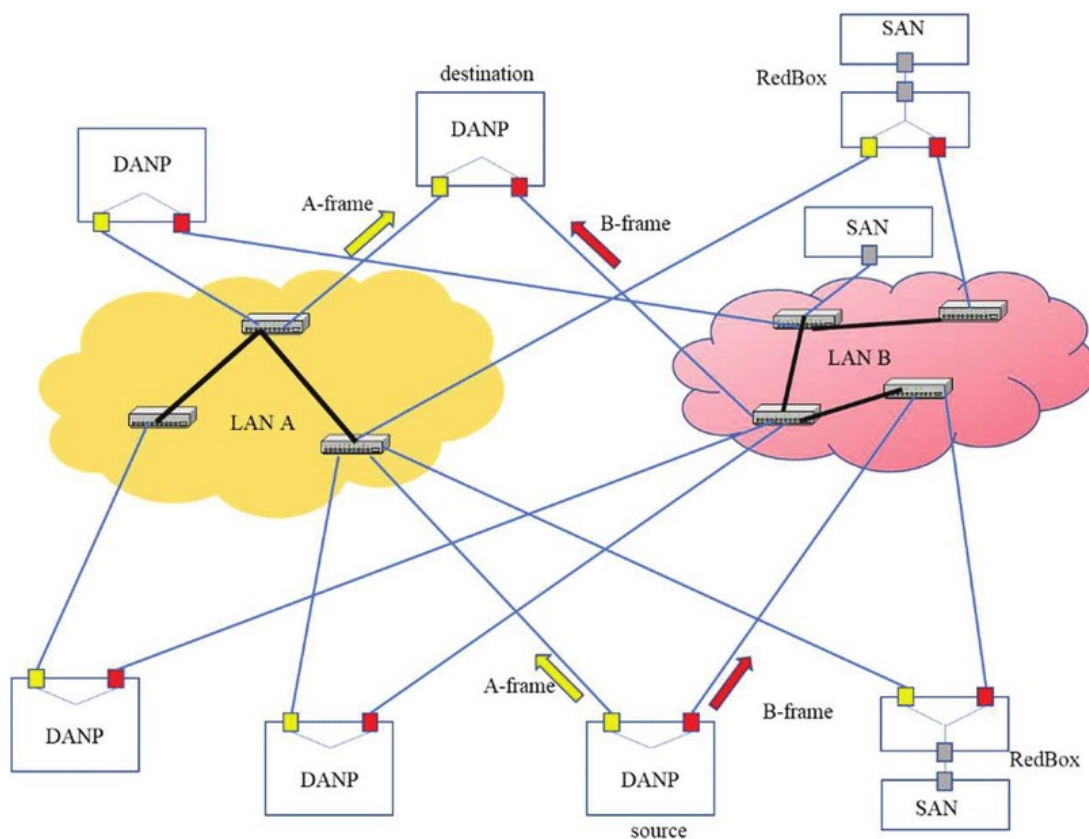
Kuva 5. HSR protokollan toimintaperiaate (Cisco, 2024).

3.2.3 Parallel redundancy protocol (PRP)

PRP on yksi sähköaseman tiedonsiirrossa suosituista protokollista. Mingxu (2023) mukaan PRP koostuu kahdesta itsenäisestä renkaasta, jotka eivät ole yhteydessä toisiinsa. Kumpikin rengas muodostaa oman LAN-verkon, joista käytetään usein nimitystä LAN A ja LAN B. Näiden verkkojen erottamisen helpottamiseksi kummankin verkon kaapeloinnissa saatetaan käyttää eri värisiä kaapeleita. Verkoissa on pääasiassa kahden tyyppisiä solmuja. Doubly attached node implementing PRP (DANP) -solmut ovat PRP-protokollaa tukevia laitteita, jotka kykenevät liittymään suoraan kumpaankin LAN-verkkoon samanaikaisesti. Single attached node (SAN) -solmut ovat laitteita, jotka kykenevät liittymään vain yhteen LAN-verkkoon eikä redundanttisuus toteudu tällöin. Redboxin avulla kuitenkin voidaan liittää myös SAN-solmut kumpaankin LAN-verkkoon samanaikaisesti. PRP:n rakenteen ansiosta tiedonsiirrosta saadaan hyvin toimintavarma ja vikasietoinen, sillä

toisen verkon kaatuminen ei vielä estä tiedonsiirtoa. Tämän vuoksi PRP protokollaa hyödynnetään usein IED-laitteiden tiedonsiirrossa.

Kuvassa 6 havainnollistetaan PRP toimintaa. DANP-solmussa datakehys monistetaan kahdeksi ja lähetetään kumpaankin verkkoon. Ennen lähettämistä kehykseen lisätään vielä redundancy control trailer (RCT), jotta viestin vastaanottaja erottaa kumpi kopio on kyseessä. RCT sisältää muun muassa sekvenssinumeron, josta tunnistetaan saman kehyksen kopiot sekä tiedon kumpaan LAN-verkkoon kehys on lähetetty. Näiden tietojen avulla vastaanottaja tietää onko kyseessä jo saapuneen kehyksen kopio. Mikäli kyseessä on aikaisemmin saapuneen kehyksen kopio niin se poistetaan. SAN-solmussa viestin kopiointia ei tapahdu vaan solmu lähettää vain yhden kehyksen siihen LAN-verkkoon, johon se on kytketty. Jos SAN-solmu on kytketty redboxiin niin redbox kopioi kehyksen, lisää siihen RCT:n ja lähettää kumpaankin LAN-verkkoon.



Kuva 6. PRP protokollan toimintaperiaate (Thiriet, 2021).

3.3 Perinteiset sähköaseman tiedonsiirtoprotokollat

Helpottaakseen järjestelmien yhteensopivuutta on tiedonsiirtoprotokollia standardisoitu. Thomasin ja McDonaldin (2015, s. 101) mukaan yleisimpiä sovellustason tiedonsiirtoprotokollia käytönvalvontajärjestelmissä ovat IEC 61850, IEC 60870-5-101, IEC 60870-5-104 sekä Modbus.

3.3.1 Modbus RTU ja Modbus TCP

Edellä mainituista protokollista vanhin on Modbus, joka perustuu laitteiden master-slave-suhteisiin tai toisin ilmaistuna client-server-suhteisiin. Modbus organization (2006) mukaan Modbus RTU on sarjaliikenteeseen perustuva protokolla, jossa samaan väylään voidaan lähtökohtaisesti kytkeä masterin kanssa maksimissaan 31 palvelinlaitetta, mutta määrää voidaan kasvattaa, mikäli hyödynnetään RS-485 protokollaa ja toistimia.

Protokollan tiedonsiirto on kyselypohjaista, mikä tarkoittaa, että tiedonsiirto tapahtuu masterin aloitteesta. Master lähettää väylään viestin, jonka kaikki väylän laitteet vastaanottavat, mutta siihen vastaavat vain palvelimet, joille se on modbus-osoitteen perusteella tarkoitettu. Fyysisen kerroksen muodostavat tyypillisesti sarjaväylän standardit RS-232 ja RS-485. On hyvä huomata, että yleensä RS-232 kaapeloinnin maksimipituus on alle 20 metriä kun taas RS-485 kohdalla se on jopa 1200 metriä. Myös RS-232 topologia asettaa omat rajoitteensa sillä se toimii point-to-point periaatteella vain kahden laitteen välillä. RS-485 puolestaan mahdollistaa 31 palvelinlaitetta tai jopa enemmän, jos käytetään toistimia.

Modbus TCP on ethernetiin perustuva protokolla, joka mahdollistaa nopeamman tiedonsiirron ja mahdollisuuden pidempään kaapelointiin. Viestien reititys tapahtuu IP-osoitteen perusteella.

3.3.2 IEC 60870-5-101 / 103 / 104

IEC 101 on sarjaliikenteeseen perustuva protokolla, joka kehitettiin kaukokäytön tarpeisiin (McDonald, 2015, s.62). ABB:n (2013) mukaan IEC 101 mahdollistaa balansoimattoman ja balansoidun tiedonsiirron. Balansoimattomassa tilassa se toimii master-slave periaatteella ja tiedonsiirto tapahtuu vain masterin aloitteesta, kun se aloittaa kyselyn. Balansoidussa tilassa jokainen laite voi aloittaa tiedonsiirron ilman tietyn masterin kyselyä. Tällöin esimerkiksi tapahtumat voidaan lähettää viiveettä, kun masterin kyselyä ei tarvitse odottaa.

McDonaldin ja Thomasin (2015, s. 103) mukaan IEC 103 on sarjaliikenteeseen perustuva protokolla, joka on kehitetty erityisesti sähköaseman sisäiseen liikenteeseen suojauslaitteiden tarpeisiin. Sen avulla voidaan muun muassa hoitaa katkaisijoiden tilatiedot, laukaisusignaalit ja jälleenkytkennät. Toiminta perustuu laitteiden välisiin master-slave-suhteisiin, tiedonsiirto tapahtuu masterin aloitteesta. Fyysisen kerroksen voi muodostaa RS-232 tai RS-485, minkä lisäksi tiedonsiirrossa voidaan hyödyntää valokuitua.

IEC 104 on edellä käsitellyn IEC 101 protokollan kehittyneempi versio. Se perustuu ether-
netiin ja sisältää myös TCP- ja IP-kerrokset. Nykyisin sitä hyödynnetään usein kaukokäy-
tössä ala-aseman ja valvomon välillä.

3.4 IEC 61850

IEC 61850 on nykyisin keskeinen standardi sähköasemien tiedonsiirrossa. Falkin (2019, s.
39) mukaan standardin ensimmäinen versio julkaistiin vuonna 2004. Ensimmäinen versio
oli suppea verrattuna noin kuusi vuotta myöhemmin julkaistuun toiseen versioon. Toinen
versio on edelleenkin käytössä, mutta siihen on tehty lisäyksiä ajan myötä. Mackiewicz
(2006) mukaan motiivina IEC 61850 standardin luomiselle oli muun muassa tarve saada
usean eri valmistajan laitteet kommunikoimaan keskenään vaivattomasti. Taulukossa 1
on esitetty standardin rakennetta, joka koostuu kymmenestä osasta, joista käsitellään
seuraavaksi tämän työn kannalta olennaisimmat.

Taulukko 1. IEC 61850 standardin rakenne (Mackiewicz, 2006).

Luku	Aihe
1	Introduction and overview
2	Glossary and terms
3	General requirements
4	System and project management
5	Communication requirements for functions and device models
6	Configuration description language for communication in electrical substations related to IED's
7	Basic communication structure for substation and feeder equipment
7.1	- Principles and models
7.2	- Abstract communication service interface (ACSI)
7.3	- Common data classes (CDC)
7.4	- Compatible logical node classes and data classes
8	Specific communication service mapping (SCSM)

8.1	- Mapping to MMS(ISO/IEC 9506 – Part 1 and part 2) and to ISO/IEC 8802-3
9	Specific communication service mapping (SCSM)
9.1	- Sampled values over serial unidirectional multidrop point to point link
9.2	- Sampled values over ISO/IEC 8802-3
10	Conformance testing

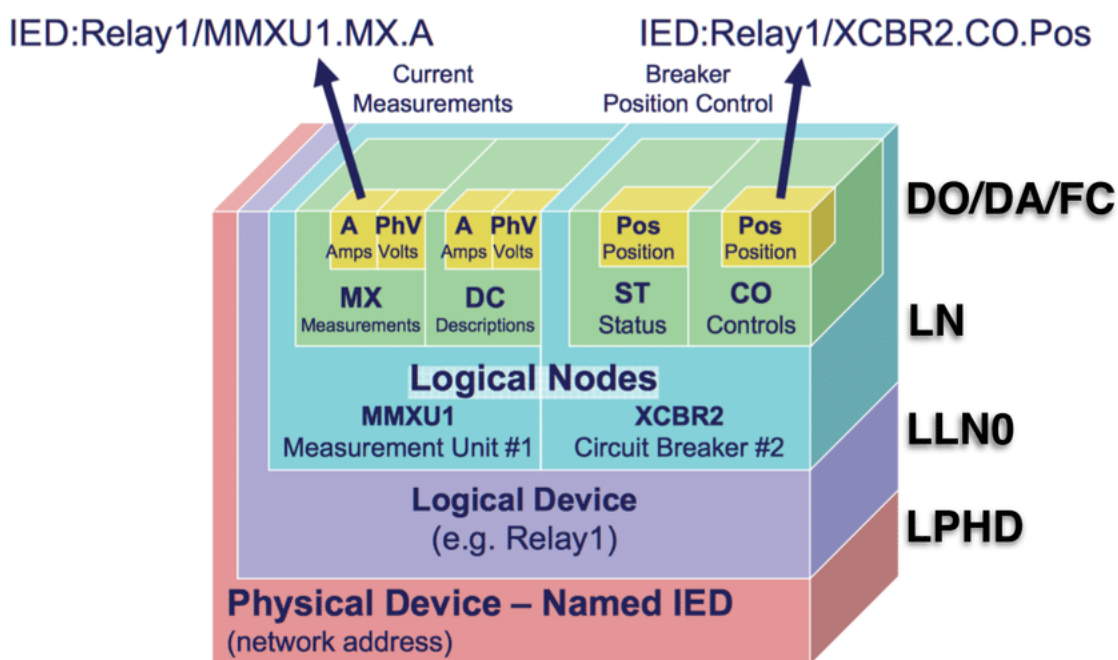
3.4.1 Abstract communication service interface (ACSI)

Standardin osiossa 7.2 käsitellään abstract communication service interface (ACSI), joka on yksi IEC 61850 standardin keskeisimmistä ominaisuuksista. Se toimii rajapintana sähköjärjestelmän komponenttien sekä analogisten arvojen ja asema-automaation välillä. Se määrittää loogisesti erilaiset palvelut sekä virtualisoi primääritason laitteet. ACSI sisältää muun muassa datamallin sekä sen eri luokkia käyttävät palvelut (Liang, 2008). Kostic (2007) mukaan jokainen IEC 61850 standardia tukeva suojarle sisältää ACSI-palvelimen, jonka avulla suojarle voi toteuttaa eri palveluita kuten esimerkiksi raportoida data-attribuuttien arvoja väylään sekä toteuttaa väylästä tulevia ohjauksikäskyjä. ACSI-palvelinten ansiosta lähetettävä laite ei tarvitse tietoja vastaanottavasta laitteesta, mikä mahdollistaa valmistajasta riippumattoman yhteensopivuuden laitteiden välillä. ACSI palvelin takaa ikään kuin yhteisen kielen laitteille, mutta se ei kuitenkaan vastaa tiedonsiirrosta laitteiden välillä vaan siitä vastaavat standardissa erikseen määritellyt protokollat.

ACSI-palveluiden keskiössä on datamalli. Mackiewicz (2006) mukaan datamallin määrittely alkaa fyysisestä laitteesta (physical device). Fyysisellä laitteella tarkoitetaan laitetta, joka liittyy tietoliikenneverkkoon. Se määritellään yleensä sen IP-osoitteen perusteella. Fyysinen laite voi sisältää yhden tai useamman loogisen laitteen (logical device). Kuvan 7 esimerkissä fyysinen laite sisältää vain yhden loogisen laitteen suojarle 1. Loogiset laitteet sisältävät loogisia solmuja (LN), jotka kuvaavat niiden toiminnallisuuksia. Esimerkiksi kuvassa solmu XCBR tarkoittaa katkaisijaa. X ilmaisee, että solmu liittyy kytkinlaitteistoon ja CBR on standardin määrittelemä lyhenne katkaisijalle (engl. circuit breaker).

Loogiset solmut puolestaan sisältävät dataobjekteja (DO), jotka kertovat minkä tyyppi-
sestä tiedosta on kyse.

Esimerkin katkaisijan dataobjektina on position (Pos), joka liittyy katkaisijan tilatietoon. Dataobjekti pitää sisällään data-attribuutit (DA), jotka sisältävät itse datan. Esimerkiksi Pos dataobjekti pitää sisällään vähintään data-attribuutit stVal eli tila, q eli viestin laatu-
luokitus sekä aikaleiman t. Malliin on merkitty myös kunkin dataobjektin toiminnallinen rajoite eli function constraint (FC). Toiminnallinen rajoite status (ST) tarkoittaa, että ky-
seisen data liittyy tilan raportointiin. Vastaavasti toiminnallinen rajoite controls (CO) tar-
koittaa, että kyseinen data liittyy ohjauskomentoihin. Dataobjektien ja -attribuuttien ominaisuuksia käsitellään seuraavaksi tarkemmin.



Kuva 7. IEC 61850 datamalli (Tebekaemi, 2016).

Kuvan 8 ensimmäisessä sarakkeessa nähdään katkaisijan loogisen solmun XCBR sisältä-
miä dataobjekteja. Toisesta sarakkeesta nähdään standardin määrittämä yleinen data-
luokka common data class (CDC), joka kuvaa dataobjektin tietojen tyyppin ja rakenteen.
Kolmannesta sarakkeesta nähdään objektin kuvaus ja viimeisestä sarakkeesta onko

objekti pakollinen (M) vai valinnainen (O). Tämän perusteella pakollisia tietoja katkaisijan loogiselle solmulle ovat muun muassa katkaisijan asentotieto, kiinni- ja aukiohjauksen esto, toimintakyky sekä tapahtuuko ohjaus paikallisesti vai etänä.

XCBR class				
Attribute Name	Attr. Type	Explanation	T	M/O
LNName		Shall be inherited from Logical-Node Class (see IEC 61850-7-2)		
Data				
Common Logical Node Information				
		LN shall inherit all Mandatory Data from Common Logical Node Class		M
Loc	SPS	Local operation (local means without substation automation communication, hardwired direct control)		M
EEHealth	INS	External equipment health		O
EEName	DPL	External equipment name plate		O
OpCnt	INS	Operation counter		M
Controls				
Pos	DPC	Switch position		M
BlkOpn	SPC	Block opening		M
BlkCls	SPC	Block closing		M
ChaMotEna	SPC	Charger motor enabled		O
Metered Values				
SumSwARs	BCR	Sum of Switched Amperes, resetable		O
Status Information				
CBOpCap	INS	Circuit breaker operating capability		M
POWCap	INS	Point On Wave switching capability		O
MaxOpCap	INS	Circuit breaker operating capability when fully charged		O

↑
↑
↑

Data Name Common Data Class Mandatory/Optional

Kuva 8. Katkaisijan looginen solmu XCBR (Mackiewicz, 2006).

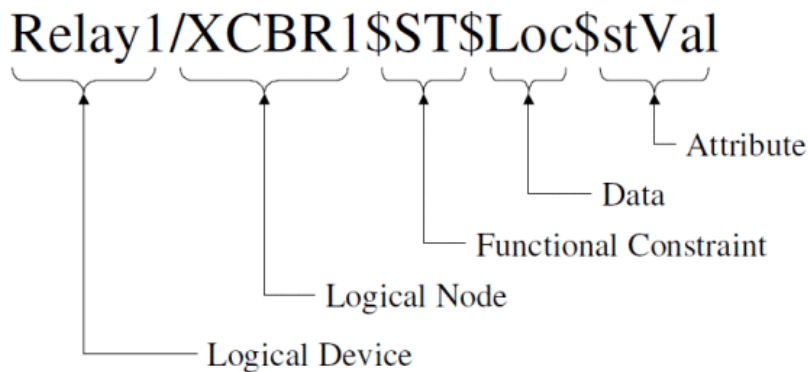
Käsitellään tarkemmin pakollista dataobjektia local operation (Loc). Se ilmaisee, tapahtuuko ohjaus paikallisesti vai etänä ja sen yleinen dataluokka on single point status (SPS), mikä tarkoittaa, että tila voidaan ilmaista yhdellä bitillä. Dual point status (DPS) puolestaan vaatii kaksi bittiä tilan ilmaisuun. Kuvan 9 ensimmäisestä sarakkeesta nähdään sen sisältämät data-attribuutit ja toisesta sarakkeesta kunkin data-attribuutin tyyppi. Kolmannesta sarakkeesta nähdään function constraint (FC), joka määrittää data-attribuutin käyttötarkoituksen. Neljännessä sarakkeesta nähdään trigger option (TrgOp), joka määrittää milloin raportointi alkaa. Trigger option for data change (dchg) tarkoittaa, että raportointi alkaa, kun kyseisen attribuutin data muuttuu. Kuudennesta sarakkeesta nähdään, onko data-attribuutti pakollinen vai valinnainen.

SPS class					
Attribute Name	Attribute Type	FC	TrgOp	Value/Value Range	M/O/C
DataName	Inherited from Data Class (see IEC 61850-7-2)				
DataAttribute					
<i>status</i>					
stVal	BOOLEAN	ST	dchg	TRUE FALSE	M
q	Quality	ST	qchg		M
t	TimeStamp	ST			M
<i>substitution</i>					
subEna	BOOLEAN	SV			PICS_SUBST
subVal	BOOLEAN	SV		TRUE FALSE	PICS_SUBST
subQ	Quality	SV			PICS_SUBST
subID	VISIBLE STRING64	SV			PICS_SUBST
<i>configuration, description and extension</i>					
d	VISIBLE STRING255	DC		Text	O
dU	UNICODE STRING255	DC			O
cdcNs	VISIBLE STRING255	EX			AC_DLND_A_M
cdcName	VISIBLE STRING255	EX			AC_DLND_A_M
dataNs	VISIBLE STRING255	EX			AC_DLN_M

↑
Functional
Constraint
↑
Mandatory/
Optional

Kuva 9. Single point status ominaisuudet (Mackiewicz, 2006).

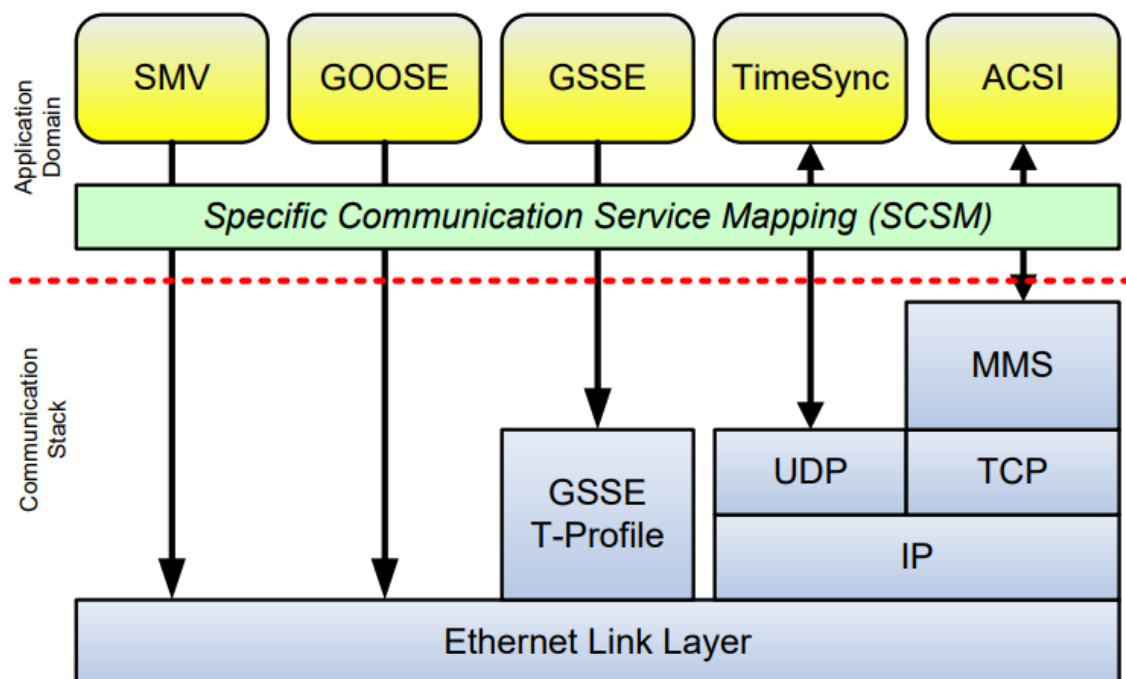
Kuvasta 10 nähdään vielä standardin mukainen viittaus datapisteeseen, jonka kaikki standardia tukevat laitteet ymmärtävät. Ozansoy (2009) mukaan näitä viittauksia ryhmäksi kokoamalla saadaan aikaan dataset. Niitä voidaan tehdä useita ja niiden avulla voidaan pyytää ja raportoida usean datapisteen arvo kerralla.



Kuva 10. IEC 61850 standardin mukainen viittaus datapisteeseen (Mackiewicz, 2006).

3.4.2 Manufacturing message specification (MMS)

Mekkasen (2015) mukaan Manufacturing message specification (MMS) on sovellustason protokolla, joka tarjoaa laitevalmistajasta riippumattoman tavan jakaa reaaliaikaista tietoa. Se toimii client-server-periaatteella mahdollistaen ACSI palvelinten toiminnan. Se mahdollistaa kommunikoinnin vertikaalisuunnassa kenttä- ja asematason laitteiden välillä. Usein palvelimena toimii esimerkiksi suojarele ja asiakkaana ala-asema. Keskeisenä ominaisuutena on virtual manufacturing device (VMD) -malli, joka määrittää miten palvelin nähdään asiakkaan näkökulmasta. Se sisältää omat objektit ja palvelut, jotka yhdistetään ACSI palveluihin. Mackiewicz (2006) mukaan MMS valittiin käyttöön siitä syystä, että se tukee monimutkaisia nimeämis- ja palvelumalleja. Suurin osa ACSI palveluista toteutetaan MMS palveluilla write ja read. Esimerkiksi, jos ala-asema haluaisi pyytää kenttäohjausyksiköltä katkaisijan tilatietoa, niin se etenisi seuraavalla tavalla. MMS käyttää ReadRequest -palvelua viitaten samalla katkaisijan dataobjektiin Relay1/XCBR1\$ST\$Pos\$stVal, minkä ACSI-palvelin yhdistää GetDataValues palveluun. Kenttäohjausyksikkö vastaa hyödyntäen MMS ReadResponse -palvelua. Standardin ideana kuitenkin on se, että tiedonsiirto olisi kyselyyn perustuvan tiedonkeruun sijaan tapahtumapohjaista raportointia. Kuten kuvasta 11 havaitaan niin MMS hyödyntää tiedonsiirrossa ethernetiä ja viestit reititetään IP-osoitteen perusteella.



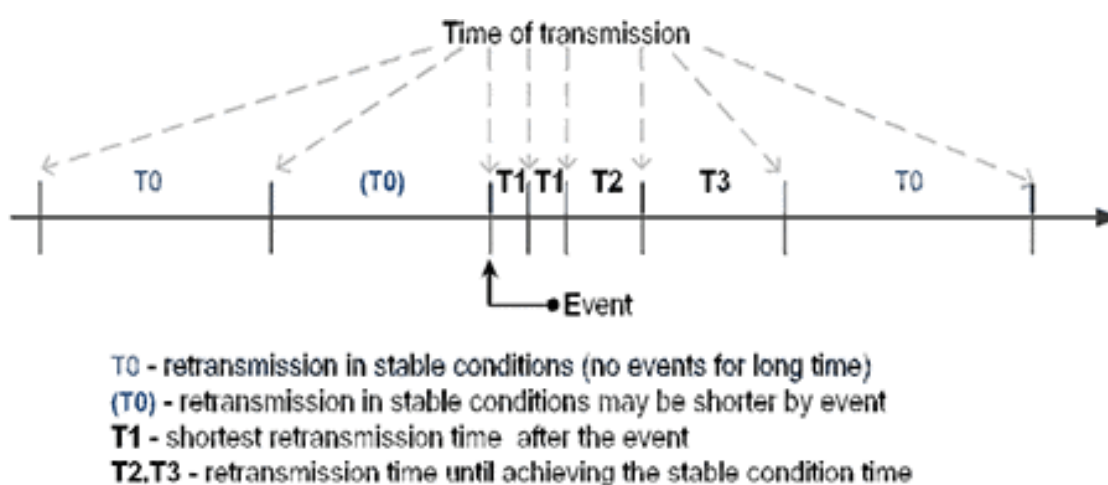
Kuva 11. Sovellustason protokollien yhteys alemman tason protokolliin (Liang, 2008).

3.4.3 Generic object oriented substation event (GOOSE)

Generic object oriented substation event (GOOSE) avulla voidaan toteuttaa sähköaseman horisontaalisessa suunnassa tapahtuva tiedonsiirto. Esimerkiksi asemaväylään kytkeytyt suojarleet voivat kommunikoida keskenään GOOSE:n avulla. Lopesin (2015) mukaan GOOSE on aikakriittinen viesti, jolla on tärkeä rooli sähköaseman suojaus- ja ohjaustoiminnoissa. Viestien tärkeyden vuoksi ne täytyy toimittaa nopeasti ja luotettavasti. Yksi toimintavarmuutta lisäävä tekijä on viestin uudelleenlähetystoiminto.

GOOSE toimii publisher-subscriber periaatteella, jossa lähettävä laite on julkaisija ja vastaanottava laite tilaaja. Kuvasta 11 nähdään kuinka GOOSE on yhdistetty suoraan OSI-mallin linkkikerrokseen, jolloin ylimääräisistä kerroksista ei aiheudu viivettä. Vastaanottajan IP-osoitteen sijaan kehykset lähetetään MAC monilähetysosoitteeseen, jonka muut laitteet voivat tilata tarpeen mukaan. Tämän myötä kuitenkin viestien toimitusvarmuus saattaa kärsiä, minkä parantamiseksi on kehitetty erilaisia toimintoja kuten uudelleenlähetystoiminto. Kuvassa 12 havainnollistetaan GOOSE:n uudelleenlähetystoiminnon

lähetyssykliä. Tapahtuman kohdalla sykli tihenee ja uudelleenlähetykset tapahtuvat ajanjakson T1 välein. Uudelleenlähetyksen ansiosta yhden kehyksen katoaminen ei vielä aiheuta ongelmia, mikäli seuraava kehys menee perille. Hiljalleen sykli harvenee ajanjakson T2 ja T3 pituiseksi, kunnes se palautuu normaalitilan T0 pituiseksi. Normaalitilassa GOOSE lähettää tasaisin väliajoin viestin, jolloin voidaan varmistua siitä, että yhteys toimii. Kukin viesti sisältää TimeAllowedToLive-arvon, joka ilmaisee tilaajalaitteille missä ajassa seuraavan viestin tulisi saapua. Jos seuraava viesti ei saavu aikaikkunan sisällä niin tilaajalaitte voi aktivoida hälytyksen yhteysvirheen merkiksi.



Kuva 12. Goose:n uudelleenlähetysoiminto (Lopes, 2015).

Kuvassa 13 nähdään GOOSE-kehysten rakenne. Lopesin (2015) mukaan kehys koostuu neljästä pääosista, jotka ovat ethernet, 802.1Q, ethertype ja GOOSE. Ensimmäinen osio koostuu viestin lähettäjän ja määränpään MAC-osoitteista. Osoitteen kolme ensimmäistä oktetia tulee olla 01-0C-CD ilmaistakseen, että kyseessä on MAC monilähetysosoite ja neljännen oktetin tulee olla 01 ilmaistakseen, että kyseessä on GOOSE-viesti. Loput kaksi oktetia osoitteesta voi määrittellä vapaasti väliltä 00-00 ja 01-FF. Kehyksen toinen osa määrittelee viestien ja verkkojen priorisoinnin. TPID arvo on 0x8100, mikä osoittaa, että paketti sisältää VLAN tunnisteita. PCP ilmaisee datan prioriteetin 802.1Q määrittämän asteikon mukaisesti, jossa 0 on pienin ja 7 suurin prioriteetti. CFI ilmaisee, onko osoite kanonisessa muodossa, mutta se on nykyisin korvattu tunnisteella DEI, joka ilmaisee, voidaanko paketti jättää toimittamatta ruuhkatilanteessa (Huawei, 2024). VID

ilmaisee mihin VLAN-verkkoon kyseinen paketti kuuluu ja sen käyttö on valinnaista. Ethernet-kehys ilmaisee mitä protokollaa kehyksen data hyödyntää sovelluserroksessa, joka on GOOSE:n tapauksessa 88b8 heksadesimaalimuodossa esitettyinä. Neljännen osion APPID auttaa vastaanottavia laitteita tunnistamaan, mitkä viestit kuuluvat niille. Osiossa on varattu myös tuleville sovelluksille kaksi paikkaa. Viimeisenä on gose protocol data unit (goosePDU), joka sisältää useita tietokenttiä.

Ethernet		802.1Q				Ethertype	GOOSE				
MAC dest	MAC src	TPID	PCP	CFI	VID	Type	APPID	length	Reserved 1	Reserved 2	goosePDU

Kuva 13. Goose-viestin ethernet-kehiksen rakenne (Lopes, 2015).

Goosepdu sisältämät tietokentät nähdään kuvasta 14. Falkin (2018, s. 184) mukaan control block reference on viittaus toimintalohkoon, josta viesti on peräisin. Time allowed to live määrittää ajan minkä kuluessa seuraavan viestin tulisi saapua yhteyden toimiessa. DataSetReference kertoo minkä tietojoukon dataa on toimitettu viestissä. GOOSEID on käyttäjän konfiguroitavissa oleva merkkijono, jonka avulla voidaan helpottaa kyseisen viestin suodattamista muiden joukosta. Jos sitä ei konfiguroida erikseen niin oletusarvona se on sama kuin control block reference. Viestin aikaleima käy ilmi kohdasta event timestamp. State number ilmaisee viestin datassa tapahtuvat muutokset kasvattaen lukuarvoaan aina yhdellä, kun viestin datassa tapahtuu muutos. Sequence number puolestaan ilmaisee, kuinka mones viesti on kyseessä viimeisimmän datassa tapahtuneen muutoksen jälkeen. Se siis kasvaa yhdellä joka viestin kohdalla, kunnes state number muuttuu, jolloin sequence number alkaa taas kasvamaan nolasta. Simulation bit tai test kertoo, onko kyseinen viesti testi. Configuration revision ilmaisee tietojoukon konfigurointiversion ja arvo kasvaa aina kun tietojoukkoa muokataan. Needs commissioning antoi ensimmäisessä versiossa arvon tosi, kun julkaisija oli havainnut merkittävän ongelman, mutta toisessa versiossa vastaavassa tilanteessa julkaisija lopettaa viestien lähettämisen. Number dataset entries kertoo viestin sisältämän datapisteiden määrän. Lopuksi viestin kohdasta data löytyy itse viestin data.

```

v goose
  IEC GOOSE
  {
    Control Block Reference*: LD1/LLN0.CB22_Status-R-GOOSE
    Time Allowed to Live (msec): 20
    DataSetReference*: LD1/LLN0.StatusofCB22
    GOOSEID*: LD1/LLN0.CB22_Status-R-GOOSE
    Event Timestamp: 2022-06-16 16:13.24.096393 Timequality: 0a
    StateNumber*: 1
    SequenceNumber*: 0
    Simulation Bit FALSE
    Config Revision*: 1
    Needs Commissioning*: FALSE
    Number Dataset Entries: 1
    Data
    {
      BOOLEAN: TRUE
    }
  }

```

Kuva 14. Goose protocol data unit sisältö (Hussain, 2023).

3.4.4 Prosessiväylä ja sampled values

Perinteisten sähköasemien lisäksi voidaan toteuttaa myös digitaalisia sähköasemia, joissa hyödynnetään IEC 61850 standardin määrittämää prosessiväylää sekä sampled value (SV) viestejä. Prosessiväylällä tarkoitetaan kenttä ja prosessitason välistä ethernetiin perustuvaa tiedonsiirtoväylää, joka havaittiin myös kuvasta 2 sivulla 16. Perinteisesti kaikki mittamuuntajien mittaustiedot sekä kytkinlaitteiden ohjaukset ja tilat on johdettu yksittäin jakokaapeilta asemarakennuksen relekaapeille, minkä välimatka on kymmenistä metreistä satoihin metreihin. Prosessiväylän myötä johdotus vähenee huomattavasti, sillä asemalla saattaa olla kymmeniä kenttiä, joista kultakin kentältä kulkee kymmeniä johtoja asemarakennuksen relekaapeille.

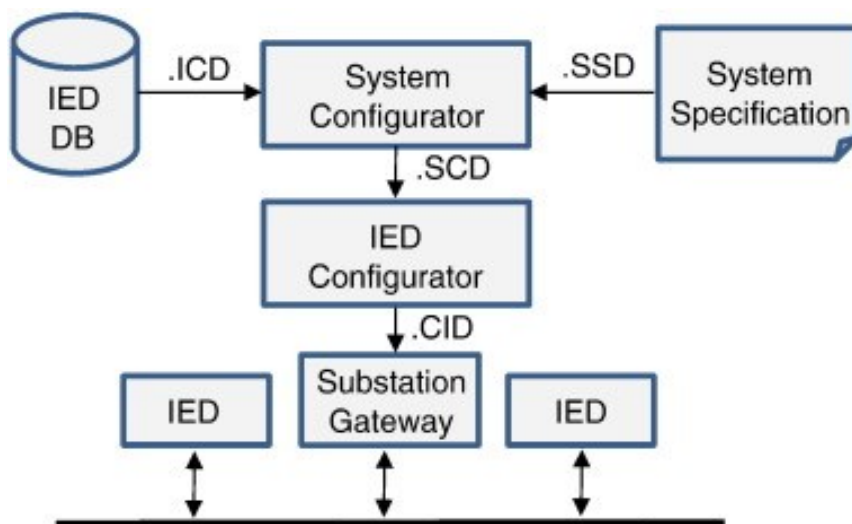
Mekkasen (2015, s. 36) mukaan mittamuuntajien mittausdata toimitetaan prosessiväylässä SV viesteinä kenttätason laitteille. SV on Goosen tavoin aikakriittinen viesti ja muistuttaa myös muilta osin paljon Goose:a sillä se toimii publisher-subscriber periaatteella. Kuten kuvasta 11 nähdään, niin SV on yhdistetty GOOSE:n tavoin suoraan OSI-mallin linkkikerrokseen tiedonsiirron nopeuttamiseksi.

Adewole (2014) mukaan merging unit (MU) toimii rajapintana perinteisten analogisten mittasignaalien ja prosessiväylän digitaalisten viestien eli SV välillä. Lisäksi sama laite voi toimia rajapintana primäärilaitteiden binääriarvojen ja prosessiväylän GOOSE sanomien välillä. Esimerkiksi Siemens 6MU85 merging unit sisältää mittausten virta- ja jänniteliihtäntöjen lisäksi binäärisiä lähtöjä kytkinlaitteiden ohjaamista varten sekä binäärisiä tuloja kytkinlaitteiden tilatietojen keräämistä varten. Näiden binääristen tietojen lähetys väylään tapahtuu GOOSE-viesteillä. Pelkästään mittaussignaalien muuntamiseen tarkoitettusta laitteesta voidaan käyttää nimeä stand alone merging unit (SAMU). MU voi olla myös sulautettu low power instrument transformer (LPIT) laitteen kanssa yhdeksi laitteeksi. Toiminta perustuu siihen, että MU ottaa signaalista näytteitä tietyllä taajuudella, jonka jälkeen paketoit ne ethernet-kehysiin. Yksi viesti voi sisältää useamman näytteen. IEC 61850-92LE määrittelee näytteenottotaajuudeksi normaaleihin suojaustoimintoihin 80 näytettä jaksonajalta ja korkeamman taajuuden vaativiin sovelluksiin kuten sähkölaadun tarkkailuun 256 näytettä jaksonajalta. 50 herzin järjestelmässä näytteenottotaajuudet ovat siis 4 kHz ja 12,8 kHz. Suuren näytteenottotaajuuden vuoksi on syytä kiinnittää huomiota prosessiväylän tiedonsiirtokapasiteetin riittävyyteen jo suunnitteluvaiheessa. SV viesteissä myös aikasykronointi on merkittävässä roolissa, sillä vääristynyt aikaviite laitteiden välillä aiheuttaa vaihesiirtymää mittaustuloksiin.

3.4.5 Konfigurointikieli ja tiedostotyypit

Standardin osiossa 6 määritellään konfigurointikuvauksen kieli eli Substation configuration language (SCL). Peng (2016) mukaan SCL-kieli perustuu XML-kieleen ja sillä voidaan kuvata laitteiden toimintoja, sekä sähköaseman ja sen tietoliikenneverkon rakenteita. Kuvasta 15 nähdään konfiguroinnin keskeisimmät tiedostomuodot. IED capability description (ICD) -tiedosto kuvaa laitteen ominaisuudet kuten mitä loogisia solmuja, toimintoja ja viestintäprotokollia laite tukee. Kunkin laitteen ICD siirretään system configurator -työkaluun. System specification description (SSD) -tiedostolla kuvataan sähköaseman rakenne single line diagram (SLD) muodossa eli yksiviivaesityksenä. Se myös kuvaa mitä loogisia solmuja järjestelmän tulee sisältää ja miten järjestelmän komponentit ovat yhteydessä toisiinsa. Kun ICD ja SSD on viety system configurator -työkaluun niin saadaan

muodostettua substation configuration description (SCD) -tiedosto, joka sisältää aseman täydellisen konfiguraation. Siinä yhdistyvät SSD-tiedoston ja ICD-tiedostojen tiedot. Configured IED description (CID) -tiedosto kuvaa SCD-tiedoston tiedot vain yhden laitteen osalta.



Kuva 15. Konfiguroinnin periaate (Peng, 2016).

3.5 Aikatahdistus

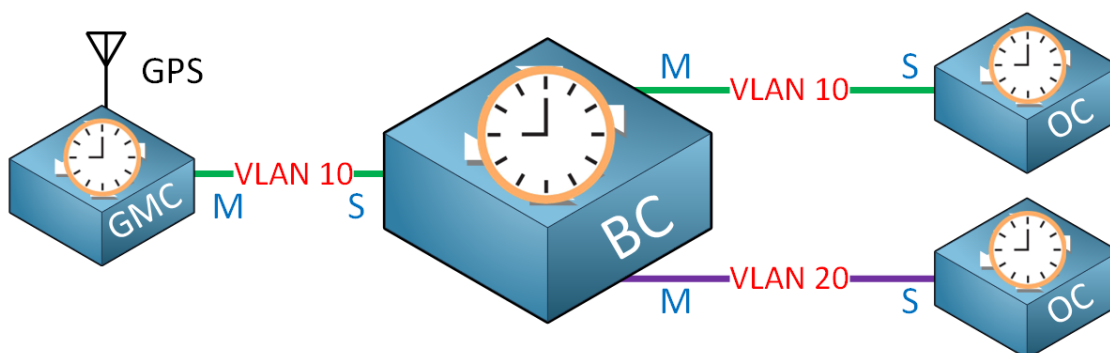
Aikatahdistus on keskeinen osa sähköaseman toimintaa ja sen rooli korostuu entisestään digitaalisella sähköasemalla. Ozanzoy (2008) mukaan laitteiden aikatahditusvaatimukset kellolle on jaettu viiteen luokkaan IEC 61850 standardissa. Luokat ja niiden tarkkuudet ovat $T1=\pm 1$ ms, $T2=\pm 100$ μ s, $T3=\pm 25$ μ s, $T4=\pm 4$ μ s ja $T5=\pm 1$ μ s. Aikasykronoinnin tarkkuus riippuu käytetystä protokollasta. Tyypillisiä protokollia ovat network time protocol (NTP), simple network time protocol (SNTP) ja precision time protocol (PTP).

3.5.1 Precision time protocol (PTP)

Edellä mainituista tarkin on IEEE 1588 standardin määrittelemä PTP, joka riittää täyttämään myös prosessiväylän aikatahditusvaatimukset mikrosekuntien aikatahditustarkkuudella. Moore (2010) mukaan IEEE 1588 määrittelee järjestelmän neljä kellotyyppiä, jotka ovat ordinary, grandmaster, boundary sekä transparent. Laitteiden välillä vallitsevat roolit

master ja slave sen mukaan onko laite aikälähde vai vastaanottaja. Tyypillisesti GPS:n avulla synkronoitu grandmaster toimii asemalla ensisijaisena aikälähteenä. Vastaavia laitteita voi olla asemalla kaksi redundanssin parantamiseksi, mutta ne voivat toimia vain yksi kerrallaan grandmasterina. Best master clock -algoritmin perusteella valitaan, kumpi toimii grandmasterina.

Kuvassa 16 havainnollistetaan laitteiden keskinäisiä rooleja. Grandmaster lähettää viestin esimerkiksi kytkimelle, jonka tyyppi on transparent. Tällöin kytkin ei synkronoi kelloa, mutta mittaa aiheuttamansa viiveen ja välittää viestin eteenpäin. Mikäli kyseessä on tyyppi boundary, niin laite synkronoi kellonsa ja lähettää ajan viestillä eteenpäin. On hyvä huomata, että boundary voi jakaa aikaa eri VLAN-verkkoihin, mutta transparent voi välittää ajan vain samaan verkkoon. Viestejä pelkäästään vastaanottavan laitteen kuten esimerkiksi IED:n tyyppi on ordinary.



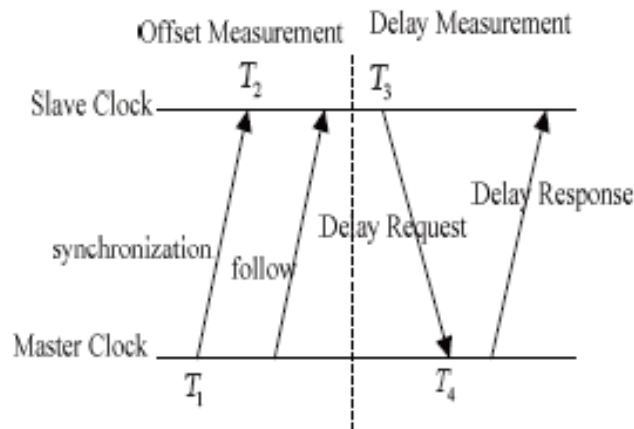
Kuva 16. PTP-protokollan toimintaperiaate (Networklessons.com, n.d.).

Wang (2010) mukaan kuvasta 17 nähdään PTP aikasynkronoinnin periaate kahden laitteen välillä. Todellinen kellonaika sisällytetään synkronointiviestiin, jonka grandmaster lähettää monilähetyksenä muille laitteille. Joissain tapauksissa synkronointiviestiin ei saada aikaleimaa T_1 viestin lähetyksestä, minkä vuoksi se lähetetään follow-viestillä. Aikaleima T_2 saadaan synkronointiviestin perille saapumisesta. Aikaleima T_3 saadaan delay request -viestin lähetyksestä ja T_4 sen perille saapumisesta. Kun aikaleimat ovat tiedossa niin ero lähteen ja vastaanottavan laitteen kellojen välillä voidaan laskea yhtälöstä

$$T_{offset} = \frac{(T_2 - T_1) - (T_4 - T_3)}{2} \quad (1)$$

Lähetyksestä aiheutunut viive voidaan puolestaan laskea yhtälöstä

$$T_{delay} = \frac{(T_2 - T_1) + (T_4 - T_3)}{2} \quad (2)$$



Kuva 17. PTP-protokollan aikaleimat (Wang, 2010).

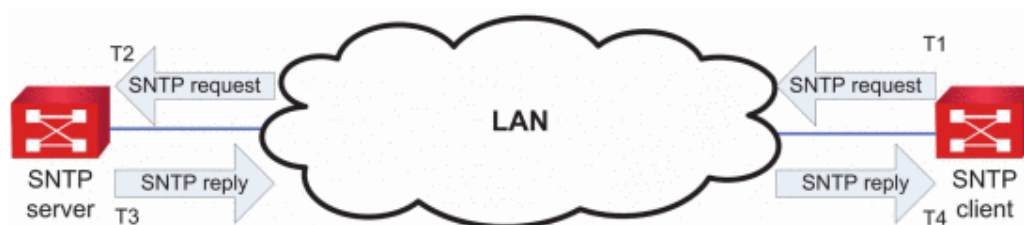
3.5.2 Network time protocol (NTP)

Neagoe (2006) mukaan NTP on yksi suosituimmista internetin aikasynkronointiprotokollista. Sitä voidaan hyödyntää myös sähköaseman aikasynkronoinnissa, vaikka sen millisekuntien luokkaa oleva aikatarkkuus onkin huonompi kuin PTP:n. NTP toimii hierarkkisesti stratum-tasoilla 0–15, joista ylin taso nolla edustaa tarkinta ajanlähdettä, kuten GPS-vastaanotinta. Kukin taso jakaa ajan alapuolella olevalle tasolle. NTP:n aikasynkronointi perustuu SNTP:n tavoin kyselypohjaiseen malliin, jossa asiakas pyytää kellonaikaa aikapalvelimelta. Kysely tapahtuu samalla periaatteella kummallakin protokollalla ja se on kuvattu tarkemmin SNTP:n yhteydessä seuraavassa luvussa. NTP protokollaa käytettäessä asiakaslaite hyödyntää risteysalgoritmia, jonka avulla se voi vertailla usean eri palvelimen tarjoamaa aikaa ja valita tarkimman, mikä vähentää virheellisen synkronoinnin riskiä. Synkronoinnin tarkkuutta kuitenkin heikentää se, että viiveiden oletetaan olevan

samat molempiin suuntiin, mikä ei aina pidä paikkaansa. Epätarkkuutta lisää myös se, että viestien aikaleimat luodaan sovellustasolla, jolloin viiveet laitteen sisäisessä käsitte-lyssä voivat aiheuttaa vääristymää.

3.5.3 Simple network time protocol (SNTP)

SNTP on kevennetty versio NTP protokollasta ja mahdollistaa aikasynkronoinnin vähintään kymmenien millisekuntien tarkkuudella. Ussoli (2013) mukaan SNTP aikasynkronointi tapahtuu NTP:n tavoin kyselypohjaisesti, mitä havainnollistetaan kuvassa 18. Asiakslaitte lähettää aikapalvelimelle kyselyviestin, joka sisältää lähetyshetken aikaleiman $T1$. Viestin saapuessa palvelimelle palvelin kirjaa ylös saapumisajan $T2$. Tämän jälkeen palvelin muodostaa vastausviestin, joka sisältää palvelimen ajan sekä aikaleimat $T2$ ja $T3$, joka saadaan vastausviestin lähetyksestä. Vastausviestin saapumisesta asiakslaitteelle saadaan vielä aikaleima $T4$, minkä jälkeen asiakslaitte voi laskea viestin viiveen sekä kellonsa poikkeaman ja säätää ajan. Yhtenä protokollan rajoitteena on se, että synkronointi voidaan toteuttaa tiheimmillään 16 sekunnin välein. SNTP ei myöskään mahdollista kellon käyntinopeuden muuttamista aikaa synkronoidessa kuten NTP, vaan säätö tapahtuu siirtämällä kello oikeaan aikaan yhdellä askeleella, mikä saattaa aiheuttaa ongelmia joissain järjestelmissä. SNTP eroaa NTP:stä myös siten, että se ei sisällä vertailualgoritmia, minkä takia asiakslaitteet eivät voi arvioida eri aikapalvelinten tarkkuutta. Asiakslaitte kykenee vastaanottamaan ajan vain yhdeltä palvelimelta ja ajan ollessa virheellinen myös asiakkaan kellonaika vääristyy.



Kuva 18. SNTP protokollan kommunikointi (Ussoli, 2013).

4 Konfigurointi ja testaus

Työn yhtenä tavoitteena on selvittää, kuinka käytönvalvontajärjestelmän käyttöliittymän konfigurointi tapahtuu Zenon-ohjelmiston avulla. Lisäksi tavoitteena on selvittää, miten Omicron-laitteistoa voi hyödyntää automaation testauksessa. Tässä luvussa perehdytään käyttöliittymän konfiguroinnin työvaiheisiin sekä järjestelmän testaamiseen.

4.1 Työkalut ja työn tavoitteet

Käyttöliittymän konfigurointiin käytetään Copa-Datan kehittämää Zenon automaatio-ohjelmistoa. Zenon tarjoaa monipuolisia ratkaisuja teollisuuden sekä energijärjestelmien valvonta- ja ohjaustarpeisiin. Tämän työn konfiguroinnit toteutetaan Zenon Engineering Studio 14 avulla. Zenon Service Engine on osa, joka suorittaa varsinaisen valvonta- ja ohjausjärjestelmän toiminnan.

Konfiguroitavan käytönvalvontajärjestelmän HMI tulee kattamaan kuusi 110 kV kenttää, mutta testiosiossa keskitytään vain yhden kentän testaamiseen, sillä muut kentät toimivat samalla periaatteella. Työmäärän rajaamiseksi konfiguroinnissa keskitytään ensisijaisesti järjestelmän toiminnallisuuden toteuttamiseen, minkä takia ohjelmiston visuaaliset toiminnot jätetään vähemmälle huomiolle. Keskeisimpinä tavoitteina konfiguroinnissa on luoda toimiva SLD-näkymä, ohjaustoiminnot, tapahtumalista ja hälytyslista.

Järjestelmän testaus suoritetaan kuvassa 19 näkyvän Omicron MBX1-laitteen sekä Omicron IEDScout-ohjelmiston avulla. Kyseistä laitetta voidaan hyödyntää myös Stationscout-ohjelmiston kanssa ja laitteesta kuuleekin usein käytettävän nimeä stationscout. MBX1 mahdollistaa IED-laitteiden simuloimisen, minkä ansiosta automaation testaaminen voidaan suorittaa ilman sähköaseman fyysisiä laitteita. IEDScoutin avulla voidaan ohjata muun muassa, mitä tilatietoja ja hälytyksiä MBX1 lähettää HMI:lle. Liittyminen laitteeseen tietokoneella tapahtuu laitteen takaportin kautta. Etuporttien kautta tapahtuu asema-automaation tietoliikennöinti.



Kuva 19. Omicron MBX1-laite (Omicron, 2021).

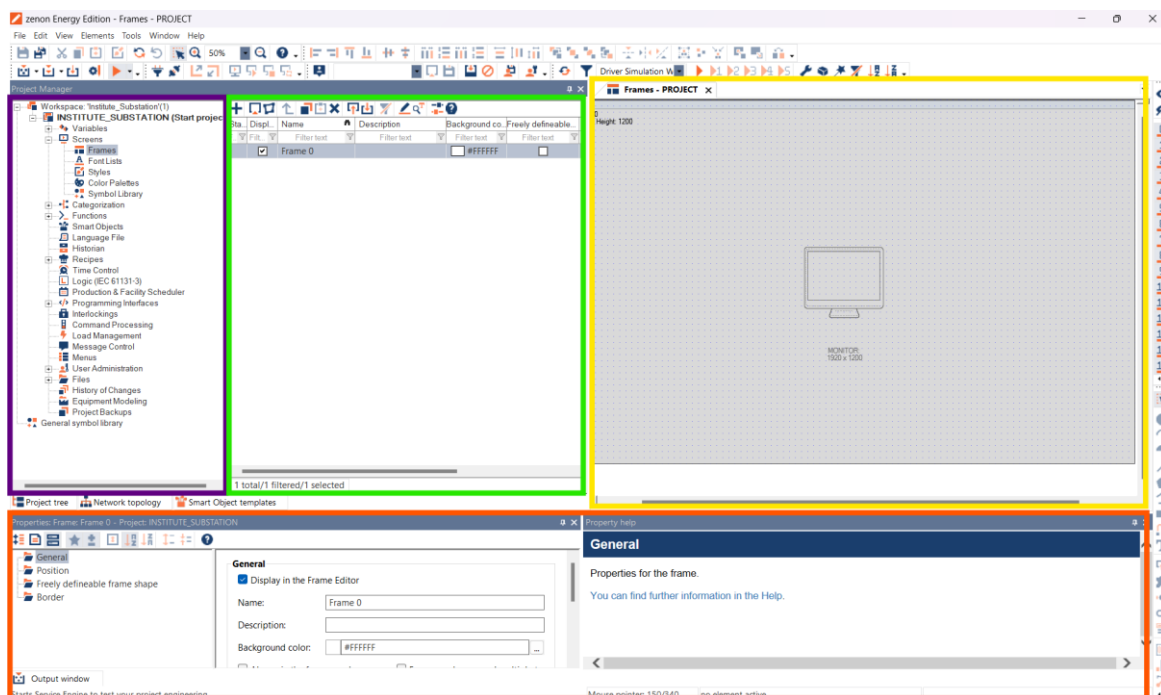
4.2 Konfigurointiprosessi

Seuraavissa osioissa käsitellään käytönvalvontajärjestelmän HMI:n konfigurointia Zenonilla. Ohjelmistossa on lukuisia eri toimintoja ja tässä käsitellään niistä tämän työn kannalta keskeisimmät. Monille asioille on myös useita vaihtoehtoisia toteutustapoja ja tässä on kuvattu vain joitain niistä. Konfigurointia varten sähköasemasta tarvitaan jonkin verran lähtötietoja. Tarvittavat tiedot saadaan hankkimalla IED-laitteiden CID-tiedostot tai aseman SCD-tiedosto. Lisäksi konfiguroinnissa voi olla tarpeen aseman IP-osoitelista ja yksiviivakaavio.

4.2.1 Projektin luominen

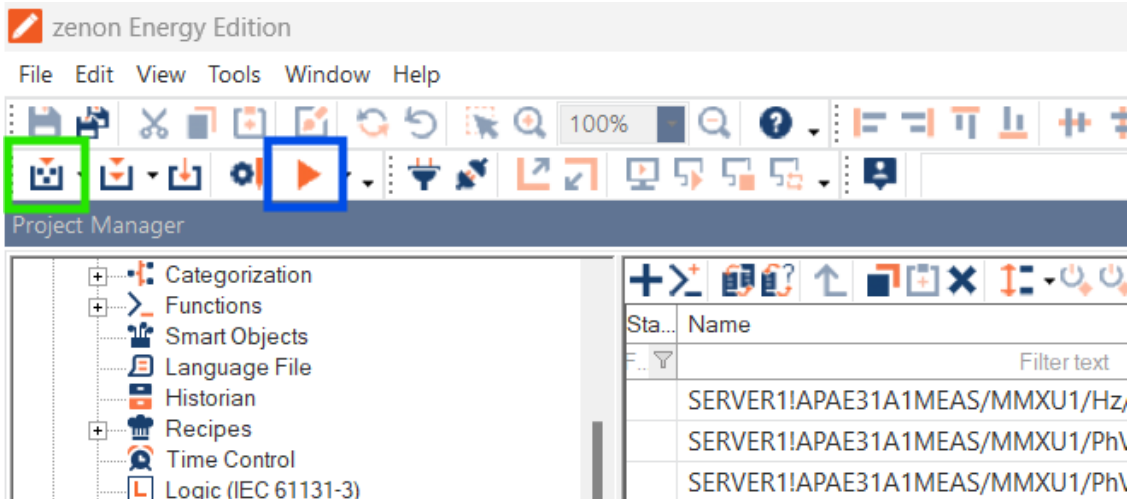
Uusi projekti aloitetaan työtilan luomisella klikkaamalla seuraavassa järjestyksessä *File*, *Workspace* ja *New Workspace*. Yksi työtila voi sisältää useita projekteja ja uusi projekti saadaan lisättyä klikkaamalla *File* ja *Project New*. Kuvassa 20 nähdään yleisnäkyminen ohjelmasta. Violetissa ikkunassa on projektipuu, jonka avulla voidaan hallita projektin ominaisuuksia ja työkaluja. Vihreään ikkunaan avautuu objektilista, jossa voidaan nähdä

erityyppiset objektit listattuna. Haluttu objektista voidaan avata projektipuusta, kuten kuvassa kehysten lista. Keltaisessa ikkunassa voidaan muokata objekteja, jotka saadaan avattua ikkunaan objektistasta. Oranssissa ikkunassa voidaan asettaa objektien parametrejä, mitä helpottaa kunkin parametrin kohdalla oikealle puolelle avautuva ohje.



Kuva 20. Zenon Engineering Studio yleisnäkymä.

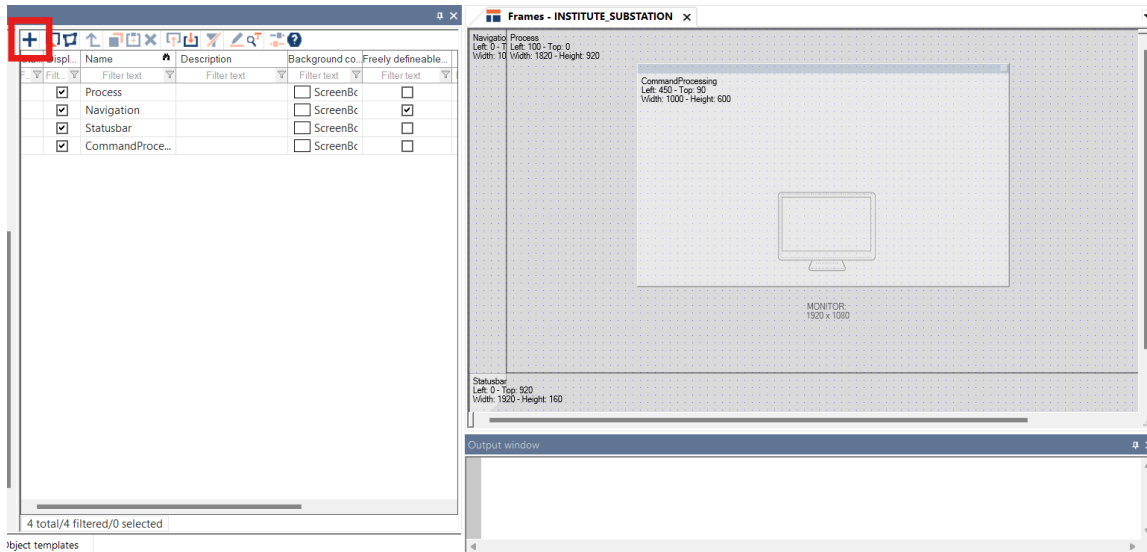
Konfiguroinnin jälkeen, muutokset täytyy aina ladata service engineen ja se tapahtuu kuvassa 21 esitetyn vihreän ikkunan painikkeesta. Sinisen ikkunan painikkeesta puolestaan käynnistetään service engine. Kun tiedot on ladattu service engineen niin se täytyy vielä päivittää, joka tapahtuu käynnistämällä service engine uudelleen tai funktiota hyväksi käyttäen. Työn helpottamiseksi kannattaa mahdollisimman alussa luoda painike ja yhdistää siihen funktio *reload project online*. Tämän painikkeen avulla saadaan päivitettyä service engineen nopeasti aina muutosten jälkeen. Painikkeiden ja funktioiden luominen käsitellään tarkemmin myöhemmissä kappaleissa.



Kuva 21. Service engineen muutosten lataaminen ja käynnistäminen.

4.2.2 Näyttöjen rakenteet

Rakenteen määrittely alkaa kehysten (Frames) luomisella. Kehykset ovat kiinteästi määriteltäviä alueita, joihin sijoitetaan myöhemmin luotavat näytöt (Screens). Kuvasta 22 nähdään, kuinka projektiin on luotu neljä kehystä vasemman yläkulman painikkeesta. Vastaavaa pluspainiketta käytetään myös muilla objektilistoilla, kun niille halutaan lisätä objekteja kuten esimerkiksi muuttujia. Kehyksen parametreissa olennaisinta on sen paikan määrittäminen. Oikealla olevasta ikkunasta nähdään kunkin kehysten paikka.

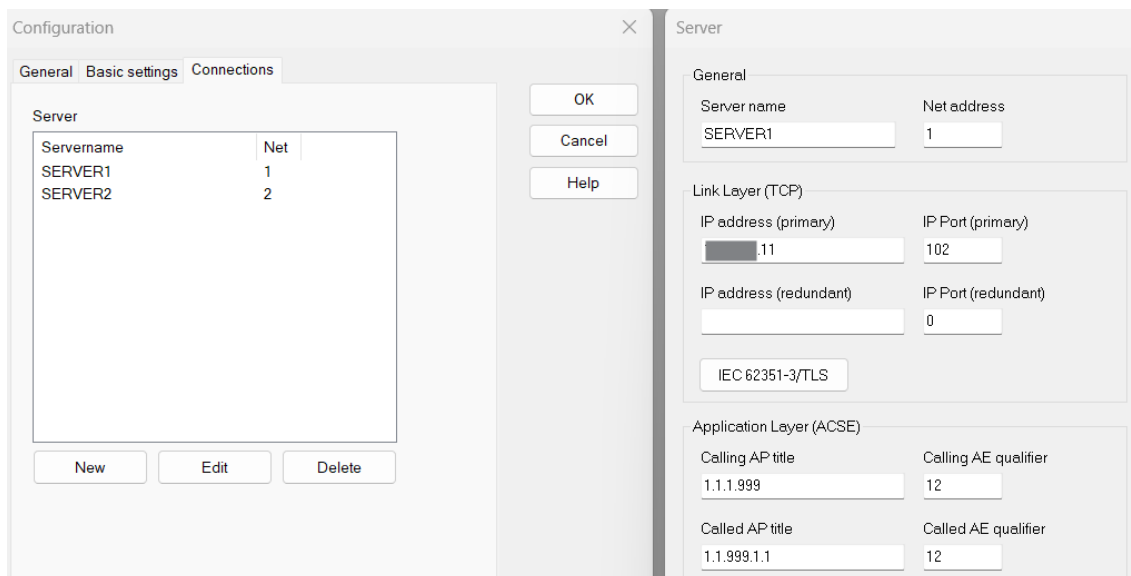


Kuva 22. Kehysten luominen.

Seuraavaksi voidaan luoda kehysten tavoin näyttöjä avaamalla projektipuusta näyttöjen objektilista ja klikkaamalla yläkulman pluspainikkeesta. Tämän myötä avautuvasta ikkunasta valitaan mihin kehukseen näyttö sijoitetaan sekä näytön tyyppi listasta, joka sisältää myös valmiiksi rakennettuja näyttöjä, joita hyödynnetään myöhemmin.

4.2.3 Ajurit

Ajurit (Drivers) mahdollistavat yhteyden IED-laitteisiin. Projektin ajureiden hallinta tapahtuu avaamalla projektipuusta kohta *Drivers*, joka löytyy kohdan *Variables* alapuolelta. Uuden ajurin lisääminen tapahtuu pluspainikkeesta. Seuraavaksi valitaan ajurin tyyppi, joka on tässä tapauksessa IEC 61850 driver. Seuraavaksi avautuu kuvassa 23 näkyvä *Configuration*-ikkuna, jonka kohdasta *Connections* päästään lisäämään uusi IED-laite. Tässä kohdassa tärkeintä on asettaa IED-laitteen IP-osoite. Tässä tapauksessa syötetään simuloitavan kenttäohjausyksikön IP-osoite. Ajurin luonnin jälkeen voidaan alkaa tuomaan kyseisen laitteen signaaleja eli muuttujia konfigurointiohjelmaan.

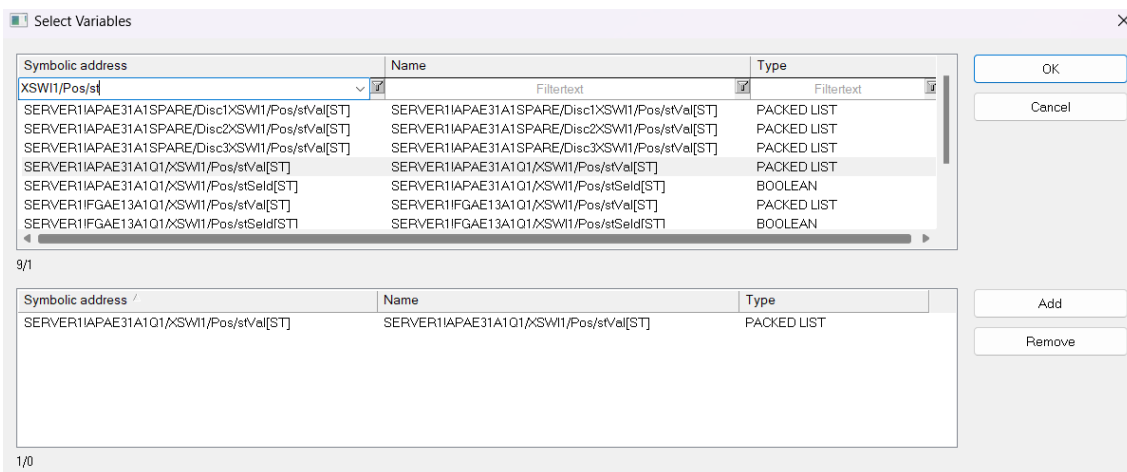


Kuva 23. IED-laitteen lisääminen.

4.2.4 Muuttujat

Muuttujat edustavat järjestelmässä signaaleja, joiden avulla eri toiminnot saadaan toteutettua ja tilat raportoitua. Uusien muuttujien luominen tapahtuu avaamalla muuttujalista projektipuusta kohdasta *Variables* ja klikkaamalla pluspainikkeesta. Seuraavaksi avautuvassa ikkunassa tulee määrittää muuttujan nimi, ajuri ja datan tyyppi.

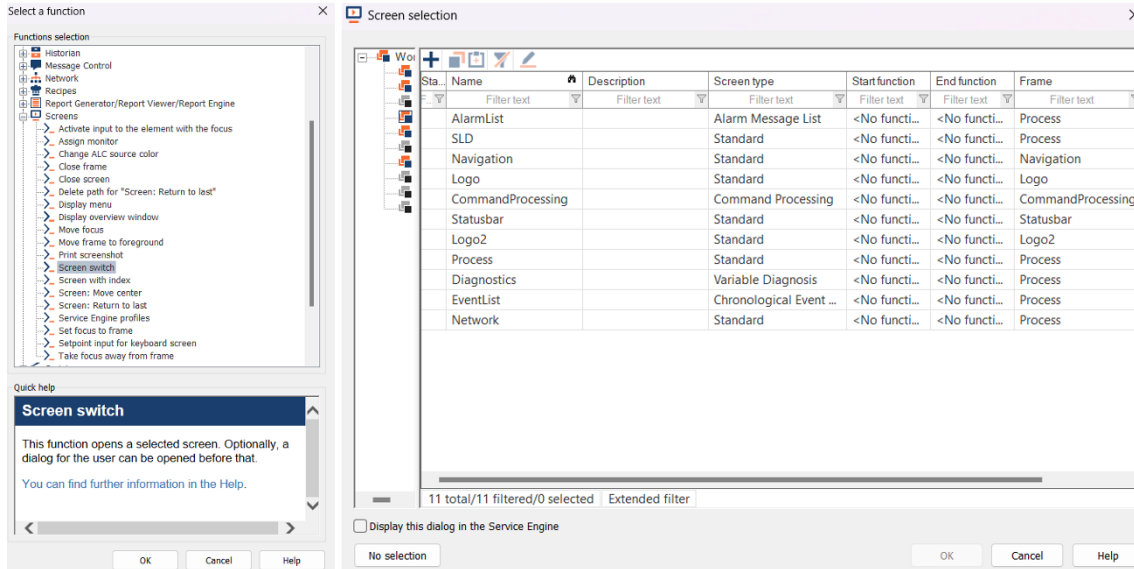
Muuttujia voidaan myös tuoda ulkopuolelta projektiin. IED-laitteen muuttujien tuonti tapahtuu helposti CID-tiedoston avulla. Valitaan ajurilistalta ajuri, jonka muuttujia halutaan tuoda ja klikataan yläpaneelissa olevaa alanuolipainiketta. Seuraavaksi avautuvasta ikkunasta valitaan laite, jonka muuttujia halutaan tuoda sekä tuodaanko muuttujat laitteesta vai tiedostosta. Tässä tapauksessa muuttujat tuodaan tiedosta ja kuvasta 24 voidaan nähdä esimerkkinä erottimen Q1 tilatietomuuttujan tuominen.



Kuva 24. Muuttujien tuominen CID-tiedostosta.

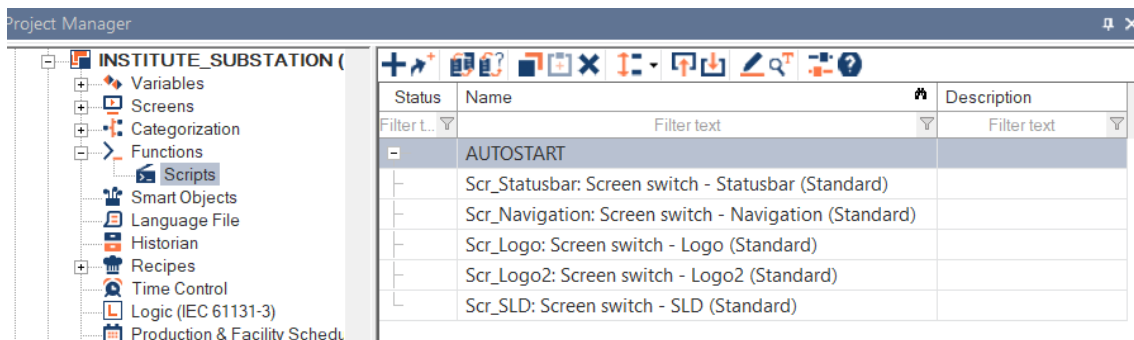
4.2.5 Funktiot

Funktioiden avulla voidaan toteuttaa toimintoja ja ensimmäisenä luodaan funktio, jolla saadaan suoritettua näytön vaihto. Projektipuun kohdasta *Functions* päästään tarkastelemaan ja luomaan uusia funktioita. Luodaan funktio näytön vaihtamiseksi klikkaamalla pluspainikkeesta, jonka myötä avautuu kuvan 25 vasemmanpuoleinen valikko. Valitaan avautuvasta valikosta *Screens* ja *Screen Switch*. Seuraavaksi näytölle avautuu kuvan 25 oikeanpuoleinen valikko, josta tulee valita näyttö, jonka halutaan avautuvan funktiolla. Myöhemmin luotavat painikkeet saadaan toimimaan yhdistämällä niihin funktioita.



Kuva 25. Näytönvaihtofunktion luominen.

Näyttöjen avauksessa voidaan hyödyntää skriptiä, jolloin näytöt saadaan avautumaan automaattisesti, kun ohjelma käynnistetään. Skripti saadaan luotua projektipuusta funktioiden alapuolelta kohdasta *Scripts*. Luodaan Kuvan 26 mukainen *Autostart*-skripti, johon liitetään funktiot, jotka avaavat navigointinäytön, tilinäytön sekä SLD-näytön. Tämän myötä edellä mainitut näytöt avautuvat automaattisesti aina, kun service engine käynnistetään. On syytä huomata, että funktiot toteutetaan järjestyksessä ylhäältä alaspäin, joten niiden järjestyksellä on merkitystä ikkunoiden näkyvyyteen, jos kehysten sijainnit ovat päällekkäisiä.



Kuva 26. Skriptin luominen.

4.2.6 Visualisointi symboleilla ja yhdistetyillä elementeillä

Symboleiden (Symbols) avulla saadaan visualisoitua erilaiset komponentit, tilanteet ja tapahtumat, minkä myötä käyttäjä saa havainnollisen kuvan sähköjärjestelmän rakenteesta sekä tilasta. Symboleihin päästään käsiksi avaamalla projektipuusta valikko *Screens* ja *symbol library*. Symboleita voidaan luoda itse pluspainikkeesta tai tuoda valmiina kohdasta *import XML*. Kun symboli on tallennettu symbolikirjastoon niin sitä voidaan hyödyntää uudelleen raahaamalla se näytölle. Symboleita hyödynnetään esimerkiksi painikkeissa ja kytkinlaitteiden tilan ilmaisussa.

Yhdistetyt elementit (Combined elements) mahdollistavat symboleiden dynaamisuuden, mikä tarkoittaa, että symbolin ulkoasu saadaan muuttumaan tilanteen mukaan. Niitä hyödynnetään erityisesti kytkinlaitteiden visualisoinnissa. Yhdistetyn elementin luominen aloitetaan luomalla uusi symboli ja avaamalla se muokattavaksi ikkunaan. Sen jälkeen avataan ylävalikosta *Elements* ja valitaan *Combined Element*. Seuraavaksi näytölle avautuu kuvan 27 ikkuna, jossa päästään määrittämään elementin ominaisuudet. Kuvassa nähdään esimerkki erottimen Q1 elementistä, jossa ensimmäisenä on valittu yhdistettävä muuttuja vihreään laatikkoon. Kyseinen muuttuja ilmaisee kahdella bitillä erottimen neljä tilaa, mitkä tulee lisätä ruutuun *Conditions*. Keltaiseen laatikkoon asetetaan muuttujan arvo, jolla tila saavutetaan ja punaiseen laatikkoon symboli, joka esittää kyseisessä tilassa. Sama toistetaan kaikille eri tiloille. Kiinni-tila ilmaistaan mustalla symbolilla ja auki-tila valkoisella symbolilla. Virhetilaa kuvaa symboli, jonka päällä on rasti ja välitilaa kuvaa symboli, joka on puoliksi musta.

The screenshot shows a software configuration window with several sections:

- Conditions:** A table with columns for 'Value' and 'Status'. The first row is highlighted in blue. Below the table are buttons for 'New', 'Delete', 'Up', and 'Down'.
- Formula variables:** A list containing one entry: 'X01: S01_132kV_Q1_ST', which is checked and highlighted with a green box.
- Value:** A bit mask editor showing bits 31 down to 0. Bits 0, 1, and 2 are set to 1, while others are 0. A yellow box highlights the bit mask and a 'Value' input field containing the number 3. A 'Formula...' field is located below the bit mask.
- Variable state:** A grid of checkboxes for various variables: M1, M2, M3, M4, M5, M6, M7, M8, NET_SEL, REVISION, PROGRESS, TIMEOUT, INVALID, OFF, FM_TR, RM_TR, ALT_VAL, N_UPDATE, GI, SPONT, T_UNSYNC, BL_870, SB_870, NT_870, T_DEV, OV_870, SE_870, T_INVALID, CB_TRIP, CB_TR_I, and Not acknowledged.
- Display element:** A section with a 'Symbol from library:' field containing a crossed-circle symbol. To its right is an 'Edit linking rules...' button. Below this are 'Color' settings for 'Line color' and 'Fill color', each with a black color swatch and a 'From ALC' checkbox. To the right of these are 'Specific settings' with checkboxes for 'Original symbol colors' and 'Flashing'.

Kuva 27. Yhdistetyn elementin luominen.

Yhdistetyn elementin kopioiminen onnistuu symbolin tavoin, mutta sen yhteydessä täytyy muokata symboliin yhdistettyä muuttujaa, mitä varten avautuu automaattisesti valikko, mikä nähdään kuvasta 28. Kenttiin syötetään alkuperäisestä muuttujasta korvattava osa Q1 sekä osa Q9, millä se korvataan. Samaa toimintoa voidaan hyödyntää myös, jos halutaan kopioida kokonaisia kenttiä. Tällöin korvataan esimerkiksi osa AC01 osalla AC02.

Enter element

Edit linking rule

Linking rule

Source (with '*' or '?') Example: Tank1*

Q1

Replace by Example: Tank2 Preview

Q9

All variables that start with Tank1 are replaced by Tank2, for example: Tank1.Temperature is replaced by Tank2.Temperature

Replace variables using

Name

Identification

Options

Case sensitive

Hierarchic names

Apply multiple rules

Do not consider interlockings

Consider interlocking variables

Consider interlockings

Preview

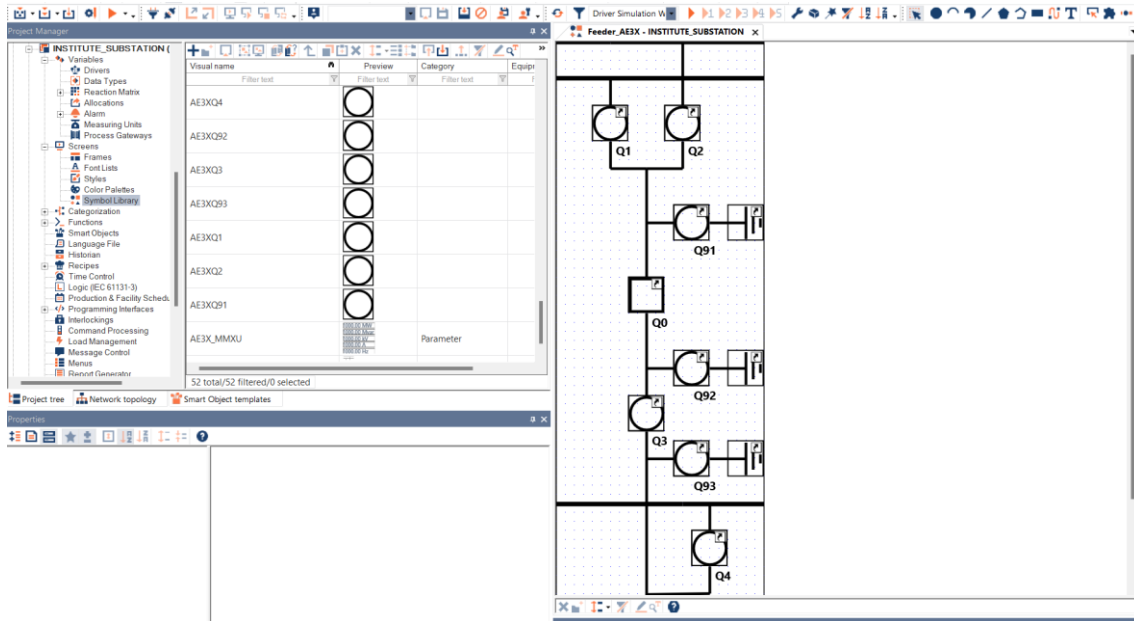
of the variables and functions to be substituted by the source/target texts

Source	Target
S01_132kV_Q1_ST	S01_132kV_Q9_ST

Kuva 28. Muuttujien linkittäminen yhdistettyjen elementtien kopioinnissa

4.2.7 SLD:n luominen

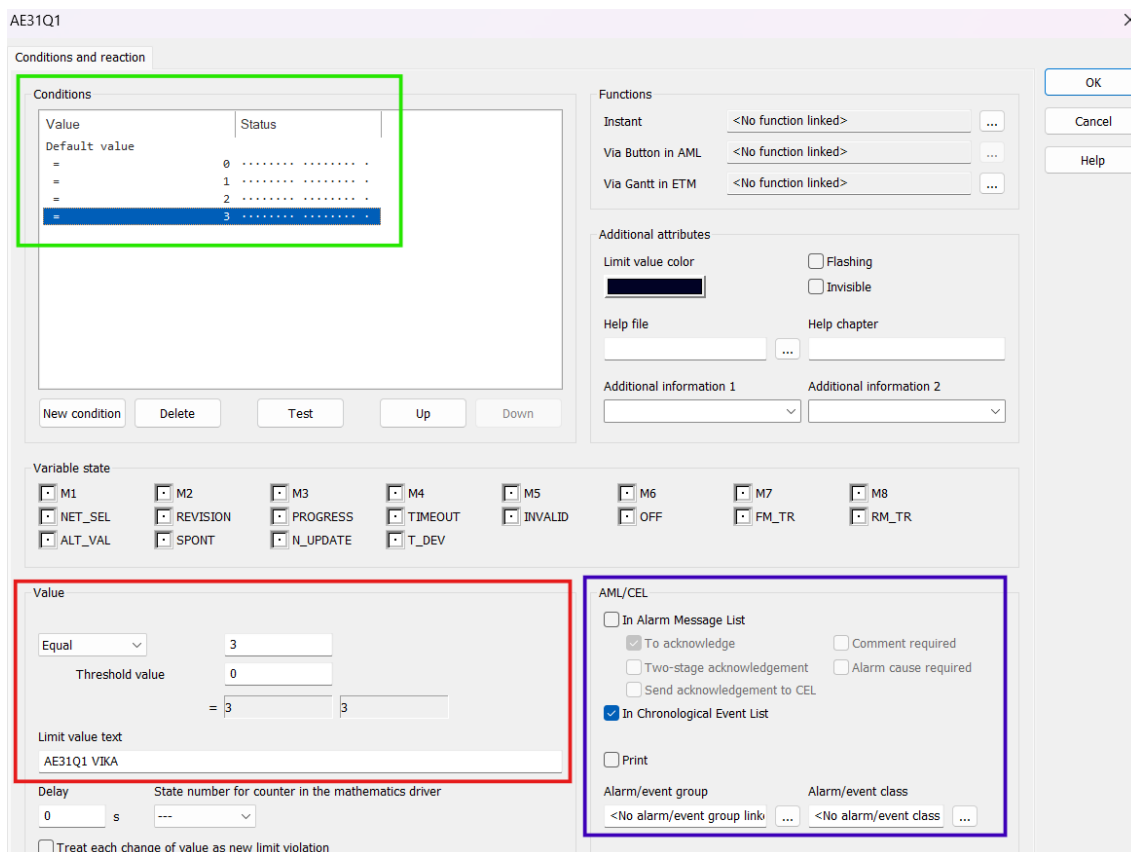
Yhdistettyjen elementtien luonnin jälkeen voidaan alkaa rakentamaan SLD-näyttöä, jossa hyödynnetään yhdistettyjä symboleita. Yhdistetty symboli tarkoittaa symbolia, joka sisältää symboleita tai yhdistettyjä elementtejä. Kuvassa 29 nähdään esimerkki yhdistetystä symbolista, johon on tuotu tarvittavat symbolit ja elementit. Aloitetaan ensimmäisen kentän tekeminen luomalla uusi symboli. Ensimmäisenä voidaan määrittellä symbolin sopiva koko, joka on tässä tapauksessa 230 pikseliä leveä ja 660 pikseliä korkea. Tämän jälkeen symboliin voidaan symbolikirjastosta raahata aiemmin luotuja yhdistettyjä elementtejä ja symboleita. Viivojen luonti tapahtuu klikkaamalla kuvassa 29 näkyvää yläpaneelin painiketta, jossa on viivan symboli. Kun yhdistetty symboli on valmis, niin tallennetaan se. Seuraavaksi avataan muokattavaksi SLD-näyttö kohdasta *Screens* ja raahataan sinne symbolikirjastosta äsken luodun kentän yhdistetty symboli. Samaa symbolia voidaan hyödyntää myös muihin samantyyppisiin kenttiin, kunhan tehdään jokaisen kentän kohdalla kuvassa 28 esitetty muuttujien yhdistäminen. Lopputuloksena saadaan kuvan 35 mukainen SLD-näyttö.



Kuva 29. Yhdistetyn symbolin luominen.

4.2.8 Reaktiomatriisi

Reaktiomatriisin (reaction matrix) avulla voidaan luoda erilaisia logiikoita, mitä muuttujien arvojen muutoksista seuraa. Se muistuttaa hyvin paljon yhdistetyn elementin luomista. Reaktiomatriisia hyödynnetään nyt tapahtumien raportoinnissa tapahtumalistalle. Matriisissa määritellään mikä teksti seuraa muuttujien eri arvoista. Kuvasta 30 nähdään esimerkki reaktiomatriisin luonnista. Vihreässä ruudussa nähdään eri arvot, joilla matriisi toteuttaa toimintoja. Punaisessa ruudussa määritellään arvo sekä teksti, joka sen seurauksena saadaan. Sinisessä ruudussa valitaan, halutaanko teksti esimerkiksi hälytyslistalle tai tapahtumalistalle. Matriisin luonnin jälkeen on vielä tärkeä yhdistää se muuttujan muuttujan asetusten kautta. Matriisi saadaan yhdistettyä muuttujaan avaamalla muuttujan asetuksista kohta *Limit values* ja valitsemalla oikea matriisi valikosta.

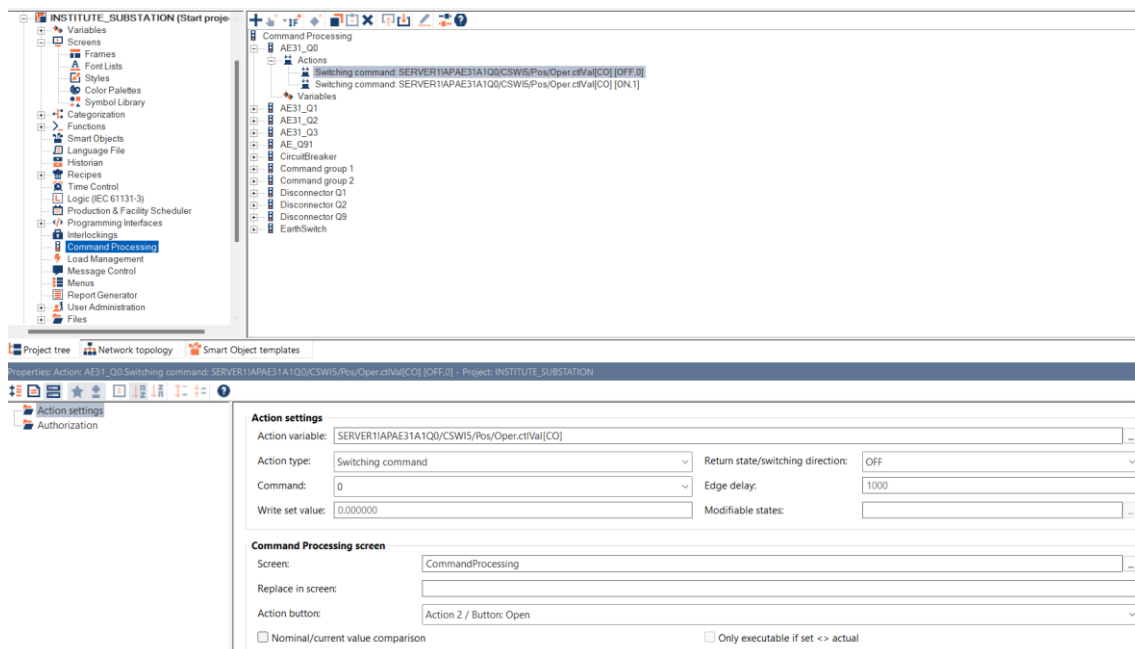


Kuva 30. Reaktiomatriisin luonti.

4.2.9 Komennot

Komentojen käsittelyä (command processing) hyödynnetään kytkinlaitteiden ohjauskäskyjen lähettämisessä. Komentojen toteutuminen kuitataan lähettämällä muuttunut tilatieto IED-laitteelta. Komentoikkunan luominen aloitetaan luomalla komentonäyttö sekä kehys, johon komentonäyttö avautuu. Näytön tyyppiä määrittäessä hyödynnetään valmista *command processing screen* -näyttötyyppiä. Näytön ja kehysten luonnin jälkeen avataan projektipuusta kohta *Command processing* ja luodaan uusi komentoryhmä plus-painikkeesta. Komentoryhmän asetuksiin asetellaan muuttuja, jolla kytkennän toteutumisesta indikoidaan eli kyseisen kytkinlaitteen tilatietomuuttuja. Lisäksi yhdistetään komentoryhmä aikaisemmin luotuun komentonäyttöön. Seuraavaksi luodaan komennot auki ja kiinni. Kuvasta 31 nähdään aukaisukomennon asetukset. Lopuksi komentoryhmä täytyy vielä yhdistää muuttujaan etenemällä muuttujan asetuksissa kohtaan *Write set*

value ja *Command group*. Tämän jälkeen komentoikkuna avautuu aina klikatessa kytkinlaitteen symbolia SLD-ikkunassa.



Kuva 31. Komentojen luonti.

4.2.10 Tapahtumalista

Tapahtumalista kerää järjestelmässä tapahtuvat muutokset ja sen avulla saadaan tietoa mitä tapahtui ja milloin. Tapahtumalistan luonti aloitetaan kehiksen ja näytön luomisella. Näytön luonnissa voidaan hyödyntää Zenonin tarjoamaa valmista tapahtumalistanäyttöä, joka löytyy nimellä *Chronological event list*. Tapahtumat voidaan tuoda tapahtumalistalle määrittämällä raja-arvot muuttujan asetuksissa kohdassa *Limit values* ja valitsemalla tuonti tapahtumalistalle. Tapahtumien raportointiin voidaan hyödyntää myös reaktiomatriisia, mitä käsiteltiin aiemmassa kappaleessa. Tapahtumalistan avaamista varten luodaan funktio ja painike navigointipaneeliin. Service enginessä on mahdollista suodattaa listan tapahtumia tai muokata näytettäviä tietoja tarpeen mukaan klikkaamalla tapahtumalistan yläreunassa olevasta painikkeesta *Filter*.

4.2.11 Hälytyslista

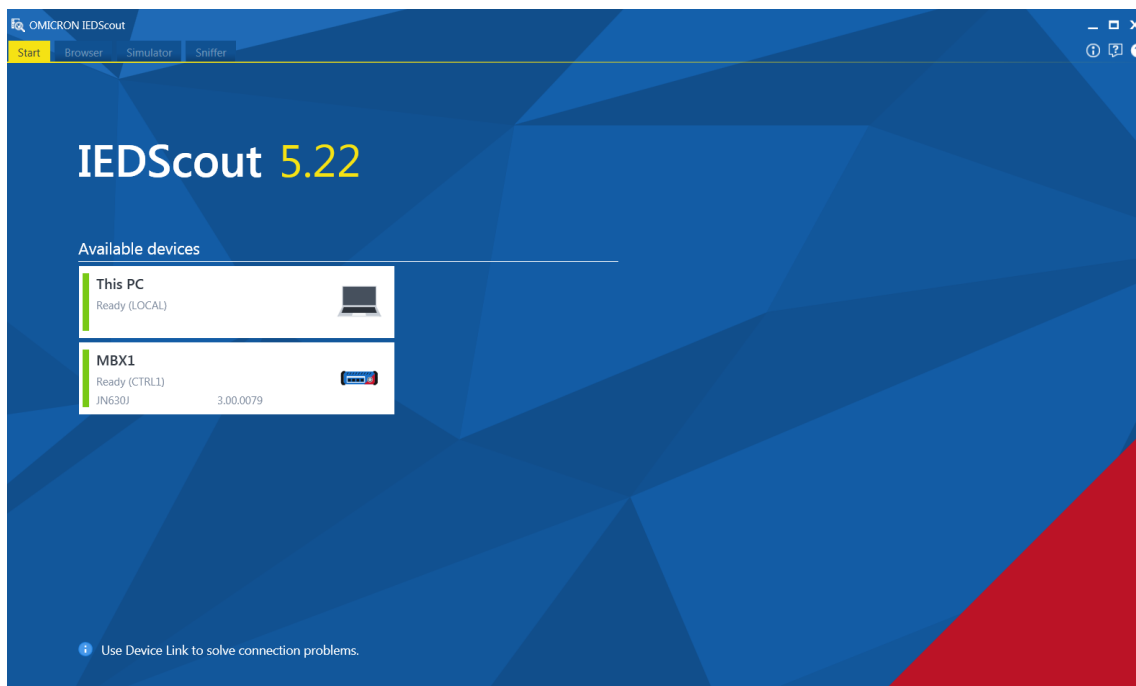
Hälytyslistalle saadaan koottua järjestelmän hälytykset ja sen rakenne muistuttaa tapahtumalista. Hälytyslista voi hyödyntää samaa kehystä kuin tapahtumalista, mutta oma näyttö täytyy luoda. Näytön luomista voidaan nopeuttaa hyödyntämällä Zenonin hälytyslistanäyttöä, joka löytyy uutta näyttöä luodessa valikosta nimellä *Alarm message list*. Hälytykset tuodaan hälytyslistalle, joko määrittelemällä muuttujan asetuksissa raja-arvot ja raportointi hälytyslistalle tai luomalla reaktiomatriisi. Myös hälytyslistan avaamiseksi täytyy luoda funktio ja painike navigointipaneeliin. Projektipuun hälytysvalikosta voidaan luoda erilaisia hälytysluokkia ja -ryhmiä. Valmis hälytyslista pitää sisällään erilaisia suodatusvalintoja, joita voidaan muokata service enginessä hälytyslistan yläreunan suodatuspainikkeen *Filter* kautta.

5 Testaus ja tulokset

Järjestelmän toiminnan testaus suoritetaan perinteisten signaalitestausten tapaisesti eli varmistetaan IED-laitteilta lähtevien signaalien saapuminen HMI:lle. Suurimpana erona perinteisiin testausmenetelmiin on se, että IED-laitteet simuloidaan Omicron-laitteistolla.

5.1 IEDScout-ohjelmiston valmistelu ja käyttö

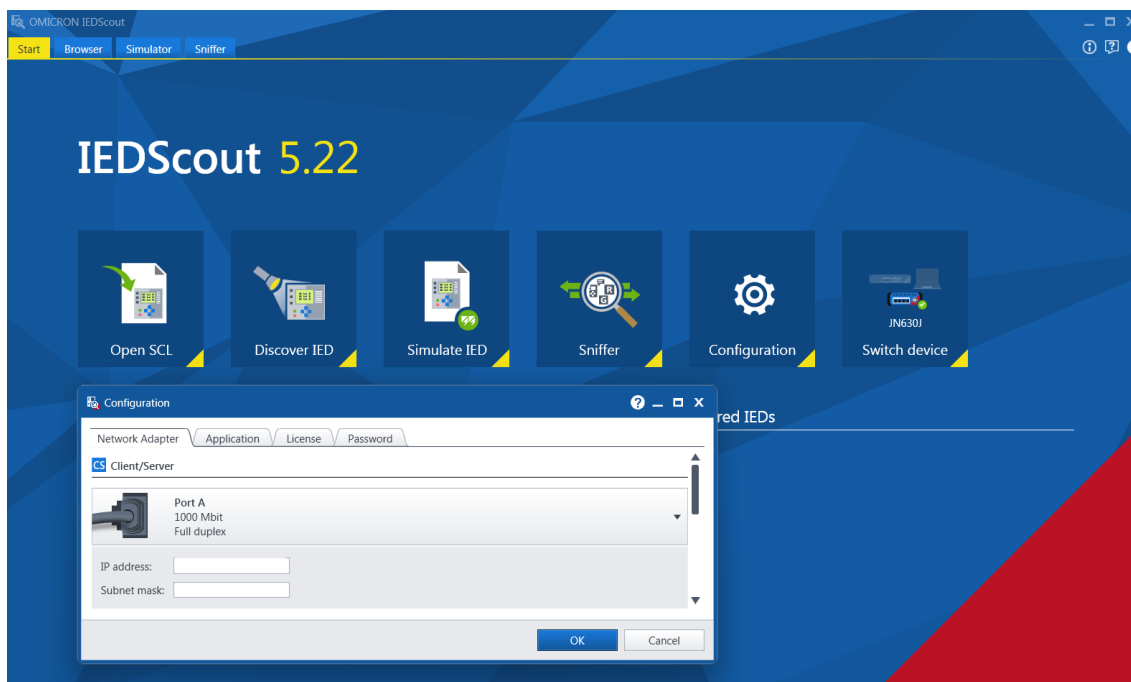
Kuvassa 32 nähdään ohjelmiston käynnistyksen myötä avautuva näyttö. Aloitettaessa ohjelmiston käyttö tulee MBX1 laitteeseen muodostaa yhteys kytkemällä tietokoneesta kaapeli laitteen takana olevaan porttiin CTRL1 tai CTRL2. Tämän jälkeen listalle ilmestyy näkyviin laite, jota klikkaamalla voidaan aloittaa ohjelmiston käyttö.



Kuva 32. IEDScoutin laitteenvallinta

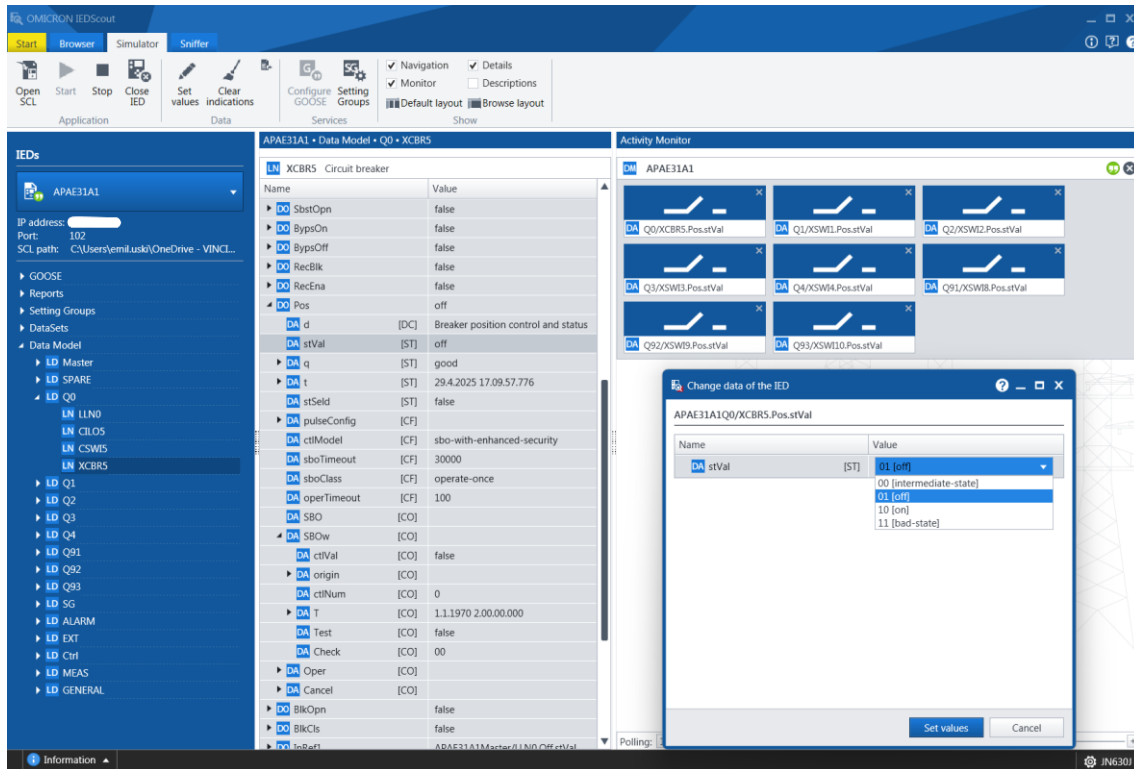
Yhteyden muodostamisen jälkeen avautuu ohjelman varsinainen aloitusnäyttö, joka nähdään kuvassa 33. Ensimmäisenä määritellään laitteen etuportin IP-osoite ja maski *Configuration*-painikkeen kautta. Simuloitavalla sähköasemalla on määritelty osoite

testilaitteelle, joten käytetään sitä. Etuportin kautta muodostetaan yhteys tietokoneella toimivalle HMI:lle ja seuraavaksi tulee asetella tietokoneen kyseisen portin IP-osoite ja maski. Tietokoneen portin osoitteena käytetään aseman osoitelistasta löytyvää HMI:n virtuaalikoneen osoitetta.



Kuva 33. IEDScout aloitusnäyttö.

Osoitteiden määrittelyn jälkeen voidaan tuoda IED-laitteen CID-tiedosto painikkeesta *Simulate IED*. Tuodaan AE31 kenttäohjausyksikön A1 CID-tiedosto, joka pitää sisällään kaiken tarpeellisen tiedon IED-laitteesta. Tiedoston tuonnin jälkeen ohjelmassa avautuu simulointi-ikkuna, joka nähdään kuvassa 34. Simulointi voidaan käynnistää yläpaneelin painikkeesta *Start*. Vasemman reunan paneelista päästään käsiksi data-attribuutteihin, joita voidaan raahata näytölle helpottaakseen niiden tilan havainnointia ja arvon muuttamista. Muuttujan arvon muuttaminen tapahtuu valitsemalla haluttu muuttuja ja klikkaamalla yläpaneelin painikkeesta *Set values*. Näytön oikeaan alakulmaan avautuneesta ikkunasta päästään valitsemaan ja asettamaan haluttu muuttujan arvo. Samanaikaisesti voidaan simuloida myös useampia laitteita ja niiden tuominen ohjelmaan onnistuu esimerkiksi raahaamalla CID-tiedostot simulointinäytölle.

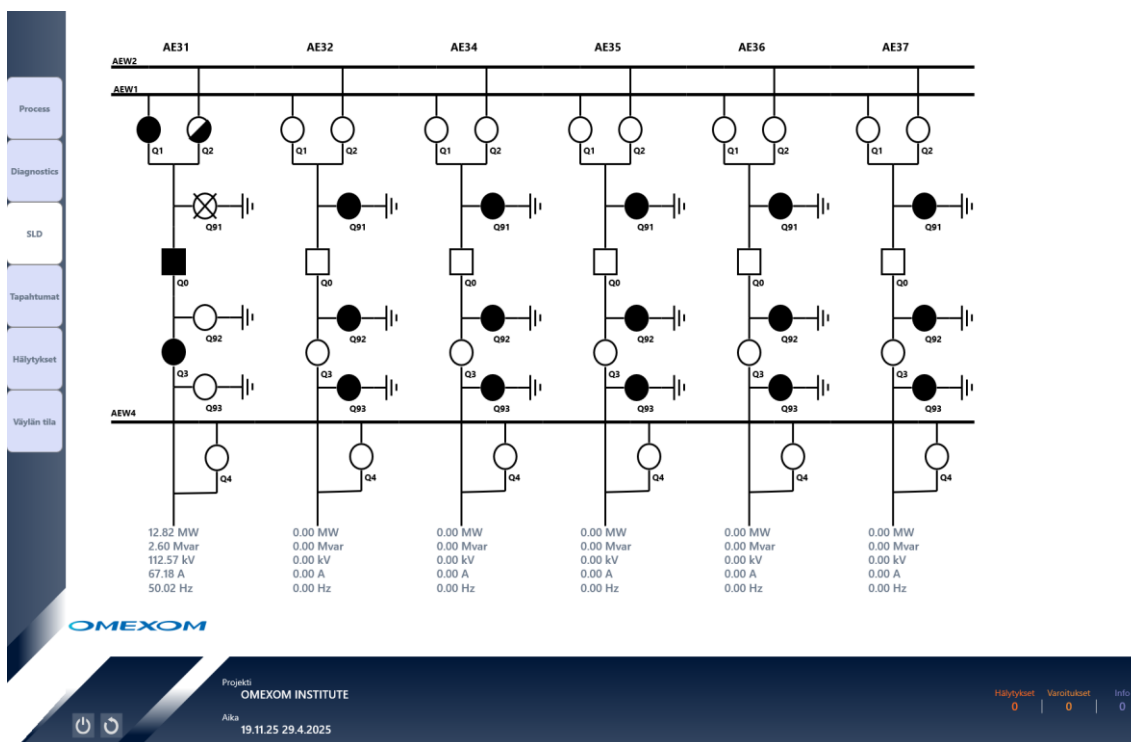


Kuva 34. IED-laitteen simulointi IEDScoutilla.

5.2 Testaus

5.2.1 SLD:n testaus

Luotu SLD sisältää kuusi kenttää, joista testivaiheessa keskitytään kentän AE31 testaamiseen. Kuvassa 35 nähdään aikaan saatu SLD-näyttö. Kentät koostuvat kahdesta pääkiskosta ja yhdestä apukiskosta, sekä katkaisijasta ja seitsemästä erottimesta, joista kolme on maadoituserottimia. Kullakin kytkinlaitteella on neljä tilatietoa, joiden välittyminen kenttäohjausyksiköltä HMI:lle testataan kentän AE31 osalta. Kuvassa olevat symbolit ilmaisevat erottimien kaikki neljä mahdollista tilaa, jotka ovat väli, auki, kiinni ja vika. Konfiguroinnin mukaisesti musta ilmaisee tilan kiinni ja valkoinen tilan auki. Rasti puolestaan ilmaisee virhetilan ja puoliksi musta symboli välitilan. Näiden lisäksi testataan, että kaikki mittaustiedot päivittyvät ikkunan alaosaan.



Kuva 35. SLD:n testaus.

5.2.2 Tapahtumalistan testaus

Tapahtumalistan testauksessa varmistetaan, että kytkinlaitteiden tilatiedot päivittyvät myös tapahtumalistalle. Testaus toteutetaan muuttamalla tilatietomuuttujien arvoja IEDScoutissa. Kuvasta 36 voidaan havaita, että kunkin kytkinlaitteen osalta kaikki neljä tilatiedon muutosta tulevat listalle. Lisäksi listalta nähdään tapahtuman aika sekä signaali. Testataan myös listan suodatuksen toiminta ajan ja tapahtuman nimen perusteella.

Filter [!]*[!][Ret,0d,10h,0m,0s][Cat: (incl) Filter...

Filter profiles Save Import Export Delete

Aika	Tapahtuma	Signaali
30.4.2025 11.09.10.175	AE31Q93 VIKA	SERVER11APAE31A1Q93/XSW110/Pos/stVal[ST]
30.4.2025 11.08.57.700	AE31Q93 KIINNI	SERVER11APAE31A1Q93/XSW110/Pos/stVal[ST]
30.4.2025 11.08.50.160	AE31Q93 AUKI	SERVER11APAE31A1Q93/XSW110/Pos/stVal[ST]
30.4.2025 11.08.39.556	AE31Q93 VÄLI	SERVER11APAE31A1Q93/XSW110/Pos/stVal[ST]
30.4.2025 11.08.08.725	AE31Q92 VIKA	SERVER11APAE31A1Q92/XSW19/Pos/stVal[ST]
30.4.2025 11.08.03.787	AE31Q92 KIINNI	SERVER11APAE31A1Q92/XSW19/Pos/stVal[ST]
30.4.2025 11.07.55.309	AE31Q92 AUKI	SERVER11APAE31A1Q92/XSW19/Pos/stVal[ST]
30.4.2025 11.07.50.494	AE31Q92 VÄLI	SERVER11APAE31A1Q92/XSW19/Pos/stVal[ST]
30.4.2025 11.07.37.007	AE31Q91 VIKA	SERVER11APAE31A1Q91/XSW18/Pos/stVal[ST]
30.4.2025 11.07.26.721	AE31Q91 KIINNI	SERVER11APAE31A1Q91/XSW18/Pos/stVal[ST]
30.4.2025 11.07.18.145	AE31Q91 AUKI	SERVER11APAE31A1Q91/XSW18/Pos/stVal[ST]
30.4.2025 11.07.12.704	AE31Q91 VÄLI	SERVER11APAE31A1Q91/XSW18/Pos/stVal[ST]
30.4.2025 11.06.53.151	AE31Q4 VIKA	SERVER11APAE31A1Q4/XSW14/Pos/stVal[ST]
30.4.2025 11.06.45.204	AE31Q4 KIINNI	SERVER11APAE31A1Q4/XSW14/Pos/stVal[ST]
30.4.2025 11.06.39.514	AE31Q4 AUKI	SERVER11APAE31A1Q4/XSW14/Pos/stVal[ST]
30.4.2025 11.05.39.845	AE31Q4 VÄLI	SERVER11APAE31A1Q4/XSW14/Pos/stVal[ST]
30.4.2025 11.05.26.081	AE31Q3 VIKA	SERVER11APAE31A1Q3/XSW13/Pos/stVal[ST]
30.4.2025 11.02.14.408	AE31Q3 KIINNI	SERVER11APAE31A1Q3/XSW13/Pos/stVal[ST]
30.4.2025 11.02.07.996	AE31Q3 AUKI	SERVER11APAE31A1Q3/XSW13/Pos/stVal[ST]
30.4.2025 11.02.02.865	AE31Q3 VÄLI	SERVER11APAE31A1Q3/XSW13/Pos/stVal[ST]
30.4.2025 11.01.47.462	AE31Q2 VIKA	SERVER11APAE31A1Q2/XSW12/Pos/stVal[ST]
30.4.2025 11.01.43.300	AE31Q2 KIINNI	SERVER11APAE31A1Q2/XSW12/Pos/stVal[ST]
30.4.2025 11.01.38.392	AE31Q2 AUKI	SERVER11APAE31A1Q2/XSW12/Pos/stVal[ST]
30.4.2025 11.01.32.168	AE31Q2 VÄLI	SERVER11APAE31A1Q2/XSW12/Pos/stVal[ST]
30.4.2025 11.01.14.772	AE31Q1 VIKA	SERVER11APAE31A1Q1/XSW11/Pos/stVal[ST]
30.4.2025 11.01.10.273	AE31Q1 KIINNI	SERVER11APAE31A1Q1/XSW11/Pos/stVal[ST]
30.4.2025 11.01.05.784	AE31Q1 VÄLI	SERVER11APAE31A1Q1/XSW11/Pos/stVal[ST]
30.4.2025 11.01.00.089	AE31Q1 AUKI	SERVER11APAE31A1Q1/XSW11/Pos/stVal[ST]
30.4.2025 11.00.19.876	AE31Q0 VIKA	SERVER11APAE31A1Q0/XCBR5/Pos/stVal[ST]
30.4.2025 11.00.13.149	AE31Q0 KIINNI	SERVER11APAE31A1Q0/XCBR5/Pos/stVal[ST]
30.4.2025 11.00.07.718	AE31Q0 AUKI	SERVER11APAE31A1Q0/XCBR5/Pos/stVal[ST]
30.4.2025 11.00.00.142	AE31Q0 VÄLI	SERVER11APAE31A1Q0/XCBR5/Pos/stVal[ST]
30.4.2025.10.55.31.636	Device:INSTITUTE_SURSTATIONF_reloadod4	

OMEXOM

Projekt: OMEXOM INSTITUTE

Aika: 11.11.13 30.4.2025

Hälytykset: 0 Varoitukset: 0 Info: 0

Kuva 36. Tapahtumalistan testaus.

5.2.3 Hälytyslistan testaus

Hälytyslistan osalta testataan kahden hälytyksen saapuminen hälytyslistalle. Hälytysten testaus puolestaan tapahtuu muuttamalla hälytysmuuttujien arvoa IEDScoutissa. Kuvasta 37 nähdään, että listalle tulevat hälytykset SF6 vuoto ja SF6 paine. Myös hälytyksen poistumisesta tulee ilmoitus listalle. Hälytyksen lisäksi listalta nähdään aika sekä signaali. Näytön alapaneelin oikeasta reunasta voidaan havaita aktiiviset hälytykset, vaikka hälytyslista ei olisikaan auki.

Filter: [H] [30.4.2025 11.00.00] [30.4.2025] Filter...

Filter profiles: [Dropdown] Save Import Export Delete

Aika	Hälytys	Signaali
>>30.4.2025 11.37.57.213	AE31Q0 SF6 PAINE HÄLYTTÄÄ	SERVER11APAE31A1ALARM/SCBR_GAPC2/Op/general[ST]
>>30.4.2025 11.33.37.361	AE31Q0 SF6 VUOTO NORMAALI	SERVER11APAE31A1ALARM/SCBR_GAPC1/Op/general[ST]
>>30.4.2025 11.33.12.748	AE31Q0 SF6 VUOTO HÄLYTTÄÄ	SERVER11APAE31A1ALARM/SCBR_GAPC1/Op/general[ST]
>>30.4.2025 11.34.59.984	AE31Q0 SF6 PAINE NORMAALI	SERVER11APAE31A1ALARM/SCBR_GAPC2/Op/general[ST]
>>30.4.2025 11.34.38.540	AE31Q0 SF6 PAINE HÄLYTTÄÄ	SERVER11APAE31A1ALARM/SCBR_GAPC2/Op/general[ST]

OMEXOM

Projekt
OMEXOM INSTITUTE

Aika
11.38.12 30.4.2025

Hälytykset: 1 | Varoitukset: 0 | Info: 0

Kuva 37. Hälytyslistan testaus.

5.3 Tulokset

Konfigurointiprosessin tuloksena saatiin hyvä käsitys Zenonin käytöstä ja keskeisimmistä ominaisuuksista. Tämän työn seurauksena on huomattavasti paremmat valmiudet lähteä toteuttamaan uusia projekteja. Lisäksi testivaiheessa selvisi, kuinka testauksessa voidaan hyödyntää laitteiden simulointia Omicron IEDScoutilla sekä MBX1-laitteella.

Järjestelmän toiminta saatiin testattua onnistuneesti. Ensimmäisenä testattu SLD näytti kaikkien kytkinlaitteiden tilatiedot oikein. Myös jännitteen, virran, taajuuden sekä päto- ja loistehon mittausarvot päivittyivät näytölle oikein. Toisessa kohdassa testattu tapahtumalista toimi tavoitteiden mukaisesti näyttäessään tapahtumat kytkinlaitteiden tilatietojen muutoksista. Myös listan suodattaminen ajan ja tapahtuman perusteella toimi hyvin. Viimeisenä testattu hälytyslista toimi odotusten mukaisesti. Katkaisijan kaasun paineesta sekä vuodosta saatiin hälytykset listalle ja myös tilan palautuessa normaaliksi saatiin ilmoitus. Myös näytön alapaneelin hälytysten indikointi toimi, mikä mahdollistaa

aktiivisten hälytysten havaitsemisen, vaikka hälytyslista ei olisikaan auki. Lisäksi listan suodattaminen ajan ja hälytyksen perusteella toimi myös hälytyslistalla. Ohjaustoiminnoissa ilmeni vielä kehitystarpeita, sillä niitä ei saatu täysin toimimaan diplomityön aika-
taulun puitteissa. Vaikka HMI:ltä annettu ohjaussignaali saapui kenttäohjausyksikölle asti, varsinainen ohjaustoiminto ei toteutunut. Kenttäohjausyksikön lukitukset pyrittiin poistamaan ennen testejä, jotta ne eivät estäisi ohjauksia. Ensimmäisinä jatkotoimenpiteinä voisi tutkia tuleeko ohjauksikäsky antaa kahdella bitillä sekä kuinka ohjaussignaalin muuttujan voi muuttaa kaksibittiseksi Zenonissa. Lisäksi voisi tutkia vielä kenttäohjausyksikön logiikan ehtoja ohjauksille sekä liittyykö ongelma select before operate -toimintoon. Järjestelmän testauksen perusteella voidaan todeta, että Omicron IEDScout ja MBX1 yhdessä ovat toimiva ratkaisu tilanteisiin, joissa kaikkia IED-laitteita ei ole saatavilla käyttöön testivaiheessa.

6 Yhteenveto

Työn päällimmäisenä tavoitteena oli perehtyä sähköaseman käytönvalvontajärjestelmän käyttöliittymän rakentamiseen Copa-Datan kehittämällä Zenon-ohjelmistolla ja kehittää yleistä tietämystä sähköasema-automaatiosta. Yhtenä työn tavoitteena oli mahdollistaa tulevien koestajien helpompi työn aloitus, perehtymällä aiheeseen tämän työn avulla. Teoriaosuuden aiheet valikoituivat osittain sen perusteella, minkä olen itse kokenut olennaiseksi tiedoksi ensimmäisen vuoden aikana koestustehtävissä. Alun teoriaosuudessa perehdyttiin käytönvalvontajärjestelmään sekä sen roolia kasvattaneisiin muutoksiin sähköverkossa. Muun muassa uusiutuvan ja hajautetun sähköntuotannon seurauksena sähköjärjestelmä on monimutkaistunut, minkä seurauksena vaaditaan entistä enemmän reaaliaikaista valvontaa ja nopeaa reagoitua häiriöiden välttämiseksi. Yhdessä asema-automaation kanssa käytönvalvontajärjestelmä mahdollistaa valvonta- ja ohjaustoimintojen keskittämisen, mikä lisää verkon käyttövarmuutta. Muutosajureina ovat toimineet myös automaatiolaitteiden pienentyneet kustannukset, ethernet-pohjaisen tiedonsiirron yleistyminen sekä tiedonsiirtoprotokollien standardointi.

Ensimmäisen tutkimuskysymyksen tavoitteena oli selvittää mistä asema-automaatio koostuu. Aihetta käsitellessä todettiin, että asema-automaatio voidaan jakaa asema-, kenttä- ja prosessitasoon. Tiedonsiirrossa keskeisessä roolissa ovat asemalla tiedonkeruu ja ohjausyksikkönä toimiva ala-asema sekä asemaväylä, joka mahdollistaa tiedonsiirron kenttätason IED-laitteiden sekä ala-aseman välillä. Valvonnassa ja ohjauksessa keskeisessä roolissa on HMI, joka toimii rajapintana käytönvalvontajärjestelmän ja ihmisen välillä. Erilaisilla verkkotopologioilla ja -protokollilla voidaan vaikuttaa tiedonsiirron toimintavarmuuteen. Yleisimpiä käytönvalvontajärjestelmässä hyödynnettäviä tiedonsiirtostandardeja ovat IEC 61850 sekä IEC 60870 versiot 101, 103 ja 104. IEC 61850 mahdollistaa aseman vertikaalisen tiedonsiirron MMS-viesteillä, horisontaalisen tiedonsiirron GOOSE-viesteillä sekä prosessiväylän liikenteen sampled values -viestien avulla. Motiivina IEC 61850 luomiseen on ollut mahdollistaa kommunikointi eri valmistajien laitteiden välillä. Se määrittelee myös SCL-konfigurointikielen sekä useita eri tiedostotyyppejä. Asema-automaatiossa keskeisessä roolissa on myös aikasykronointi. Aikasykronointia

varten on kehitetty protokollat SNTP, NTP sekä PTP, jonka aikatarkkuus riittää myös sessiväylän tarpeisiin.

Toinen tutkimuskysymys oli, miten Copa-Datan kehittämän Zenon-ohjelmiston avulla voidaan konfiguroida käytönvalvontajärjestelmän HMI. Tarkemmin ottaen haluttiin selvittää mitkä ovat konfiguroinnin keskeisimmät komponentit ja työvaiheet sekä mitä lähtötietojatietoja sähköasemasta tarvitaan konfigurointiin. Suurin osa konfiguroinnissa tarvittavista lähtötiedoista saadaan hankkimalla IED-laitteiden CID-tiedostot tai aseman SCD-tiedosto, minkä lisäksi voi olla tarpeen aseman yksiviivakavio ja IP-osoitelista. Zenon sisältää valtavasti erilaisia ominaisuuksia, minkä takia konfiguroinnissa keskityttiin ensisijaisesti järjestelmän toiminnallisuuteen ja visuaaliset toiminnot jätettiin vähemmälle huomiolle. Konfigurointi aloitettiin työtilan ja projektin luomisella, jonka jälkeen alettiin luomaan kehyksiä ja näyttöjä, jotka mahdollistavat käyttöliittymän rakentamisen. Seuraavaksi luotiin IEC 61850 ajuri, joka mahdollistaa kommunikoinnin IED-laitteiden kanssa. Ajurin luonnin jälkeen tuotiin IED-laitteiden CID-tiedostoista ACSI datamallin mukaisia muuttujia, joiden avulla voidaan raportoida muun muassa kytkinlaitteiden tiloja ja mittauksia. Symboleiden ja yhdistettyjen elementtien luonti oli keskeinen osa konfigurointia, sillä niiden avulla rakennetaan SLD, jonka tavoitteena on havainnollistaa järjestelmän tilaa. Yhdistetyt elementit mahdollistavat muuttujan arvon mukaan vaihtuvan symbolin, mikä on ideaalinen kytkinlaitteiden tilojen esittämiseen. Yhdistämällä symboleihin funktioita, saadaan luotua painikkeita, joita hyödynnettiin näyttöjen avaamiseen ja muuhun navigointiin järjestelmässä. Kytkinlaitteiden ohjaustoiminnot voidaan konfiguroida command processing -osiossa. Sekä tapahtuma-, että hälytyslistan luonti tapahtuu kohtuullisen suoraviivaisesti hyödyntämällä valmiita näyttöjä, jonka jälkeen eri tapahtumat ja hälytykset tulee linkittää näille listoille. Raportoinnissa voidaan hyödyntää reaktiomatriiseja, joiden avulla voidaan määrittää muuttujan arvosta riippuva tulostettava teksti. Lopuksi vielä reaktiomatriisin asetuksissa määritetään, kummalle listalle muuttujan arvonmuutos halutaan raportoida.

Viimeisenä tutkimuskysymyksenä oli selvittää kuinka Omicron IEDScout-ohjelmistoa voidaan hyödyntää järjestelmän testaamisessa. IEDScout-ohjelmisto mahdollistaa yhdessä Omicron MBX1-laitteen kanssa IED-laitteiden simuloimisen ja sitä kautta automaation testaamisen ilman IED-laitteita. Testauksen valmistelussa tärkeimpiä vaiheita ovat kaapeleiden yhdistäminen tietokoneen ja MBX1 välille, IP-osoitteiden ja maskien määrittäminen MBX1 etuportille sekä tietokoneen portille. Näiden jälkeen voidaan tuoda CID-tiedostot IEDScout-ohjelmaan ja käynnistää simulointi.

Tämän diplomityön myötä saatiin hyvä yleiskäsitys asema-automaation rakenteesta, laitteista, topologioista ja standardeista sekä protokollista. Työn avulla tulevat koestajat voivat perehtyä asema-automaatioon, mikä laajentaa osaamista ja helpottaa työskentelyn aloittamista. Työssä päästiin perehtymään myös IEDScout-testausohjelmistoon, jonka todettiin olevan toimiva ratkaisu vastaavanlaisiin tilanteisiin, joissa kaikkia IED-laitteita ei ole saatavilla. Lisäksi työssä saatiin käytännön kokemusta käytönvalvontajärjestelmän käyttöliittymän konfiguroinnista Copa-Datan kehittämällä Zenon-ohjelmistolla, mikä antaa hyvän pohjan lähteä kehittämään osaamista oikeiden asiakasprojektien parissa.

Lähteet

- ABB. (2015). IEC 60870-5-101/104 communication protocol manual. Noudettu 2.1.2025 osoitteesta: https://library.e.abb.com/public/a1a7b958919e415b86bcc54332ae3652/REC615_IEC101-104prot_757805_ENb.pdf?x-sign=+s3lj4klvX2DQogwk-bAMBVEYrnrng8ta8A9oxmK5ymyZbYOCque+78qjgvBFJZxfN
- ABB. (2018). REC615 and RER615 IEC 61850 engineering guide. Noudettu 30.3.2025 osoitteesta: <https://techdoc.relays.protection-control.abb/r/REC615-and-RER615-IEC-61850-Engineering-Guide/2.0/en-US/Defining-IEDs-and-starting-IEC-61850-Configuration-tool>
- Adewole, Tzoneva. (2014). Impact of IEC 61850-9-2 standard based process bus on the operating performance. IFAC Proceedings Volumes, 47, 2245-2252. <https://doi.org/10.3182/20140824-6-ZA-1003.00598>
- Cisco. (2017). STP. Noudettu 12.2.2025 osoitteesta: <https://www.ciscopress.com/articles/article.asp?p=2832407&seqNum=5>
- Cisco. (2024). Redundancy protocol configuration guide. Noudettu 12.2.2025 osoitteesta: https://www.cisco.com/c/en/us/td/docs/switches/lan/cisco_ie3X00/software/17_4/b_redundancy_17-4_iot_switch_cg/m_hsr_iosxe_iotswitch.html
- Clarke, G. R., Reynders, D., & Wright, E. (2004). Practical modern SCADA protocols: DNP3, 60870.5 and related systems. Newnes.
- Elovaara, J., Haarla, L., & Otatiето. (2011). Sähköverkot: II, Verkon suunnittelu, järjestelmät ja laitteet. Otatiето Helsinki University Press.
- Falk, H. (2018). IEC 61850 Demystified. Artech.
- Heavy.AI. (n.d). Noudettu 12.2.2025 osoitteesta: <https://www.heavy.ai/technical-glossary/network-topology>
- Huawei. (2024). Fundamentals of CFI. Noudettu 30.3.2025 osoitteesta: <https://support.huawei.com/enterprise/en/doc/EDOC1100306156/2c90d6d5/fundamentals-of-cfi-used-as-the-internal-drop-priority>
- Huotari. (2022). Suojausjärjestelmä ja koestaminen IEC-61850 digitaalisella sähköasemalla. <https://urn.fi/URN:NBN:fi:amk-2022110421985>

- Hussain, Yaohao, Roomi, Mahima & Chang. (2023). An open-source framework for publishing/subscribing IEC 61850 R-Goose and R-SV. *SoftwareX*, 23, 101415. <https://doi.org/10.1016/j.softx.2023.101415>
- Kostic & Frei. (2007). Modelling and using IEC 61850-7-2 (ACSI) as an API. *2007 IEEE Lausanne Power Tech*, 713-719. <https://doi.org/10.1109/PCT.2007.4538403>
- Kuusisto. (2023). Energiamurros mullistaa sähköverkon – Miten taataan joustavuus ja huoltovarmuus. Noudettu 6.11.2024 osoitteesta: <https://www.rejlers.com/fi/ajankohtaista/projektit-ja-nakemykset/energiamurros-mullistaa-sahkoverkon/>
- Lakervi, E., Partanen, J., & Otatiето. (2008). *Sähkönjakelutekniikka*. Otatiето Helsinki University Press
- Liang & Campbell. (2008). Understanding and simulating the IEC 61850 standard. Noudettu 30.3.2025 osoitteesta: <http://hdl.handle.net/2142/11457>
- Lopes, Muchaluat-Saade, Fernandes & Fortes. (2015). Geese: A traffic generator for performance and security of IEC 61850 networks. *2015 IEEE 24th International Symposium on Industrial Electronics (ISIE)*, 687-692. <https://doi.org/10.1109/ISIE.2015.7281552>
- Mackiewicz. (2006). Overview of IEC 61850 and benefits. *2006 IEEE Power Engineering Society General Meeting*. <https://doi.org/10.1109/PES.2006.1709546>
- Mekkanen. (2015). On reliability and performance analyses of IEC 61850 for digital SAS. [Väitöskirja, Vaasan yliopisto]. Osuva. <https://urn.fi/URN:ISBN:978-952-476-645-6>
- Mingxu, Bingjun & Bo. (2023). Design and implementation of industrial real-time ethernet redundant network based on parallel redundancy protocol. *2023 3rd International Conference on Computer Science, Electronic Information Engineering and Intelligent Control Technology (CEI)*, 92-98. <https://doi.org/10.1109/CEI60616.2023.10527882>
- Modbus.org. (2006). Modbus over serial line – Specification and implementation guide. Noudettu 1.1.2025 osoitteesta: https://www.modbus.org/docs/Modbus_over_serial_line_V1_02.pdf

- Moore, Midence & Goraj. (2010). Practical experience with IEEE 1588 high precision time cynchronization in electrical substation based on IEC 61850 process bus. IEEE PES General Meeting, 1-4. <https://doi.org/10.1109/PES.2010.5589311>
- Neagoe, Cristea & Banica. (2006). NTP versus PTP in Computer Networks Clock Synchronization. 2006 IEEE International Symposium on Industrial Electronics, 317-362. <https://doi.org/10.1109/ISIE.2006.295613>
- Networklessons.com. (n.d). Introduction to precision time protocol. Noudettu 6.4.2025 osoitteesta: <https://networklessons.com/ip-services/introduction-to-precision-time-protocol-ntp>
- Nguyen, Semog & Jong. (2016). A novel ring-based dual paths approach for reducing redundant traffic in HSR networks. Computer networks, 110, 338-350. <https://doi.org/10.1016/j.comnet.2016.10.009>
- Omexom. (2025). Tietoa meistä. Noudettu 13.5.2025 osoitteesta: <https://www.omexom.fi/yhtiomme/>
- Omicron. (2021). Functional testing of IEC61850 based substation automation systems. Noudettu 29.4.2025 osoitteesta: <https://www.omicronenergy.com/en/news/details/functional-testing-of-iec-61850-based-substation-automation-systems/>
- Ozansoy, Zayegh & Kalam. (2008). Time synchronisation in a IEC 61850 based substation automation system. 2008 Australasian Universities Power Engineering Conference, 1-7. Noudettu 6.4.2025 osoitteesta: <https://ieeexplore.ieee.org/document/4812969>
- Ozansoy, Zayegh & Kalam. (2009). Object modeling of data and datasets in the international standard IEC 61850. IEEE Transactions on power Delivery, 24, 1140-1147. <https://doi.org/10.1109/TPWRD.2008.2005658>
- Peng, Liang, Luo, Pan & Li. (2016). Analyses and comparisons of SCL files in substation configurator. 2016 3rd International Conference on Systems and Informatics (ICSAI), 297-300. <https://doi.org/10.1109/ICSAI.2016.7810971>

- Siemens. (2020). Siprotec Merging Unit 6MU85 manual. Noudettu 5.4.2025 osoitteesta: https://cache.industry.siemens.com/dl/files/599/109773599/att_1010614/v1/SIP5_6MU85_V08.01_Manual_C074-1_en.pdf
- Tebekaemi. (2016). Designing an IEC 61850 based power distribution substation simulation/emulation testbed for cyber-physical security studies. Noudettu 16.3.2025 osoitteesta: https://www.researchgate.net/figure/Structural-Composition-of-an-IEC-61850-based-IED_fig2_316167202
- Thiriet. (2021). Real-time performance and security of IEC 61850 process bus communications. Noudettu 12.2.2025 osoitteesta: https://www.researchgate.net/figure/Dual-LAN-PRP-network-including-DANP-and-SAN-connected-directly-or-via-RedBox_fig5_350749212
- Thomas, M. S., & McDonald, J. D. (2015). Power system SCADA and smart grids. CRC Press.
- Ussoli & Prytz. (2013). SNTP time synchronization accuracy measurements. 2013 IEEE 18th Conference on Emerging Technologies & Factory Automation (ETFA), 1-4. <https://doi.org/10.1109/ETFA.2013.6648120>
- Wang & Fei. (2010). Research of time synchronization in digital substation based on IEEE 1588. The 2nd International Conference on information Science and Engineering, 2320-2325). <https://doi.org/10.1109/ICISE.2010.5691671>
- Wojdak. (2003). Rapid spanning tree protocol: A new solution from an old technology. Noudettu 27.2.2025 osoitteesta: <http://pdf.cloud.opensystemsmedia.com/xtca-systems.com/PerfTech.Mar03.pdf>