

# AUTOMAATIOSUUNNITTELUN TEHOSTAMINEN TIA PORTAL - KIRJASTON AVULLA

Lahti Precision Oy

LAHDEN AMMATTIKORKEAKOULU  
Tekniikan ala  
Kone- ja tuotantotekniikan koulutusohjelma  
Opinnäytetyö  
Kevät 2019  
Roope Penttinen

## Tiivistelmä

Tekijä(t) Penttinen, Roope	Julkaisun laji Opinnäytetyö, AMK Sivumäärä 27	Valmistumisaika Kevät 2019
Työn nimi <b>Automaatiosuunnittelun tehostaminen TIA Portal -kirjaston avulla</b> Lahti Precision Oy		
Tutkinto Kone- ja tuotantotekniikka		
Tiivistelmä <p>Opinnäytetyön tarkoituksena oli tehostaa Lahti Precision Oy:n ohjaussuunnittelua selvittämällä eri vaihtoehtoja logiikkaohjelmoinnin nopeuttamiseksi sekä yhdenmu- kaistamiseksi. Tavoitteena oli pyrkiä käyttämään mahdollisimman paljon valmiiksi tehtyjä ohjelmakoodeja sekä saada mahdolliset muutokset valmiissa koodeissa kaikil- le suunnittelijoille keskitetyksi.</p> <p>Työssä perehdyttiin työskentelyä tehostaviin keinoihin sekä Siemens TIA Portal - ohjelmointiympäristöön ja sen tarjoamaan kirjastointimahdollisuuteen. Lisäksi työssä perehdyttiin kirjaston mahdollistamien automaattisten suunnittelutoimintojen käyttämi- seen jatkossa.</p> <p>Lopputulokseksi saatiin järkevä ratkaisu suunnittelun tehostamiseksi käyttäen TIA Portal -kirjastoa. Kirjaston käyttämiseen sekä ylläpitämiseen suunniteltiin yritykselle sopiva ratkaisu, joka ei ole liian haastava henkilöstölle. Jatkossa yrityksessä pyritään tekemään projektit kirjaston perusteella.</p>		
Asiasanat Automaatiosuunnittelu, Tehostaminen, TIA Portal, TIA Portal -kirjasto		

## Abstract

Author(s) Penttinen, Roope	Type of publication Bachelor's thesis	Published Spring 2019
	Number of pages 27	
Title of publication <b>Enhance automation engineering using TIA Portal -library</b> Possible subtitle(s)		
Name of Degree Mechanical and production engineering		
Abstract <p>The objective of the Bachelor's thesis was to enhance automation engineering in Lah-ti Precision Oy by finding different methods to speed up and standardize PLC pro-gramming. The goal was to try to use most of the already coded blocks and to get all the changes centralized for everybody.</p> <p>In this thesis we got acquainted with the methods enhancing working, and with the Siemens TIA Portal -software and its library option. This thesis also introduces the possibilities of using automatic engineering, enabled by the library, in the future.</p> <p>As a result, automation engineering was enhanced using the TIA Portal -library. For the use and maintenance of the library, an appropriate solution for the company was designed, which is not too challenging for the staff. In the future, the aim is to make projects based on the library.</p>		
Keywords Automation engineering, enhance, TIA Portal, TIA Portal -Library		

## SISÄLLYS

1	JOHDANTO.....	1
2	LAHTI PRECISION OY .....	2
3	TUOTTAVUUS .....	3
3.1	Tuottavuus yleisesti .....	3
3.2	Turhat kustannukset .....	3
3.3	Yhteistyö .....	3
3.4	Tuotteistaminen .....	4
3.5	Tuotteistamisen tasapaino.....	4
3.6	Tuotteistamisen hyödyt, haitat ja riskit .....	4
3.7	Tuotteistamisen kannattavuus ja tavoitteet .....	5
3.8	Tuotteistamisprosessi .....	6
3.9	Osallistava prosessisykli.....	7
4	TIA PORTAL .....	9
4.1	Simatic Step 7 .....	9
4.2	Simatic WinCC .....	10
4.2.1	Screens.....	11
4.2.2	Koodit.....	11
4.2.3	Tägit.....	12
4.2.4	Hälytykset .....	12
4.2.5	Grafiikkapinnat.....	13
5	SUUNNITTELUN TEHOSTAMINEN .....	14
5.1	Tavoitteet .....	14
5.2	Haasteet ja ratkaisut .....	14
5.3	Kirjasto .....	15
5.4	Tyypit.....	16
5.5	Tyyppien käyttö .....	17
5.6	Master kopiot .....	18
5.7	Master kopioiden käyttö .....	19
5.8	Kielikäännökset.....	19
6	KIRJASTON HALLINTA.....	21
6.1	Kirjaston hallinta kierto.....	21
6.2	Osa-alueen vastaavien vastuut .....	22
6.3	Kirjaston sijainti .....	22

6.4 Kirjaston rakenne .....	22
7 JATKOKEHITTÄMINEN.....	24
8 YHTEENVETO .....	25
LÄHTEET.....	26

## LYHENTEET JA MÄÄRITELMÄT

Datatyypin (Data type)

Käyttäjän määrittämä taulu datasta, jota voidaan muokata keskeisesti yhdestä paikasta. Ne voivat sijaita useamman toiminnon rajapinnassa tai datalohkossa kerrallaan.

Grafiikkapinta (Faceplate)

Ryhmästä grafiikkaelementtejä ja ohjauselementtejä voidaan muodostaa kokonaisuus, jolle voi määrittää rajapinnan.

HMI

Human Machine Interface eli ihmisen ja koneen välinen rajapinta.

I/O

Input ja Output eli logiikan tulot ja lähdöt.

Master kopio (Master copy)

TIA Portal -kirjaston toinen osa-alue, jonne voidaan kirjastoida kopioita PLC:n ohjelmaloikoista sekä HMI:n käyttöliittymän kokonaisuuksista.

PLC

Programmable Logic Controller eli ohjelmoitava logiikkaohjain.

Ponnahdusikkuna (Popup)

Käyttöliittymässä prosessinäkymän päälle avautuva tietyn laitteen tai ohjausalueen hallintasivu.

TIA Portal

Totally Integrated Automation (TIA) Portal on Siemensin ohjelmoitavien logiikoiden, käyttöliittymälaitteiden ja taajuusmuuttajien ohjelmointisovellus.

Toimintolohko (Funktionblock)

Logiikkaohjelmoinnissa käytettävä toiminto, joka suorittaa siihen ohjelmoidun koodin ja säilyttää sille staattisiksi määritellyissä muuttujissa arvot datablokin muistissa.

Tyyppi (Type)

TIA Portal -kirjaston toinen osa-alue, jonne voidaan kirjastoida yksilöllisellä versionumerolla ja yhteydellä projektipuun ja projektikirjaston välillä PLC:n ohjelmalohkoja sekä HMI:n käyttöliittymän osia.

Tägi

Yksilöllinen symbolinen osoite logiikan tuloille, lähdöille ja muistialueen muuttujille.

## 1 JOHDANTO

Opinnäytetyön tavoitteena on selvittää TIA Portaalien tarjoaman kirjaston etuja Lahti Precisionin automaatio-suunnittelussa. Yrityksessä käytetään pääasiassa Siemensin automaatiokomponentteja, jotka ohjelmoidaan TIA Portal -ohjelmistolla. Logiikkaohjelmakoodissa pyritään käyttämään valmiiksi tehtyjä koodeja mahdollisimman paljon, jotta samaa koodia ei tarvitse tehdä useasti. Valmiiden ohjelmaosien siirtyminen projektista toiseen on suoritettu manuaalisesti kopioimalla toisen työntekijän projektista valmiit osat, jolloin mahdolliset muutokset ovat vain yhdessä projektissa ja siitä kopioituissa osissa. Yrityksessä on kokeiltu TIA Portaalien tarjoaman kirjaston käyttämistä ohjelmakoodien ja käyttöliittymän osien taltioimiseksi ja siirtämiseksi uusiin projekteihin aikaisemmin. Osa suunnittelijoista on kokenut sen hyväksi ja toimivaksi tavaksi, josta on herännyt ajatus kirjaston laajemmasta hyödyntämisestä automaatio-suunnittelussa.

Opinnäytetyön alussa perehdytään työskentelyn tuottavuuteen, jonka avulla kirjastoa on lähdetty suunnittelemaan. Siinä tarkastellaan yrityksen turhia kustannuksia, yhteistyötä muiden osastojen kesken, tuotteistamisprosessia sekä sen hyötyjä, haittoja ja riskejä.

Seuraavassa osassa tutustutaan TIA Portaaliin, sen eri ohjelmointikieliin, sen tukemiin käyttöliittymiin ja niiden osiin. Opinnäytetyössä perehdytään TIA Portaalien tarjoaman kirjaston osiin, sen tarjoamiin etuihin suunnittelussa sekä pohditaan sen vaikutuksia suunnittelutyön tehostamiseksi. Lisäksi mietitään muita mahdollisia vaihtoehtoja saman lopputuloksen saavuttamiseksi. Yritykselle tehdään kirjaston käyttöä ja hallintaa varten muodostettu kokonaisuus kirjastosta osana suunnittelua. Opinnäytetyön lopussa pohditaan, kuinka kirjastosta saataisiin mahdollisimman suuri hyöty jatkossa.



## 2 LAHTI PRECISION OY

Alkujaan Lahti Precisionin emoyhtiö ryhtyi valmistamaan rautasänkyjä sekä korjaamaan koneita vuonna 1908 P. Kuivalaisen perustamassa konepajassa. Rautaisten sänkyjen kysynnän vähentyessä vuonna 1914 perustettiin erillinen yhtiö Lahden Vaaka Oy ja alettiin valmistaa punnituslaitteita. Aluksi yritys valmisti vaakoja kotitalouksille sekä kaupoille, josta se siirtyi kehittämään niitä teollisuuteen, varastoihin sekä autojen punnitukseen. (Lahti Precision Oy 2018b.)

Yritys panosti 1960-luvulla sähköisiin vaakoihin ja ensimmäinen sellainen toimitettiin vuonna 1964. Yritys oli kasvussa ja aloitti suuremmat laitostoimitukset valmisbetoniasemille ja lasitehtaisiin. Tämän jälkeen sähköisistä vaaosta tuli entistä suosituimpia ja yrityksen elektroniikkatehdas aloitti toiminnan vuonna 1975. (Lahti Precision Oy 2018b.)

1980 -luvulla ulkomailla alkoi olla kysyntää punnitukseen liittyvälle automaatiolle ja yritys kansainvälistyi alkaen toimittaa kokonaisia prosessikokonaisuuksia. Asiakkaalle alettiin toimittaa valmiita lasitehtaan annostuslaitoksia sekä laasti- ja tasoitetehtaita. (Lahti Precision Oy 2018b.)

Raute Oyj päätti luopua punnitusliiketoiminnasta vuonna 2004. Kolme vuotta sen jälkeen yrityksen nimeksi vaihdettiin Lahti Precision. Tuolloin yrityksen liiketoiminta-alat olivat lasiteollisuus, laastiteollisuus sekä teollinen punnitus. (Lahti Precision Oy 2018b.)

Opinnäytetyön tekemisen aikana toukokuussa yritys ilmoitti myyneen lasiliiketoiminnan saksalaiselle yritykselle. Tämän seurauksena Lahti Precision keskittyy ainoastaan laasti ja punnitus liiketoiminnan kehittämiseen. (Lahti Precision Oy 2018a.)

Pitkän punnitus tietämyksen ansiosta yritys on Pohjoismaissa merkittävin teollisuuden punnitus- ja annostusjärjestelmien toimittaja. Tällä hetkellä Lahti Precision on yksi kansainvälisesti johtavista laastitehtaiden toimittajista erinomaisen punnitus- sekä annostustekniikan johdosta. (Lahti Precision Oy 2018d.)

Yritys sijaitsee Lahdessa Sopenkorven teollisuusalueella, jossa sillä on omat suunnittelu- ja tuotantotilat. Teollisen punnituksen osalta asiakkaat ovat suurimmalta osin kotimaassa. Laastin osalta yrityksellä on paljon kansainvälisiä asiakkaita. Ulkomaan vientiä yrityksellä on Pohjoismaihin, Venäjään sekä sen lähimaihin, Lähi-itään, Amerikkaan, Intiaan sekä Kaakkois-Aasiaan. Tällä hetkellä yrityksessä työskentelee noin 140 henkilöä. (Lahti Precision Oy 2018c.)

## 3 TUOTTAVUUS

### 3.1 Tuottavuus yleisesti

Yrityksen työskentelytapoja, henkilökunnan ammattitaitoa sekä osaamista seurataan yrityksen kannattavuudella. Toiminnan tuottavuutta voidaan tarkastella mittaamalla tehdyn työn suhdetta saatuun tuotokseen (kaava 1). (Sakki 2018.)

$$\text{tuottavuus} = \frac{\text{tuotos}}{\text{työpanos}}$$

Kaava 1. Tuottavuuden määritelmä

Tuotos on ohjelmistosuunnittelussa laitokseen valmiiksi tehty logiikkaohjelma sekä valmis käyttöliittymä. Työpanos on logiikkaohjelman ja käyttöliittymän tekemiseen kulunut aika. Tuottavuutta saadaan parannettua, jos valmiin tuotoksen tekemiseen kuluu aikaisempaa vähemmän aikaa.

### 3.2 Turhat kustannukset

Tuottavuuden kasvattamisessa ilmenee yleensä toimenpiteitä, jotka eivät lisää tuotoksen arvoa vaan vievät vain kustannuksia. Turhat kustannukset tulee saattaa minimaalisiksi tai pyrkiä poistamaan kokonaan, jotta tuottavuus on mahdollisimman hyvä (Sakki 2018). Ohjelmistosuunnittelussa turhia kustannuksia ovat useissa laitoksissa olevat samanlaiset osakokonaisuudet, joihin tehdään yksilöllisiä ohjelmakoodeja tai käyttöliittymän osia. Niihin tulisi tehdä mahdollisimman moneen kohteeseen sopiva ohjelmakoodi, jota voi lähes suoraan käyttää samanlaisissa laitoksissa ilman turhaa työtä.

### 3.3 Yhteistyö

Tehtaan laitteilla on yksilöllinen sijaintinumero, joka esiintyy myös logiikkaohjelmassa laitetta ohjattaessa. Jotta valmiita kokonaisuuksia pystyisi käyttämään mahdollisimman paljon, tulee sopia yhteistyössä mekaniikka- ja sähkösuunnittelun kanssa laitteille vakionumerot sekä vakiosymbolinimet. Laitenumeron sekä symbolinimen ollessa sama eri laitoksissa ihanteellista olisi, jos niissä voitaisiin hyödyntää lähes valmista ohjelmakoodia uusissa laitoksissa suunniteltaessa. Tällä tapaa saataisiin pienennettyä ohjelmakoodin tekemiseen käytettyä työpanosta, jolloin tuottavuus kasvaisi.

### 3.4 Tuotteistaminen

Palvelun ja sen synnyttävän arvon keskittämistä eri osia kuvaamalla ja yhdenmukaistamalla kutsutaan tuotteistamiseksi. Henkilöstön ja asiakkaan osallistaminen tuotteistamisprosessiin auttaa muodostamaan parhaan näkemyksen palvelun tarjoamasta arvosta. Osallistava tuotteistaminen itsessään motivoi ja sitouttaa siihen osallistuvia. Parhaassa tapauksessa ajatusmallit sekä toimintatavat muuttuvat parempaan suuntaan sekä uusia innovaatioita voi syntyä. Toimiva tuotteistaminen mahdollistaa siihen osallistujien oppimisen toisiltaan, joka auttaa yhteisen käsityksen luomisessa asiasta. Yhteinen ymmärrys tuotteistettavasta asiasta yhtenäistää palvelutoimintaa sekä mahdollistaa uudelleen käytävyyttä ja tehokkuutta, jotka tuovat lisäarvoa tuotteistuksen osapuolille. (Tuominen, Järvi, Lehtonen, Valtanen & Martinsuo 2015, 5.)

Tuotteistamista on kahta tasoa. Ulkoisessa tuotteistamisessa asiakkaalle näkyvät palvelut kuvataan ja yhdenmukaistetaan, jolloin muodostuu yhtenäinen kuva asiakkaalle tärkeistä palveluista. Sisäisessä tuotteistamisessa palvelutuotannot määritetään ja kiteytetään kirjoittamalla palveluprosessi, toimintatavat ja vastuut. Myös sisäisessä tuotteistamisessa tulee ajatella asiakkaalle näkyvät palveluprosessit sekä asiakkaan omat prosessit. (Tuominen ym. 2015, 5.)

### 3.5 Tuotteistamisen tasapaino

Vaihtelevissa laitostoimituksissa projektien välillä esiintyy eroja laitteissa, laitteiden määrissä sekä koko prosessissa. Laitosten väliset erot vaativat räätälöintiä jokaisessa projektissa, minkä vuoksi täytyy löytää sopiva tasapaino vakioinnin ja räätälöinnin välille. Tavoitteena on pyrkiä vakioimaan mahdollisimman paljon palveluprosessia, jotta jokaisessa projektissa ei tarvitse keksiä sitä uudelleen, koska se vähentää palvelun tehokkuutta. Tuotteistamista ei saa tehdä liikaa, ettei palvelusta muodostu liian yksinkertainen, jolloin sitä ei pysty hyödyntämään useissa muuttuvissa projekteissa. Yhdeksi suurimmista haasteista nousee oikean tasapainon saavuttaminen räätälöinnin ja vakioinnin välillä. (Tuominen ym. 2015, 5 - 6.)

### 3.6 Tuotteistamisen hyödyt, haitat ja riskit

Yrityksessä palveluja tarjotessa pyritään useimmiten mahdollisimman hyvään tehokkuuteen ja myyntiin, joiden lisäksi tulisi ottaa huomioon yhteisen käsityksen muodostuminen ja tiedon jakaminen. Parempaa ja saumattomampaa palvelua voidaan tarjota yhtenäistä-

mällä toimintatapoja. Tuotteistamisessa tulee havainnoida hyödyt ja haasteet, jotta ne voidaan tasapainottaa. (Tuominen ym. 2015, 6-7.)

Oikeanlailla toteutettuna tuotteistaminen mahdollistaa hyötyjä palveluun. Seuraavia hyötyjä on mahdollista saada:

- Tasalaatuisempi palvelu saavutetaan yhtenäistämällä toimintatapoja ja jakamalla osaamista muille, jolloin laatu ei välittömästi riipu yhdestä henkilöstä.
- Toistettava palvelu saadaan aikaan muodostamalla yhteinen toimintatapa ja vakioimalla palvelua.
- Sisäistä tiedonjakoa ja yhteistyötä saadaan parannettua osallistamalla ja sitouttamalla henkilöstöä tuotteistamisprosessiin.
- Tuotteistuksessa muodostunut yhteinen näkemys palvelusta helpottaa jatkossa palvelun kehittämistä. (Tuominen ym. 2015, 7.)

Tuotteistamisessa voi syntyä haasteita sekä riskejä, joita on hyvä pohtia ennen sen aloittamista. Mahdollisia haasteita ja riskejä on monia:

- Asiakasnäkökulman hävittäminen, jos tuotteistamisessa ei osallista asiakasta prosessiin. Tällöin palvelu ei välttämättä vastaa kaikkia käyttökohteitaan.
- Parhaimmat toimintatavat jäävät käyttämättä, jos henkilöstö kokee tuotteistamisen uhkana.
- Liian tiukat säännökset toimintatavoissa saattavat vähentää improvisointia sekä vähentää asiantuntijoiden motivaatiota.
- Innovointi vähenee tai loppuu kokonaan, jos palvelusta tehdään liian vakioitu. (Tuominen ym. 2015, 8.)

### 3.7 Tuotteistamisen kannattavuus ja tavoitteet

Tuotteistamisessa kannattaa harkita tarkasti, mitä kaikkea kannattaa tuotteistaa, jotta kokonaisuudesta tulee mahdollisimman toimiva. Ensimmäisenä kannattaa pohtia, voidaanko jotakin vakioida. Voiko vakiointi parantaa tehokkuutta ja helpottaako se asiakasta ymmärtämään palvelun paremmin? Yksi vaihtoehto on käyttää modulaarisia kokonaisuuksia, joista on helppo kasata toimiva kokonaisuus. (Tuominen ym. 2015, 8-9.)

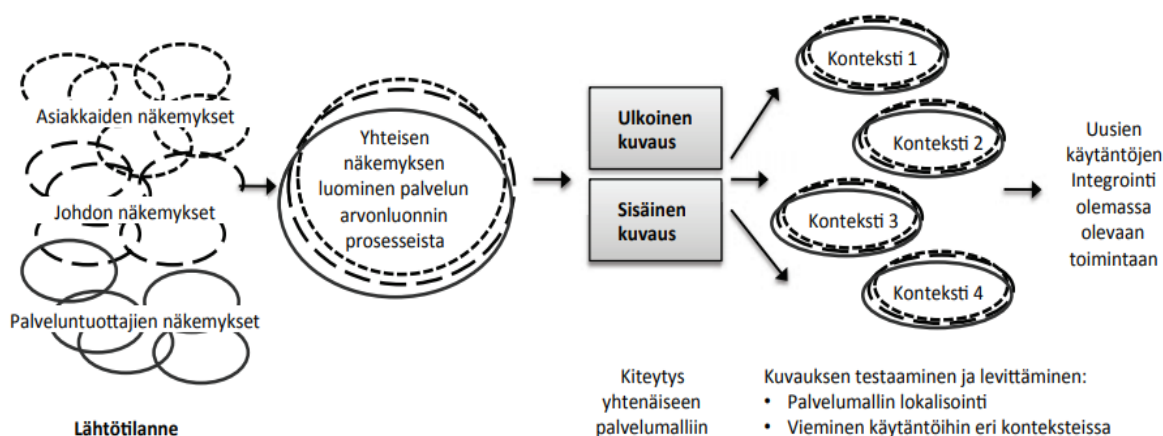
Onnistuneen tuotteistamisen kannalta on lähtökohtaisesti suunniteltava järkevät tavoitteet, joita tuotteistamisella pyritään saavuttamaan. Tavoitteita määriteltäessä on tärkeä huomioida kaikkien osapuolten näkemys asiaan, jotta kaikki ovat mahdollisimman motivoituneita. Hyviä tuotteistamisen tavoitteita ovat

- palvelun tuottamisen tehostaminen
- myynnin ja markkinoinnin tehostaminen
- yhteistyön ja sisäisen viestinnän tehostaminen
- helposti myytävä palvelu
- tasalaatuisuus
- toistettavuus
- jatkokehityksen helppous. (Tuominen ym. 2015, 9.)

### 3.8 Tuotteistamisprosessi

Tuotteistamisprosessi alkaa kaikkien osapuolten näkemysten avaamisesta, jonka perusteella voidaan muodostaa lähtötilanne ja yhteinen päämäärä (kuva 1). Itse prosessi sisältää ulkoisen ja sisäisen tuotteistamisen kiteytettyinä ja kuvattuina. Prosessiin osallistuu henkilöstö sekä asiakas.

Valmis tuotteistettu palvelu testataan eri ympäristöissä ja toimivat uudet toimintatavat liitetään vanhoihin tapoihin. Tämä varmistaa uuden ja vanhan tavan yhteensopivuuden ja käytettävyyden, jolloin ne ovat kaikilla käytettävissä (kuva 1). (Tuominen ym. 2015, 9.)



Kuva 1. Tuotteistamisprosessi (Tuominen ym. 2015, 12.)

### 3.9 Osallistava prosessisykli

1. Tavoitteiden laadinta: Prosessin aloittamiseksi tulee ensimmäiseksi tunnistaa, mitä halutaan tuotteistaa, mitä hyötyä sillä tavoitellaan ja onko sille tarvetta. Kaikkein parhaimman näkemykseen asiasta saa, kun ottaa kaikki prosessiin osallistuvat henkilöt mukaan tavoitteiden pohdintaan.

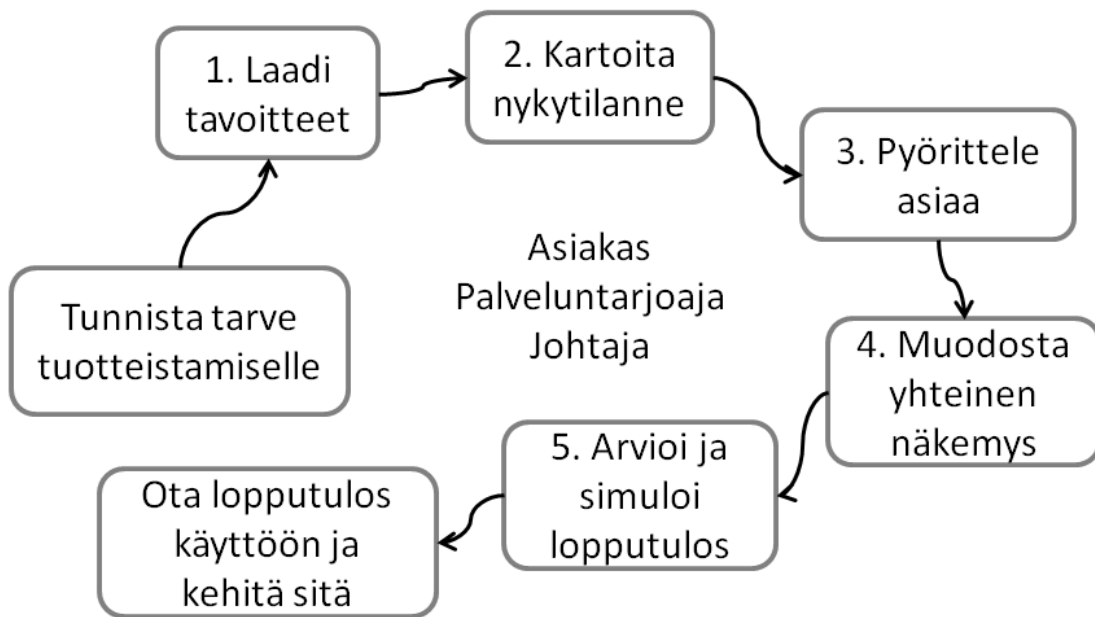
2. Nykytilanteen kartoitus: Olennainen osa tuotteistamista on nykytilanteen kartoittaminen, jossa listataan tulevia vaiheita varten ajan tasalla oleva sen hetkinen tieto, taito ja ymmärrys. Siihen otetaan mukaan asiakas, henkilöstöä ja muut tuotteistamisprosessiin osallistuvat henkilöt.

3. Asian pyörittely: Tarkoituksena on saada aikaan pohdintaa nykyisistä toimintatavoista sekä murrettua kangistuneet ajattelutavat, jotka saattaisivat estää uusien näkökulmien syntymisen.

4. Yhteinen näkemyksen muodostaminen: Kaikki prosessiin liittyvät henkilöt mukaan lukien henkilöt, joilla on valtaa päättää asiasta, kokoontuvat muodostamaan yhteisen näkemyksen. Tällöin uudesta tavasta tulee mahdollisimman laajakatseinen ja se sisältää eri näkökulmia.

5. Lopputuloksen arviointi ja simuloi: Henkilöt, joiden työhön prosessi vaikuttaa pohtivat uuden tavan käyttämistä ja mahdollisia jatkokehitystarpeita. Kun lopputuloksesta suoritetaan yhteinen simulointi, saadaan ihmiset näkemään muiden prosessiin osallistuvien työskentelytavat. Tällöin heidän ymmärrys muiden tekemästä työstä lisääntyy ja jokaisen oma rooli selkeytyy.

Tuotteistamisprosessin valmistuttua tulee uutta toimintatapaa ylläpitää, arvioida ja kehittää jatkuvasti (kuva 2). (Tuominen ym. 2015, 12.)



Kuva 2. Tuotteistamissykli (Tuominen ym. 2015, 12.)

## 4 TIA PORTAL

Siemens tarjoaa TIA Portal -ohjelmistoa, jossa on yhdistettynä ohjelmoitavat logiikat, käyttöliittymä visualisointi, taajuusmuuttaja käytöt, turvatekniikka sekä väyläliitynnät samassa ohjelmassa (Siemens AG 2018e). Erona aikaisempiin Simatic Manager-, Simatic WinCC- sekä Sinamics StartDrive -sovelluksiin on, että aikaisemmin kaikki olivat erillisiä sovelluksia sekä ne avasivat erillisiä ohjelmaikkunoita tietyn asian tekemiseksi. Nykyisessä versiossa kaikki kolme ovat samassa sovelluksessa sekä ohjelma avaa erilaisia välilehtiä eri toimintojen tekemiseksi. TIA Portal kokonaisuudessa Siemens on pyrkinyt tekemään sovelluksesta mahdollisimman käyttäjäystävällisen, tehokkaan sekä luotettavan. (Siemens AG 2018e.)

### 4.1 Simatic Step 7

Logiikkaohjaimen toimivuuden takaamiseksi halutulla tavalla tarvitsee tehdä ohjelmakoodi. Ohjelmakoodin luomiseen käytetään Simatic Step 7 kokonaisuutta TIA Portaalissa. Sillä voidaan toteuttaa logiikkaohjainten ohjelmointi pienistä yksittäisten laitteiden ohjauksista suuriin ja vaativiin kokonaisuuksiin (Siemens AG 2018d). Ohjelmakoodin luomiseen on käytettävissä useampia erilaisia ohjelmointikieliä. IEC 1131-3 -standardin mukaisia ohjelmointikieliä ovat tikapuukaavio (LAD), funktiolohkokaavio (FBD), strukturoituteksti (ST), käskylista (STL) sekä sekvenssiohjelmointi (SFC). (Kettunen 2015, 27.)

Tikapuukaaviota (LAD) voidaan ajatella vastaamaan sähkökuvien relekaavioita, koska se näyttää lähinnä siltä. Siinä ovat normaalisti avoin - sekä normaalisti suljettu -koskettimet vastaavat Boolean muuttujia, jotka kuvaavat logiikkaohjaimen fyysisiä sisääntuloja. Sähkökuvien tapaan siitä löytyy kela, jolla ohjataan logiikkaohjaimen fyysinen lähtö päälle. Lähtö voidaan asettaa päälle laittamalla koskettimet haluttuun tilaan. Ohjelmointi-ikkunan vasemmassa laidassa oleva pystyviiva kuvaa jännitteellistä linjaa. Tikapuukaaviossa voidaan käyttää myös esimerkiksi ajastimia ja laskureita, joita voidaan käyttää koskettimina halutuilla arvoilla. (Kettunen 2015, 7-8.)

Funktiolohkokaavio (FBD) perustuu graafisiinkuvakkeisiin, jotka vastaavat digitaalitekniikan ohjauspiirien toimintoja. Siitä löytyy myös erilaisia toimintalohkoja esimerkiksi ajastimia ja laskureita. (Kettunen 2015, 27.) Ohjelmointityyli on helppo omaksua, mikä vuoksi kokemattomien on helppo aloittaa logiikkaohjelmointi siitä.

Strukturoitu teksti (ST) on korkean tason tekstipohjainen kieli vaativimpiin ohjausratkaisuihin. Siemens käyttää TIA Portalissa SCL (Structured Control Language) ohjelmointikieltä



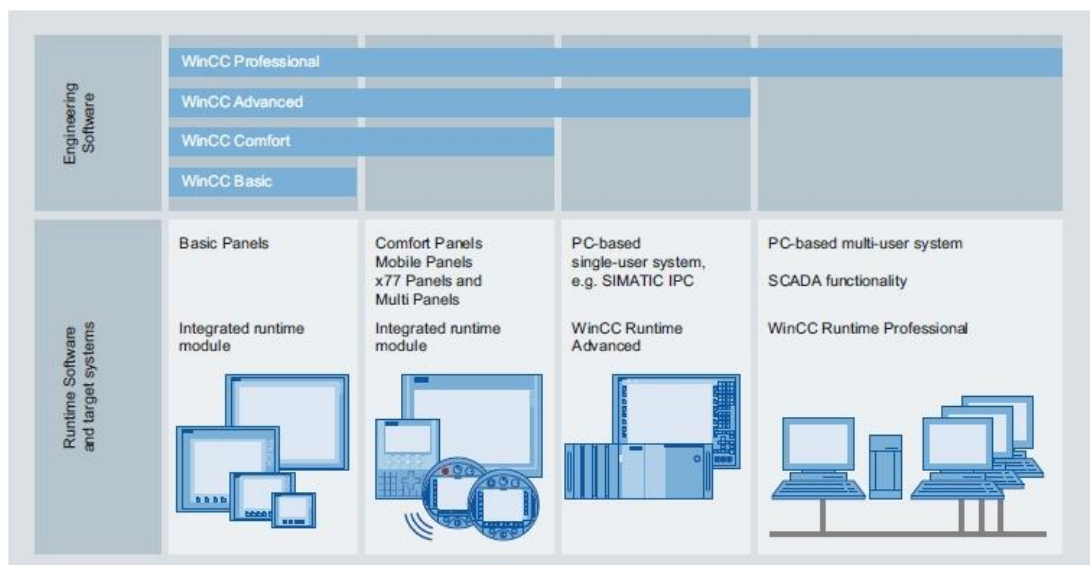
strukturoituna tekstikielenä, joka noudattaa IEC 61131-3 sekä ASCII standardeja. Ohjelmointitavassa voidaan käyttää If...Then-, Case-, For-, While- ja Repeat-rakenteita. (Siemens AG 2018a.) Näillä rakenteilla on helppo toteuttaa erilaisia toimintoja, joita ei muilla kielillä ole välttämättä mahdollista toteuttaa. C-ohjelmointikielen osaajan on helppoa toteuttaa tekstipohjaisella kielellä ohjaustoimintoja helposti.

Käskylista (STL) on tekstipohjainen matalan tason ohjelmointikieli. Ohjelmointityyli koostuu erilaisista operaattoreista ja niihin kohdistuvista operandeista, jotka ovat käskyjen kohteita. (Kettunen 2015, 27 36.)

Sekvenssiohjelmointi (SFC) on graafinen ohjelmointitapa, jota suoritetaan sekvenssi kerrallaan (Kettunen 2015, 27). Seuraavaan sekvenssiin siirtyminen edellyttää sille annettujen ehtojen täyttymistä, jonka jälkeen ohjelma siirtyy suorittamaan seuraavaa sekvenssiä.

## 4.2 Simatic WinCC

Jotta kone saadaan tekemään muutakin kuin vain yhtä toimintoa, tarvitaan HMI (Human Machine Interface) eli ihmisen ja koneen välinen rajapinta. Rajapinta toteutetaan TIA Portalissa WinCC kokonaisuudella, joka kattaa pienistä operointipaneeleista suuriin tietokantapohjaisiin tietokonetoteutettuihin käyttöliittymiin. WinCC:stä on saataville neljä eri versiota, joita ovat Basic, Comfort, Advanced ja Professional. Basic- versiolla pystyy tekemään käyttöliittymää perussarjan paneeleihin. Comfort- versiolla pystyy perus sarjan lisäksi tekemään Comfort- sarjan paneeleita. Advanced tasolla pystyy tekemään aikaisempien lisäksi rajallisia tietokonepohjaisia käyttöliittymiä. Professional- versiolla pystyy tekemään kaikkien edellä mainittujen lisäksi tietokantapohjaisia ratkaisuja. Professional-käyttöliittymässä on lisäksi enemmän ominaisuuksia kuin Advancedissa (kuva 3). (Siemens AG 2018b.)



Kuva 3. WinCC versiot (Siemens AG 2018b.)

#### 4.2.1 Screens

Operointipaneeleissa sekä tietokonepohjaisissa käyttöliittymissä on aina vähintään yksi käyttöliittymäkuva, josta pystyy ohjaamaan laitteita sekä prosessin tilaa. WinCC:stä löytyy valikko editori suurempia kokonaisuuksia varten, jolla pystyy luomaan valikon avulla sivulta toiselle siirtymisen. Pienemmissä tapauksissa sivulta toiselle siirtyessä voidaan käyttää painiketta, jota painettaessa aukeaa uusi kuva.

Graafisen ulkoasun luomiseksi WinCC:ssä on perus kuvakkeita, älykkäitä elementtejä sekä valmiita ohjauskohteita. Perus kuvakkeita ovat erilaiset viivat, muodot, tekstikentät, grafiikka näkymät ja putket. Niissä on vähiten ohjauksellisia ominaisuuksia. Älykkäissä elementeissä on enemmän ominaisuuksia, joilla voidaan ohjata ja kuvata prosessia. Sieltä löytyy erilaisia arvoja ilmaisevia kohteita sekä arvoa ilmaisevia tilapalkkeja, joihin pystyy myös syöttämään prosessiin vaikuttavia arvoja. Älykkäissä elementeissä on myös perus painikkeita, joita painettaessa pystyy muuttamaan bitin tai bittien tilaa. Ohjauskohteissa on kaikkein eniten edistyneimpiä toimintoja, joita ovat trendikuvaajat, taulukot, internetselain, hälytyslista sekä logiikkakoodin monitorointi-ikkuna. (Siemens AG 2017b, 33, 89-92.)

#### 4.2.2 Koodit

Koodit (scripts) ovat C- tai Visual Basic- ohjelmointikielellä tehtäviä toimintoja käyttöliittymään. Koodit mahdollistavat laajempia toimintoja käyttöliittymässä verrattuna TIA Portalin tarjoamiin perustoimintoihin. Paikalliset koodit ovat esimerkiksi käyttöliittymän painiketta

klikatessa tapahtuvaksi määritellyt toiminnot. Paikalliset koodit ovat saatavilla vain WinCC Professional -laitteissa. (Siemens AG 2017b, 1345.)

#### 4.2.3 Tägit

Laitteen ohjaamiseksi tarvitsee logiikkaohjaimen sekä käyttöliittymän välillä olla yhteys, jotta PLC ohjaa laitteita halutulla tavalla. Yhteys luodaan laitteet ja verkot näkymässä, jonka jälkeen käyttöliittymän muuttujat voidaan määrittää vastaamaan jotakin saman tiedostotyyppin muuttujaa logiikkaohjaimella. Muuttujien määrittäminen tapahtuu käyttöliittymän HMI tägit -sivulla.

#### 4.2.4 Hälytykset

Toimintojen tilaa ja virheitä voidaan esittää hälytyksinä käyttöliittymässä. WinCC:stä löytyy järjestelmän määrittämät hälytykset, käyttäjän määrittämät hälytykset ja ohjaimen määrittämät hälytykset (Siemens AG 2017b, 737.). Hälytyksistä selviää helposti virheen muodostumisajankohta sen näyttämän päivämäärän ja ajan perusteella (taulukko 1).

Taulukko 1. Esimerkki hälytyksestä

No.	Aika	Päivämäärä	Hälytysteksti	Tila	Hälytysluokka
3.	10:28:31	27.8.2018	Siilon pinta matalalla	Tuleva	Varoitus

Järjestelmän hälytykset kertovat käyttöliittymälaitteeseen liittyvistä virheistä. Esimerkiksi yhteyden puuttumisesta logiikkaohjaimen ilmoitetaan järjestelmän hälytyksellä. Käyttäjän määrittämiä hälytyksiä ovat käyttöliittymässä muodostettavat analogiset hälytykset, yksittäiset hälytykset ja käyttäjä hälytykset. Analogiset hälytykset menevät päälle muuttujan saavuttaessa tietyn arvon. Yksittäiset hälytykset ilmaistaan, kun muuttuja bitti muuttaa tilansa päälle. Käyttäjä hälytyksille määritetään hälytysnumero ja sitä voidaan käyttää logiikkakoodissa vaikuttamaan ohjelmakiertoon. Ohjaimen määrittämät hälytykset muodostetaan logiikkaohjaimessa, jolloin käyttöliittymä näyttää vain ohjaimen välittämät tiedot operaattorille. Ohjaimen muodostamia hälytyksiä ei ole saatavilla kaikille siemensin tarjoamille käyttöliittymäpääteille kuten perussarjan käyttöliittymäpaneelleille. (Siemens AG 2017b, 745-749.)

#### 4.2.5 Grafiikkapinnat

Ryhmästä grafiikkaelementtejä ja ohjauselementtejä voidaan muodostaa Grafiikkapinta, jolle voi määrittää rajapinnan (Siemens AG 2017b, 262.). Rajapinnan kautta Grafiikkapintaan voi tuoda muuttujia, joilla voi muodostaa dynaamisia ominaisuuksia tai näyttää prosessin tilaa. Grafiikkapinnoissa käyttämällä rajapintaa voidaan muodostaa modulaarisia kokonaisuuksia, joka helpottaa elementin käyttämistä useammassa eri kohteessa.

## 5 SUUNNITTELUN TEHOSTAMINEN

### 5.1 Tavoitteet

Vertaillaessa kolmen eri projektin vakio toimilaitteiden toimintolohkoja ja HMI Grafiikkapintoja löytyi kaikkien projektien toimintolohkoista ja Grafiikkapinnoista pieniä eroja toisiinsa. Eri suunnittelijat tekevät erilaisia ohjelmakoodeja, minkä seurauksena jatkossa projekti projektilta erot kertautuisivat. Erojen poistamiseksi projektien välillä tarvitsee vakio toimintolohkot saada keskitetysti kaikkien suunnittelijoiden projekteihin. TIA Portaalista löytyy kirjasto ominaisuus, jonka avulla on helppo varastoida elementtejä keskitetysti. Toinen vaihtoehto elementtien saamiseksi kaikkien projekteihin voisi olla yhteen tiettyyn projektiin komponenttien varastoiminen. Tässä vaihtoehdossa viimeisimmän version tieto toimintolohkosta sekä muutosten siirtäminen saattaisi muodostua haasteelliseksi.

Kirjaston käyttäminen yrityksessä kaikkien suunnittelijoiden kesken yhdenmukaistaa projekteja sekä niissä käytettyjä ohjelmakoodeja, symbolinimiä sekä kommentteja. Tämä helpottaa toisen suunnittelijan tekemän projektin jatkamista sekä testaamista, koska ohjelmarakenne ja ohjelmointityyli ovat yhdenmukaisia. Uudella tavalla työskentely saattaa aluksi viedä normaalia enemmän aikaa, mutta kaikkien totuttua uuteen tapaan tulee työskentelystä entistäkin tehokkaampaa ja selkeämpää. Tyypitettyjä vakio toimintolohkoja on helppo hallita keskitetysti kirjastonhallinnan sivulla. Sen kautta on myös helppo poistaa projektissa käyttämättömät Tyypit ja niiden versiot, jotka ovat aikaisemmin turhaan roikku-neet projekteissa mukana.

### 5.2 Haasteet ja ratkaisut

Suunnittelijoilla saattaa olla kynnys siirtyä uuteen työskentelytyyliin, jos kirjastoa on vaikea käyttää. Jos tämän hetkinen suunnittelutapa on huomattavasti nopeampi hyvän lopputuloksen saamiseksi, saattaa osa ajatella, että kirjasto ei ole kannattava toimintatapa työskennellessä.

Kirjaston tulee olla helppokäyttöinen, jotta kaikki pystyvät hyödyntämään sen tarjoamia mahdollisuuksia työssään. Uusi työskentelytapa vaatii hyvät ja selkeät käyttöohjeet, jotka sisältävät yksityiskohtaiset ohjeet siitä, kuinka kirjaston avulla aloitetaan uusi projekti. Kirjaston ylläpitäjät tarvitsevat lisäksi selkeät ohjeet kirjaston elementtien luomiseen ja hallitsemiseen.

Tulevan pohjaprojektin eri osa-alueiden mallit eivät saa poiketa liikaa valitusta suunnitteluytyylistä. Kirjastosta tulee löytyä myös suurin osa elementeistä, joita sinne voidaan varastoida. Tarkoituksena on, että kenenkään ei tarvitse uuden projektin alkaessa selvittää, mistä tarvittavat osat löytyvät. Kaikkia edellä mainittuja haasteista pyritään välttämään kirjaston suunnittelussa sekä toteutuksessa. Tällä varmistetaan lopputuloksen olevan mahdollisimman käyttäjäystävällinen ja suunnittelutehokas.

### 5.3 Kirjasto

Elementtejä voidaan varastoida kirjastoon, jotta niitä voidaan hyödyntää uusissa projekteissa. Projektit sisältävät projektikirjaston, joka on yhdistetty projektiin. Projektia tallennettaessa tai suljettaessa tapahtuu myös kyseinen toimenpide projektikirjastoon. Globaalikirjastoja voi käyttää useissa eri projekteissa ja ne eivät ole osana projektia. Kirjastosta voi kopioida ja siirtää elementtejä toiseen kirjastoon. Niitä voi käyttää malliratkaisujen varastoinniseksi keskitettyyn paikkaan. Projektin sisältämistä toimintolohkoista voi tehdä Tyyppejä, jotka löytyvät projektikirjastosta. Projektikirjaston elementit voi kopioida globaalikirjastoon, jolloin useampi henkilö voi hyödyntää elementtejä projekteissaan. Projektikirjasto ja globaalikirjasto sisältävät kahden erilaisia kirjaston elementtejä, jotka ovat Tyypit ja Master kopiot. (Siemens AG 2017c, 7.)

Globaalikirjastoja ovat järjestelmäkirjastot, yrityskirjastot sekä käyttäjänkirjastot. Siemens tarjoaa omille tuotteilleen järjestelmäkirjastoissa valmiita toimintolohkoja, joita voi siirtää projektiin. Järjestelmäkirjastoihin sekä sen kohteisiin ei voi tehdä muutoksia. Nämä kirjastot näkyvät aina automaattisesti TIA Portaalissa sen ollessa avattuna. (Siemens AG 2017c, 8.)

Yrityskirjastot ovat tarkoitettu organisaatioiden käyttöön. Niiden avulla on helppo varastoida yrityksessä yhteisesti käytettäviä kohteita. Kirjaston voi sijoittaa yhteiselle verkkoasemalle, josta siihen on helppo päästä useista eri tietokoneista. Näissä kirjastoissa on TIA Portalin automaattinen hallinta, jolloin uusimmat versiot kirjastoista ovat automaattisesti näkyvillä sen käyttäjillä. Jotta yrityskirjasto toimii oikein ja näkyy kaikilla suunnittelijoilla automaattisesti, TIA Portaalissa, tulee konfigurointitiedostoon lisätä kyseisen kirjaston kansion osoite. (Siemens AG 2017c, 8.)

Käyttäjänkirjastot ovat itsenäisiä kirjastoja, joita voidaan lähettää muille käyttäjille. Sijoitettaessa käyttäjänkirjasto verkkolevylle voi useampi käyttäjä avata kirjaston lukutilassa. TIA Portalin vanhemmilla versioilla tehtyjä kirjastoja voidaan käyttää uudemmassa versiossa, mutta se tulee ensin päivittää uudempaan versioon. (Siemens AG 2017c, 8.) Uudemmallalla

versiolla tehtyjä kirjastoja ei voi avata vanhemmalla versiolla, jonka vuoksi yrityksessä tulee käyttää vain yhtä versiota kerrallaan kirjaston kanssa.

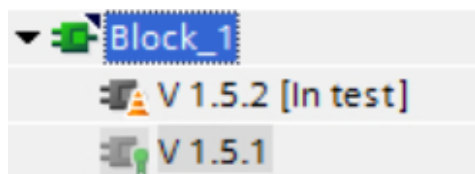
#### 5.4 Tyypit

Tyypit ovat projektin kohteita, joiden tulee voida sijaita ohjelmakierrossa. Tällaisia ovat esimerkiksi toimintolohkot, PLC datatyypit ja käyttöliittymässä käytetyt Grafiikkapinnat. Tyypillä on aina versionumero, jolla on helppo hallita uusimman version käyttämistä sekä päivittämistä. Samaa versiota Tyypistä voi käyttää vain kerran logiikkaohjaimessa, jossa Tyypeistä voidaan käyttää vain uusinta versiota. (Siemens AG 2017c, 51.) Globaaleista kirjastoista kopioidut Tyypit siirtyvät myös projektikirjastoon. Jos projektikirjastossa on valmiiksi siirrettävä Tyypit, siirtyy siitä vain uudemmat versiot kirjastoon. Tyypit esitetään mustalla kolmiolla lohkon kuvakkeen oikeassa yläkulmassa (kuva 4).



Kuva 4. Tyypin musta kolmio (Siemens AG 2017c, 51.)

Tyypien versionumeron kaksi ensimmäistä numeroa voi itse määrittää haluamakseen 1-999 väliltä. Viimeinen numero alkaa yhdestä ja muuttuu aina kun Tyypistä julkaistaan uusi versio. Tyypit voi olla kolmessa eri tilassa, joita ovat Työn alla, Testissä ja Julkaistu. Tyypit voi olla Työn alla vain, jos HMI Grafiikkapintoihin tai HMI datatyyppeihin tehdään muutoksia. Muutoin muutoksia tehdessä Tyypit on Testi tilassa. Työn alla olevia Tyyppejä voi muokata vain kirjasto näkymässä. Testissä olevilla Tyypillä on testiyhteys projektiin, jonka avulla kaikkia käytönaikaisia ominaisuuksia voi testata ennen sen julkaisemista. Kaikki julkaistussa tilassa olevat Tyypit ovat kirjoitussuojattuja. Jos niihin haluaa tehdä muutoksia, tulee ne avata Työn alla tai Testi tilaan (kuva 5). (Siemens AG 2017c, 52.)



Kuva 5. Testissä olevan ja julkaistun Tyypin kuvakkeet. (Siemens AG 2017c, 52.)

## 5.5 Tyypin käyttö

Kirjastossa Tyyppeihin kannattaa sijoittaa kaikkien vakio toimilaitteiden toimintolohkot, graafisia ohjaussekvenssejä sekä kaikki PLC datatyypit. Käyttöliittymässä järkeviä vaihtoehtoja Tyyppeihin ovat HMI Grafiikkapinnat, joita ei saa muuksi kuin Tyypeiksi (taulukko 2). Vakio toimilaitteiden ponnahdusikkunana käytetyt HMI näyttökuvat sekä mahdollisesti lähes muuttumattomien ohjausalueiden ponnahdusikkunoina käytetyt näyttökuvat kannattaa sijoittaa Tyyppeihin.

Vakio toimilaitteiden kohdalla on järkevää pohtia, kuinka varmistetaan kirjastosta löytyvien PLC datatyyppien, PLC toimintalohkojen, HMI Grafiikkapintojen sekä HMI ponnahdusikkunoiden yhteensopivuus. Ratkaisuksi tähän on valikoitunut tyyppinumeron käyttäminen, josta kaksi ensimmäistä numeroa kertovat yhteensopivuudesta. Järkeviä perusteita tyyppinumeron muuttamiseksi on uuden toiminnon lisääminen toimintalohkoon, uuden muuttujan tuominen käyttöliittymään tai mahdollisesti koko toimintolohkon muutos. Jotta kaikkien projektisuunnittelijoiden on helppoa havaita suuret muutokset Tyypeissä, tulee suurissa muutoksissa muuttaa Tyypin ensimmäistä numeroa. Suuria muutoksia, jotka vaativat ensimmäisen numeron muuttamista ovat johonkin osa-alueeseen toimimattomuuden tai toiminnan vääristymisen aiheuttavat muutokset. Toisen numeron muuttamisen aiheuttavat muutokset ovat jonkin toiminnan lisääminen, joka ei vaikuta kuin kyseisen toiminnan puuttumiseen muilla osa-alueilla. Tyypin viimeinen numero muuttuu aina automaattisesti muutoksen yhteydessä, joten viimeisellä numerolla ei ole merkitystä yhteensopivuudessa.

Muutosten jälkeen tulee kommentoida Tyypin kommenttikenttään, mitä muutoksessa on tehty. Tämä helpottaa muita projektisuunnittelijoita havaitsemaan muutokset ja sen merkityksen omassa projektissaan.



Taulukko 2. Yleiskatsaus kirjaston elementeistä (Siemens 2017a, 69.)

Library element	Type	Master copies
<b>SIMATIC PLC</b>		
OB	-	X
FB	X	X
FC	X	X
DB (global)	-	X
Technology objects	-	X
PLC tags	-	X
PLC data types	X	X
Watch and force tables	-	X
Traces	-	X
Text lists	-	X
<b>SIMATIC HMI</b>		
Screens	X	X
Faceplates	X	-
Templates (of screens)	-	X
Pop-up screens	-	X
Slide-in screens	-	-
HMI tags	-	X
Scripts	X	X
Protocols	-	X
HMI UDT	X	-
HMI style	X	-
HMI style sheet	X	-
User administration: User	-	X
User administration: Groups	-	X

## 5.6 Master kopiot

Master kopioihin voidaan sijoittaa lähes kaikki projektin kohteet, josta ne on helppo kopioida toiseen projektiin. Kokonaan valmiiksi konfiguroidun sekä ohjelmakoodin sisältävän logiikkaohjaimen voi suoraan kopioida Master kopioon. Myös valmiita käyttöliittymiä voi kopioida suoraan Master kopioihin riippumatta siitä, onko käyttöliittymä toteutettu tietokonepohjaiseksi vai pieneksi kosketuspaneeliksi. Master kopion siirtämisessä toiseen projektiin siirtyvät myös sen sisältämät Tyypit automaattisesti sekä Tyyppien linkitykset säilyvät. Laitteiden koodin tulee olla käännettynä ennen kuin ne voidaan siirtää Master kopioon, eivätkä ne saa sisältää testissä olevia Tyyppejä. Ennen kuin siirtää Globaalikirjastos-

ta Master kopioita, tulee varmistaa, että kyseinen kopio sopii kyseiseen kohteeseen esimerkiksi WinCC Professionalilla tehtyjä elementtejä ei voi siirtää WinCC Advanced kohteeseen. (Siemens AG 2017b, 45.)

## 5.7 Master kopioiden käyttö

Yrityksen kirjastosta Master kopioista löytyy pohjaprojektit logiikkaohjaimen sekä käyttöliittymän osalta erityyppisille tehtailla. Logiikkaohjaimen pohjaprojekti sisältää valmiin ohjelmarakenteen, vakio toimilaitteiden Tyyppi toimintolohkot, PLC datatyytit, PLC tägi listan sekä kyseiselle tehdastyypille ominaisen ohjausalueen mallin.

Käyttöliittymän pohjaprojektit sisältävät valmiit valikko esimerkit, käyttäjätasot, pääikkunat, prosessikuvat sekä vakio toimilaitteiden Tyyppi ponnausikkunat. Prosessikuvat sisältävät tietynlaisen laitoksen kyseisen alueen tyyppisimpien laitteiden Grafiikkapinnat mallina.

Pohjaprojektit tehdään laitoskohtaisesti mahdollisimman tarkasti kuvaamaan kyseistä laitostyyppiä. Erityyppisiä pohjaprojekteja pyritään tekemään mahdollisimman vähän, jotta muutosten synkronoinnista ei tule liian suurta työtä. Master kopioihin voidaan kirjastoida myös kohdennettuja osioita PLC:lle sekä WinCC:lle, joita voivat olla harvoin käytetyt ohjauselementit.

Laitepuolelta kannattaa kirjastoida valmiiksi oikein konfiguroituja laitteita, joita käytetään yleisesti kuten taajuusmuuttajia, verkkokytkimiä, hajautettuja I/O:ta sekä kaikkea mitä käytetään yleisesti projekteissa.

## 5.8 Kielikäännökset

Useimmiten laitokset toimitetaan eri maihin, jolloin käyttöliittymän tulee olla tehtaan käyttäjien kielellä. Jotta jokaisessa projektissa ei tarvitse kääntää jokaista sanaa uudelleen, löytyy TIA Portaalissa toiminto tekstien ulos ajamiseen Excel muotoiseksi. Siemensiltä löytyy tekstien kääntämiseen sovellus TIA Translation Tool, johon ajetaan sisään projektin tekstitiedosto sekä sanakirjatiedosto. Käännöstiedosto sisältää lähdekielisen sekä kohdekielisen sanan, jonka perusteella sovellus osaa sijoittaa oikean käännöksen vastaamaan sanaa.

Kirjastoinnin yhteydessä nousi esiin käännösten onnistuminen kirjastoitavissa elementeissä. Grafiikkapinnat vaativat kielen lisäämisen asetuksistaan, jotta käännetty teksti saadaan lisättyä siihen. Mikäli käyttöliittymän kuvista on tehty Tyyppejä, tulee niiden olla avattuna muokattavaksi käännöksen ajamiseksi sisään. Käyttöliittymän hälytykset on muo-

dostettu logiikkaohjaimessa, jolloin niiden kääntäminen onnistuu projektin teksti sivun kautta.

Hallintaprojektiin valitaan viisi yleisimmin käytettyä kieltä valmiiksi, jolloin elementit sisältävät valmiiksi yleisimmät kielet. Käyttöliittymän Tyypitettyihin kuviin lisätään tekstit näillä viidellä kielellä valmiiksi, jotta käännöstiedoston lisääminen ei tee uutta Tyypin versiota aina.

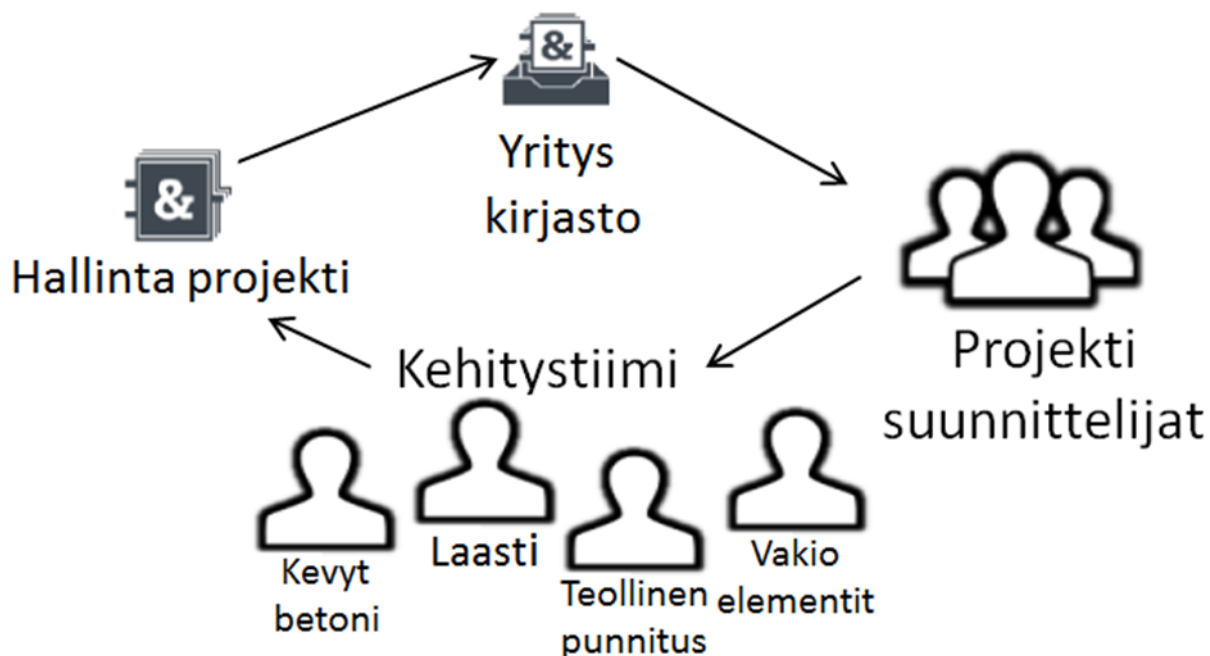
## 6 KIRJASTON HALLINTA

### 6.1 Kirjaston hallinta kierto

Hallintaprojektin perusteella luodaan yrityskirjasto, johon päivitetään kohteita. Kirjastossa on vain uusimmat versiot kaikista elementeistä. Projektisuunnittelijat aloittavat uudet projektit yrityskirjastosta löytyvillä elementeillä. Projektisuunnittelija voi olla myös alihankkija (kuva 6).

Kehitystiimissä mietitään muutoksen tarpeellisuudesta. Hyväksytty valmis muutos siirretään siirtokirjastoon, joka tapahtuu yleensä käyttöönoton jälkeen. Kehittäjiä on yksi jokaisesta osa-alueesta (laasti, kevytbetoni, punnitus, vakio elementit) kohden. Jokainen kehitystiimin jäsen ottaa vastuulleen oman osa-alueensa pohjaprojektit, joissa käytetään valmiita vakio elementtien kehittäjän tekemiä Tyyppejä. Osa-alueen vastaava kehittää tarvittaessa vastuualueelleen erityiset komponentit (kuva 6).

Hallinta projektia käytetään elementtien yhteensopivuuden varmistamiseksi sekä siirtämiseksi yrityskirjastoon. Se sisältää kaikkien osa-alueiden pohjaprojektit logiikkaohjaimelle ja käyttöliittymälle sekä vakio elementit. Jokaiselle vastuualueelle on nimetty henkilö, joka vastaa kyseisen alueen pohjaprojektin kehittämisestä (kuva 6).



Kuva 6. Kirjaston hallintakaavio

## 6.2 Osa-alueen vastaavien vastuut

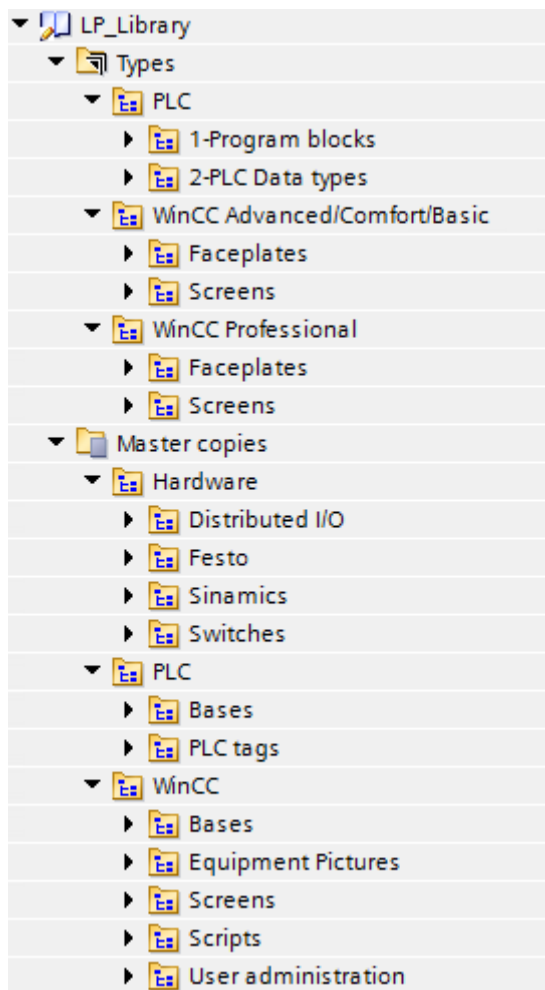
Jokainen osa-alueen vastaava kehittää tarvittaessa omia ohjausalueita, käyttöliittymäkuvia sekä vain kyseisellä alueella käytettyjä elementtejä. Heidän tehtävänä on varmistaa uusimpien vakio elementtien yhteensopivuus oman alueensa pohjaprojekteihin. Vakioelementtien kehittäjä päivittää vakiotoimilaitteiden toimintolohkoja, PLC datatyyppejä, HMI Grafiikkapintoja, HMI ponnahdusikkuna kuvia ja lisäksi kaikkia yleisesti kaikissa projekteissa käytettyjä elementtejä.

## 6.3 Kirjaston sijainti

Kirjasto sijoitetaan yrityksen verkkolevyille, jossa on kansiot kirjastolle, hallintaprojektille, grafiikoille ja dokumenteille. Kirjastokansio sisältää vain yrityskirjaston sekä sen pakatun version mahdollisten virheiden välttämiseksi. Hallintaprojektikansiosta löytyy uusin versio projektista, joka sisältää kaikkien osa-alueiden pohjaprojektit sekä vakio elementit. Grafiikkakansio sisältää uudet käyttöliittymän kuvakkeet. Dokumenttikansiosta löytyy kaikki dokumentit kirjastointiin liittyen, joita ovat selkeät ohjeet kirjaston käyttämiseen sekä teoriaa kirjastosta ja sen elementeistä.

## 6.4 Kirjaston rakenne

Kirjaston valikon tulee olla looginen, jotta sieltä on helppo löytää oikeat kohteet. Yksi toimiva vaihtoehto on tehdä kirjastosta mahdollisimman samanlainen kuin logiikkaohjaimen ja käyttöliittymän valikot ovat. Tämä helpottaa tietojen löytämistä kirjastosta sekä itse projektista. Yritykselle muodostetun kirjaston valikkorakenne on samanlainen kuin logiikkaohjelmassa sekä käyttöliittymässä (kuva 7.)



Kuva 7. Kirjaston valikkorakenne

## 7 JATKOKEHITTÄMINEN

TIA Portalissa on avoin Openness -rajapinta omille sovelluksille, joilla pystyy suorittamaan toimintoja Step 7 ja WinCC:ssä. Parhaimmassa tapauksessa tämä mahdollistaa erillisen sovelluksen avulla valmiiksi tehtyjen ohjelmalohkojen tai käyttöliittymä kuvien perusteella automaattisen logiikkaohjelman tai käyttöliittymän generoimisen. Sovelluksia on mahdollista tehdä Microsoft Visual Studiolla käyttäen C# - tai Visual Basic .NET- ohjelmointikieltä. (Siemens AG 2018c.)

Tällä hetkellä Siemensiltä löytyy SIMATIC Visualization Architect -käyttöliittymien generointi lisäosa TIA Portaaliin, joka toimii yllättävän hyvin rajallisissa käyttöliittymissä. Toinen Siemensin tarjoama sovellus on Opennessia käyttävä Scripter -generaattori, jolla voi generoida logiikkaohjelmia kirjaston elementeistä. Ominaisuuksiensa puolesta Scripter ei vielä tässä vaiheessa vastaa Lahti Precisionin tarpeisiin. Koska yrityksessä projektit eroavat yllättävän paljon toisistaan, ei välttämättä ole järkevää alkaa käyttämään logiikkakoodin generointia. Generaattorin kannalta projektien tulisi olla mahdollisimman samanlaisia, jotta sovelluksen ei tarvitse kattaa hirveää määrää eri toimintoja.

Tulevaisuudessa projektien tulisi sisältää vain tiettyjä vakio komponentteja, joista olisi helppo koota ohjausjärjestelmä sekä logiikkaohjelma. Tämä osaltaan mahdollistaisi paremmin generaattoreiden käyttämisen. Sen avulla olisi mahdollista vähentää rutiinitöitä, joka itsessään vähentää myös mahdollisia virheitä.

## 8 YHTEENVETO

Työn tavoitteena oli selvittää TIA Portal -kirjaston soveltuvuutta automaatio suunnittelun tehostamiseksi ja samalla selvittää muita mahdollisia vaihtoehtoja. Yrityksessä järkevimmäksi vaihtoehdoksi koettiin TIA Portal -kirjaston käyttäminen laajemmin kuin aikaisemmin. Tämän perusteella yritykselle muodostettiin toimintaohjeet TIA Portal -kirjaston hallitsemiseksi sekä käyttämiseksi suunnittelutyössä.

Työn valmistuttua yrityksessä on selvillä, kuinka jatkossa kirjaston avulla tulisi työskennellä teoriassa. Kirjaston käyttöönottamiseksi yrityksessä aloitetaan uusi tehdasprojekti, jossa käytetään kirjastoon sijoitettuja elementtejä. Samalla projektin aikana selvitetään mahdolliset puutteet kirjastosta ja pyritään korjaamaan ne mahdollisimman hyvin.

Kirjaston käyttöönottaminen ja kehittäminen vie aikaa. Prosessiin tulee ottaa mukaan sen tulevia käyttäjiä mahdollisimman laajasti, jotta mahdollisimman moni oppii käyttämään kirjastoa oikein ja suunnittelusta saadaan tehokkaampi.

Tulevaisuudessa voisi harkita Openness -rajapinnan kautta käytettävää sovellusta, jolla generoitaisiin projektit valmiiksi. Sen avulla olisi mahdollista vähentää suunnitteluun kuluva aikaa sekä vähentää mahdollisia virheitä.



## LÄHTEET

Kettunen, A. 2015. PLC:n perusteet, Rakenne, toiminta & ohjelmointi. Reppu: Lahden ammattikorkeakoulun opiskelijoiden tiedotus- ja verkkoopetusympäristö [viitattu 17.6.2018]. Saatavissa: <http://reppu.lamk.fi>

Lahti Precision Oy 2018a. Ajankohtaista [viitattu 2.8.2018]. Saatavissa: <https://www.lahtiprecision.com/fi/uutiset?year=2018>

Lahti Precision Oy 2018b. Historia [viitattu 19.7.2018]. Saatavissa: <https://www.lahtiprecision.com/fi/yritys/historia>

Lahti Precision Oy 2018c. Lahti Precision työnantajana [viitattu 28.7.2018]. Saatavissa: <https://www.lahtiprecision.com/fi/yritys/lahti-precision-tyonantajana>

Lahti Precision Oy 2018d. Yritys [viitattu 28.7.2018]. Saatavissa: <https://www.lahtiprecision.com/fi/yritys>

Sakki, J. 2018. Tuottavuus [viitattu 18.7.2018]. Saatavissa: <http://www.jounisakki.fi/kirja/tuottavuus.htm>

Siemens AG 2017a. Guideline for Library Handling in TIA Portal [viitattu 20.6.2018]. Saatavissa: <https://support.industry.siemens.com/cs/document/109738702/libraries-in-the-tia-portal?dti=0&lc=en-WW>

Siemens AG 2017b. SIMATIC STEP 7 Professional V14 SP1 System Manual WinCC Visualizing processes [viitattu 15.6.2018]. Saatavissa: <https://support.industry.siemens.com/cs/document/109742272/step-7-professional-v14-0?dti=0&lc=en-WW>

Siemens AG 2017c. SIMATIC STEP 7 Professional V14 SP1 System Manual Using libraries [viitattu 15.6.2018]. Saatavissa: <https://support.industry.siemens.com/cs/document/109742272/step-7-professional-v14-0?dti=0&lc=en-WW>

Siemens AG 2018a. SIMATIC STEP 7 Professional (TIA Portal), IEC Editors [viitattu 18.7.2018]. Saatavissa: <https://w3.siemens.com/mcms/automation-software/en/tia-portal-software/step7-tia-portal/step7-professional/Pages/default.aspx?tabcardname=iec%20editors>

Siemens AG 2018b. SIMATIC WinCC (TIA Portal) Engineering Software, Lisenses [viitattu 18.7.2018]. Saatavissa: <https://w3.siemens.com/mcms/automation-software/en/tia-portal-software/wincc-tia-portal/wincc-tia-portal-es/Pages/default.aspx?tabcardname=licenses>

Siemens AG 2018c. TIA Portal Openness: Introduction and Demo Application [viitattu 15.8.2018]. Saatavissa: <https://support.industry.siemens.com/cs/document/108716692/tia-portal-openness%3A-introduction-and-demo-application?dti=0&lc=en-WW>

Siemens AG 2018d. TIA Portal (SIMATIC STEP 7) [viitattu 10.7.2018]. Saatavissa:

[http://www.siemens.fi/fi/industry/teollisuuden\\_tuotteet\\_ja\\_ratkaisut/tuotesivut/automaatioteknikka/ohjelmoitavat\\_logiikat\\_simatic/ohjelmistot/tia\\_portal\\_step7.htm](http://www.siemens.fi/fi/industry/teollisuuden_tuotteet_ja_ratkaisut/tuotesivut/automaatioteknikka/ohjelmoitavat_logiikat_simatic/ohjelmistot/tia_portal_step7.htm)

Siemens AG 2018e. TIA Portal - teollisuusautomaation ohjelmistoalusta [viitattu 18.7.2018]. Saatavissa:

[http://www.siemens.fi/fi/industry/teollisuuden\\_tuotteet\\_ja\\_ratkaisut/tuotesivut/tia\\_portal.php](http://www.siemens.fi/fi/industry/teollisuuden_tuotteet_ja_ratkaisut/tuotesivut/tia_portal.php)

Tuominen, T., Järvi, K., Lehtonen, M., Valtanen, J., Martinsuo, M. 2015. Palvelujen tuotteistamisen käsikirja - Osallistavia menetelmiä palvelujen kehittämiseen [viitattu 19.7.2018]. Saatavissa: <https://aaltodoc.aalto.fi/handle/123456789/16523>