

Tuotantolinjan analytiikka TwinCAT- ympäristössä

Sami Korpi

Opinnäytetyö
Elokuu 2020
Tekniikan ala
Insinööri (AMK), sähkö- ja automaatiotekniikan tutkinto-ohjelma
Automaatiotekniikka

Tekijä(t) Korpi, Sami	Julkaisun laji Opinnäytetyö, AMK	Päivämäärä Elokuu 2020
	Sivumäärä 32	Julkaisun kieli Suomi
		Verkojulkaisulupa myönnetty: x
Työn nimi Tuotantolinjan analytiikka TwinCAT-ympäristössä		
Tutkinto-ohjelma Insinööri (AMK), sähkö- ja automaatiotekniikka		
Työn ohjaaja(t) Kuisma Ari, Häkkinen Veli-Matti		
Toimeksiantaja(t) Apex Automation Oy		
Tiivistelmä <p>Opinnäytetyön tavoitteena oli tutustua Beckhoff Automation GmbH:n TwinCAT Analytics -analysointiohjelmistoon sekä sen tarjoamiin ominaisuuksiin. Työn toinen tavoite oli tutkia ohjelmiston soveltuvuutta teollisuuden tuotantolinjojen analysointiin.</p> <p>Käytännössä TwinCAT Analyticsiin tutustuminen toteutettiin tekemällä toimeksiantajan toimistotiloihin testiympäristö, jossa oli mahdollisuus testata ohjelmiston ominaisuuksia.</p> <p>Työn konkreettisenä tuloksena saatiin vahva käsitys TwinCAT Analyticsin tarjoamista ominaisuuksista sekä sen soveltuvuudesta tuotantolinjojen analysoimiseen. Myös työn toimeksiantajalle saatiin kasattua yksityiskohtaiset ohjeet ohjelmiston käytöstä, joita voidaan hyödyntää helpottamaan tulevia asiakasprojekteja.</p> <p>Testiympäristössä tehdyn testauksen perusteella voitiin päätellä, että TwinCAT Analytics on soveltuva analysointityökalu ainakin tuotantolinjojen yksinkertaiseen kuvailevaan analytiikkaan sekä erityisesti pienten parannusten jälkeen oivallinen lisäys Beckhoffin laajaan tuotevalikoimaan.</p>		
Avainsanat (asiasanat) Data-analytiikka, TwinCAT Analytics, tuotantolinjan analytiikka		
Muut tiedot (Salassa pidettävät liitteet)/		

Author(s) Korpi, Sami	Type of publication Bachelor's thesis	Date August 2020 Language of publication: Finnish
	Number of pages 32	Permission for web publication: x
Title of publication Production Line Analytics in TwinCAT Environment		
Degree programme Bachelor of Engineering, Electrical and Automation Engineering		
Supervisor(s) Kuisma Ari, Häkkinen Veli-Matti		
Assigned by Apex Automation Oy		
Abstract <p>The aim of this thesis was to get familiar with Beckhoff Automation GmbH's TwinCAT Analytics analysis software and features what it offers. The second goal of this thesis was to investigate the suitability of the software for the analysis of industrial production lines.</p> <p>In practice, the introduction to TwinCAT Analytics was implemented by preparing a test environment to the client's office, where it was possible to test the functionality of the software.</p> <p>The concrete result of the thesis was a strong understanding of the features offered by TwinAT Analytics and its suitability for analyzing production lines. Detailed instructions for the use of the software were also compiled for the client, which can be utilized to facilitate future customer projects.</p> <p>Based on testing in the test environment, it was concluded that TwinCAT Analytics is a suitable analysis tool, at least for simple descriptive analysis of production lines and, especially after small improvements, an excellent addition to Beckhoff's wide product family.</p>		
Keywords/tags (subjects) Data Analytics, TwinCAT Analytics, production line analytics		
Miscellaneous (Confidential information)		

Sisältö

1	Johdanto	4
1.1	Opinnäytetyön tausta ja tavoitteet	4
1.2	Tutkimusmenetelmä	5
1.3	Apex Automation Oy	5
2	Datatiede	5
2.1	Data-analytiikan prosessi	6
2.1.1	Kysymysten esittäminen.....	7
2.1.2	Datan hankinta	7
2.1.3	Datan tutkinta.....	8
2.1.4	Datan mallinnus.....	8
2.1.5	Tulosten käsittely.....	8
2.1.6	Toteutus.....	9
2.1.7	Testaus.....	9
2.2	Data-analytiikan tasot	9
2.2.1	Kuvaileva analytiikka	10
2.2.2	Diagnosoiva analytiikka	10
2.2.3	Ennakoiva analytiikka	11
2.2.4	Ohjaileva analytiikka.....	11
2.3	Data-analytiikka teollisuudessa.....	11
3	TwinCAT Analytics	12
3.1	Beckhoff Automation GmbH	12
3.2	TwinCAT Analytics	12
3.2.1	Message Broker	13
3.2.2	Analytics Logger.....	13
3.2.3	Analytics Workbench.....	14
3.2.4	Analytics Storage Provider.....	14
3.2.5	Analytics Dashboard	15
4	Ohjelmiston toimintojen ja ominaisuuksien testaus	16
4.1	Laitteisto	16

	2
4.2	Datan kerääminen ja tallentaminen..... 18
4.2.1	Muuttujien valinta 18
4.2.2	MQTT Message Broker 20
4.2.3	Data Loggerin konfigurointi..... 20
4.2.4	Storage Providerin konfigurointi 22
4.3	Datan mallintaminen 23
4.3.1	Muuttujien tuominen Workbenchiin 23
4.3.2	Algoritmien lisääminen analysointiprojektiin 23
4.3.3	Testausprojektin algoritmit 24
4.3.4	Analysointimallinnuksen generoiminen 26
4.4	Analysoinnin tulosten esittäminen 26
4.5	Testaaminen 27
5	Tulokset 27
6	Pohdinta..... 28
Lähteet 30

Kuviot

Kuvio 1. Datatiede (TTY Pori 2018).	6
Kuvio 2. Data-analytiikan prosessin vaiheet (TTY Pori 2018, muokattu).....	7
Kuvio 3. Data-analytiikan tasot (TTY Pori 2018, muokattu).....	10
Kuvio 4. Analytics topologia (Beckhoff Automation GmbH & Co. KG 2018).....	13
Kuvio 5. Tietokannan rakenne.....	15
Kuvio 6. Generoitu Analytics Dashboard.....	16
Kuvio 7. CX5130 ohjelmitava logiikka.	17
Kuvio 8. Testikokoonpanon periaatekuva.....	18
Kuvio 9. Data Loggerin parametrit.	21
Kuvio 10. Analysointimuuttujien määrittely.	21
Kuvio 11. Analytics Storage Providerin käyttöliittymä.....	22
Kuvio 12. Algoritmien lisääminen analysointiprojektiin.	24
Kuvio 13. Event Timing Analysis -algoritmi Analytics Workbenchissa.	24
Kuvio 14. Edge Counter -algoritmi Analytics Workbenchissa.	25
Kuvio 15. Moottorin lämpötilan analysoinnin algoritmit.....	25
Kuvio 16. Min Max Avg -algoritmi Analytics Workbenchissa.....	26
Kuvio 17. Algoritmien Controllerit Analytics Dashboardissa.	27

Taulukot

Taulukko 1. Analysoitavat muuttujat.	19
---	----

1 Johdanto

1.1 Opinnäytetyön tausta ja tavoitteet

Teollisuudessa on ollut pitkään kasvava paine parantaa tuotannon tehokkuutta sekä samalla vähentää ylimääräisiä kustannuksia. Tämän vuoksi moni teollisuuden toimija on osoittanut kiinnostusta data-analytiikan hyödyntämiseen tuotannon tehostamisessa ja sen vaikutuksesta lähes kaikki automaatiolaitetoimittajat ovat heränneet tarjoamaan laitteistoonsa työkaluja, joita hyödyntämällä voidaan automaatiojärjestelmästä kerätä mittaustietoa eli dataa, jota analysoimalla on mahdollisesta saada tuotantoa tehostettua.

Tässä opinnäytetyössä tutustuttiin saksalaisen automaatiolaitetoimittaja Beckhoffin tarjoamaan TwinCAT Analytics-ohjelmistoon. TwinCAT Analyticsilla voidaan valmiiksi luoduilla algoritmeilla käsitellä tuotantolinjalta tulevaa dataa ja esittää se visuaalisesti tuotantolinjan valvomossa. Käytännössä TwinCAT Analyticsiin ja sen ominaisuuksiin tutustuminen toteutettiin tekemällä testiprojekti toimeksiantajan toimistotiloissa. Opinnäytetyössä käsitellään myös yleistä teoriaa datatieteestä ja data-analytiikasta, jotta saadaan luotua ymmärrys mitä data-analytiikan ohjelmistoilta täytyy vaatia.

Opinnäytetyön aiheesta hyvin mielenkiintoisen muodosti itse TwinCAT Analytics -ohjelmisto, joka on ainakin toistaiseksi hyvin vähän hyödynnetty analysointityökalu, mutta tulevaisuudessa sen käyttö tulee todennäköisesti lisääntymään. Tämä aiheutti myös haasteita opinnäytetyön suorittamiselle, koska ohjeita ohjelmiston käytölle oli saatavilla hyvin suppeasti.

Tietoperusta opinnäytetyöhön kerättiin hyödyntämällä Beckhoffin tarjoamia käyttöohjeita sekä esimerkkiprojekteja. Yleinen teoria data-analytiikasta kasattiin hyödyntämällä aiheeseen liittyvää kirjallisuutta ja aikaisemmin tuotettuja Pro gradu -tutkielmia.

1.2 Tutkimusmenetelmä

Opinnäytetyö oli konkreettinen kehittämistehtävä, jonka tavoitteena oli kehittää opinnäytetyön toimeksiantajan valmiutta tehdä samantyyppisiä projekteja tuottamalla operointiohje TwinCAT Analyticsin käytöstä sekä tutkia ohjelmiston soveltuvuus käytettäväksi tulevien analysointiprojektien työkaluna. (Kananen, J. 2015)

Opinnäytetyön tekemisen aikana haettiin vastauksia ensisijaisesti kahteen seuraavaan kysymykseen.

1. Miten TwinCAT Analytics saadaan sulautettua tehtaan automaatiolaitteistoon?
2. Onko ohjelmisto soveltuva käytettäväksi tuotantolinjan analysoinnissa?

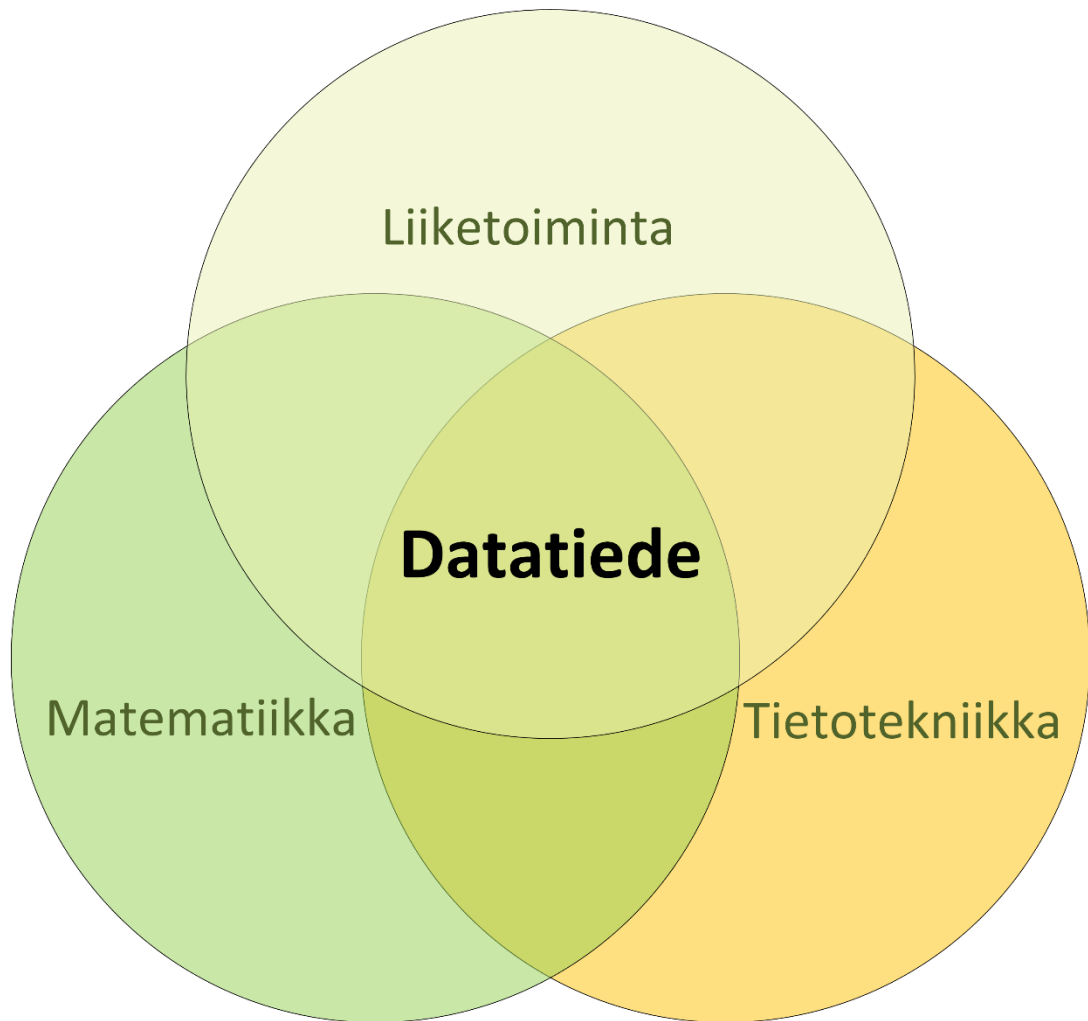
1.3 Apex Automation Oy

Toimeksiantajana tässä opinnäytetyössä toimi Apex Automation Oy. Apex Automation Oy on vuonna 1993 perustettu kokkolalainen automaatio- ja sähkösuunnitteluun erikoistunut yritys. Yrityksessä työskentelee tällä hetkellä yhteensä noin 55 työntekijää Kokkolassa ja Vaasassa.

2 Datatiede

Data-analytiikkaan vahvasti liittyvä tieteenala on datatiede. Datatiede on tieteenala, joka on erikoistunut tiedon eli datan käsittelyyn. Käytännössä datatiede on liiketoiminnan, matematiikan ja tietotekniikan yhdistelmä ja sen osaajilta vaaditaan kaikkia näitä taitoja (Kuvio 1) (TTY Pori 2018).

Lähes aina datatieteessä liiketoiminta määrittelee halutut tavoitteet. Tavoitteista selviämiseen tarvitaan matematiikkaa, jonka kaavojen ja sääntöjen avulla voidaan käsitellä dataa. Koska käsiteltävät datamäärät ovat valtavia, tarvitaan tietotekniikkaa, jonka avulla voidaan suorittaa matematiikan tarjoamat laskutehtävät (TTY Pori 2018).

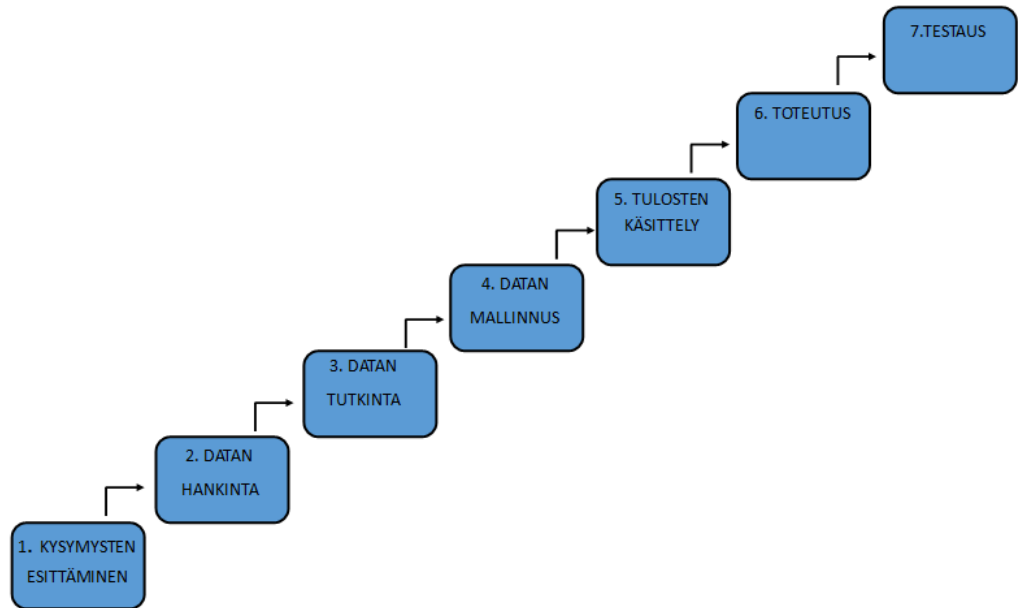


Kuvio 1. Datatiede (TTY Pori 2018).

2.1 Data-analytiikan prosessi

Yksinkertaisesti ilmaistuna data-analytiikan prosessi on toimintatapa, jonka tarkoitus on kertoa tietoa prosessista tai sovelluksesta. Tämä onnistuu keräämällä raakadataa ja käsittelemällä sitä matemaattisilla algoritmeilla, sekä esittämällä algoritmien tulokset yleensä visuaalisessa muodossa. Data-analytiikan prosessista on olemassa useita erilaisia malleja, mutta tässä työssä tutustutaan tarkemmin yksinkertaiseen 7-vaiheiseen data-analytiikan prosessiin. (Rouse 2019).

7-vaiheisen data-analytiikan prosessin vaiheet ovat 1. Kysymyksen esittäminen, 2. Datan hankinta, 3. Datan tutkinta, 4. Datan mallinnus, 5. Tulosten käsittely, 6. Toteutus ja 7. Testaus (Kuvio 2) (TTY Pori 2018).



Kuvio 2. Data-analytiikan prosessin vaiheet (TTY Pori 2018, muokattu).

2.1.1 Kysymysten esittäminen

Ensimmäisessä data-analytiikan prosessin vaiheessa asetellaan kysymykset, joiden vastaukset halutaan selvittää prosessin avulla. Kysymyksenä voisi olla esimerkiksi: Minä ajankohtana tuotannossa on eniten virheitä? Nämä kysymykset auttavat asettamaan analyysiprosessille tavoitteet, joita kohti edetään prosessin aikana (Shung 2014).

Kysymysten asettelu vaatii analysoitavan kohteen asiantuntijan sekä data-analyysin tekijän jatkuvaa vuorovaikutusta, koska esitettävät kysymykset liittyvät suoraan analysoitavan kohteen toimialaan (Eckerson 2007).

2.1.2 Datin hankinta

Prosessin toisessa vaiheessa täytyy haluttu data kerätä ja tallentaa järjestelmästä. Datin kerääminen on aina tapauskohtaista, mutta esimerkiksi automaatiojärjestelmästä voidaan kerätä mittaustietoa ja tallentaa ne ulkoiselle tietokannalle (Shung 2014).

2.1.3 Datan tutkinta

Ennen varsinaista data-analyysin määrittelyä joudutaan kerättyä dataa hieman tarkastelemaan ja mahdollisesti korjaamaan. Tässä vaiheessa kannattaa varmistaa, että kerätty on puutteetonta ja on laadultaan haluttua. Tässä vaiheessa joudutaan mahdollisesti myös jäsenellä dataa, jos se on monesta eri lähteestä kerättyä (Eckerson 2007).

2.1.4 Datan mallinnus

Data-analytiikan prosessin neljännessä ja olennaisimmassa osassa kerätylle datalle luodaan analyysimalli ja algoritmit. Algoritmi voi olla esimerkiksi vertailu, jossa verrataan kahden eri muuttujan käyttäytymistä tietyissä olosuhteissa. Tai yksinkertaisimmillaan algoritmi on yhteenlasku, joka laskee liukuhihnalle menevien kappaleiden määrän tietyssä aikajaksossa (Eckerson 2007).

Ennen analysointimallien luomista, täytyy valita mitä muuttujia analysoidaan. Projektin suuruudesta riippuen valittujen muuttujien määrä voi vaihdella paljonkin, mutta yleensä lopullisessa analyysissä käsiteltäviä muuttujia on alle 100 (Eckerson 2007).

2.1.5 Tulosten käsittely

Data-analyysin viidennessä vaiheessa esitetään analyysin tulokset. Tulokset voivat olla reaaliaikaisia tai historiatietoja. Tulokset on hyvä esittää siinä muodossa, että niistä voidaan ymmärtää tutkittu asia ilman syvempää datatieteen osaamista. Yleisimpiä muotoja tulosten esittelylle on raportit, jotka sisältävät visuaalisia kaavioita ja diagrammeja (TTY Pori 2018).

Valmiin data-analyysin perusteella voidaan miettiä mahdollisia suoritettavia toimenpiteitä tai valmis analyysi voi johtaa myös uuteen data-analyysi prosessiin, jos tuloksiin ei olla tyytyväisiä. On myös mahdollista, että tulosten avulla on opittu analysoitavasta kohteesta enemmän, joka johtaa haluun muuttaa analyysin malleja tai algoritmeja (TTY Pori 2018).

2.1.6 Toteutus

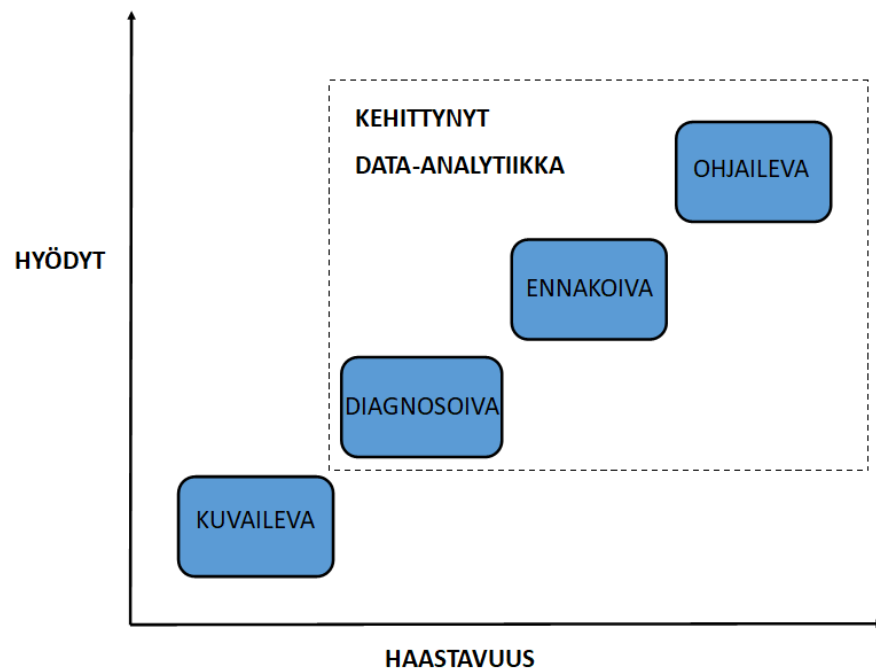
Usein analyysin tulosten perusteella voidaan toteuttaa toimenpiteitä, jolla muutetaan analysoitavasta kohteesta asioita. Toimenpiteitä suorittamalla yritetään yleensä päästä tavoitteisiin, jota data-analytiikalle on aseteltu. Teollisessa tuotannossa tavoitteena voi olla esimerkiksi tuotantolinjan keskeytysten vähentäminen (Shung 2014).

2.1.7 Testaus

Toimenpiteiden jälkeen voidaan data-analyysi tehdä uudestaan samalle kohteelle. Uutta ja vanhaa analyysia vertailemalla voidaan nähdä tehtyjen toimenpiteiden konkreettinen hyöty (Shung 2014).

2.2 Data-analytiikan tasot

On olemassa eri tasoista data-analytiikka. Yleensä se jaotellaan neljään eri perustaan: 1. Kuvaileva analytiikka, 2. Diagnosoiva analytiikka, 3. Ennakoiva analytiikka ja 4. ohjaileva analytiikka (Kuvio 3). Näistä kolme viimeistä kuvaillaan yleensä kehittyneemmäksi data-analytiikaksi, joka vaatii sen toteuttajalta hieman laajempaa ymmärrystä datatieteestä. Kuvaileva data-analytiikka on yksinkertaista ja helposti toteutettavaa, joka ei vaadi toteuttajaltaan erityisosaamista. Myös tässä opinnäytetyössä keskitytään pääsääntöisesti kuvailevan data-analytiikan toteuttamiseen (TTY Pori 2018).



Kuvio 3. Data-analytiikan tasot (TTY Pori 2018, muokattu).

2.2.1 Kuvaileva analytiikka

Kuvaileva analytiikka on data-analytiikan yksinkertaisin taso. Kuvailevassa analytiikassa otetaan kantaa vain jo aiemmin tapahtuneeseen ja se yrittää antaa vastauksen kysymykseen: Mitä tapahtui (Turunen 2016)?

Kuvailevassa analytiikassa käytetään pääsääntöisesti yksinkertaisia laskutoimituksia, joita ovat esimerkiksi keskiarvo, summa ja tulo. Kuvailevan analytiikan tulokset esitetään yleensä visuaalisesti diagrammeina ja kaavioina. Myös tässä opinnäytetyössä suoritettussa tuotantolinjan analyysissä käytetään näitä edellä mainittuja menetelmiä datan käsittelyyn ja sen esittämiseen (Turunen 2016).

2.2.2 Diagnosoiva analytiikka

Data-analytiikan tasoista seuraavaksi vaikein on diagnosoiva analytiikka, joka kertoo, miksi jotain on tapahtunut. Tätä ja kahta seuraavaa analytiikan tasoa kutsutaan yleisesti kehittyneeksi data-analytiikaksi (TTY Pori 2018).

Diagnosoiva analytiikka yrittää selvittää asioiden syy- ja seuraussuhteita. Esimerkiksi teollisuudessa voidaan yrittää selvittää mitkä tapahtumat aiheuttavat hälytyksiä automaatiojärjestelmään. Näitä tapahtumia muuttamalla voidaan mahdollisesti pienentää hälytysten määrää tulevaisuudessa (TTY Pori 2018).

2.2.3 Ennakoiva analytiikka

Ennakoiva analytiikka on analytiikan kolmanneksi haastavin taso, joka keskittyy tulevaisuuteen. Se yrittää vastata kysymykseen: Mitä seuraavaksi tulee mahdollisesti tapahtumaan? Ennakoivassa analytiikassa hyödynnetään menneisyyden dataa, jota tutkimalla voidaan ennustaa tulevaa (Turunen 2016).

Ennakoiva analytiikka tapahtuu luomalla algoritmeja, joilla voidaan ennustaa tulevaisuutta. Tätä tapaa hyödyntämällä voidaan ennustaa esimerkiksi asiakkaiden myyntikäyttäytymistä kaupassa (Turunen 2016).

2.2.4 Ohjaileva analytiikka

Ohjaileva analytiikka on data-analytiikan vaikein ja kehittynein taso. Ohjaileva analytiikka yrittää kertoa kuinka haluttu lopputulos saadaan aikaiseksi. Tätä tasoa käytetään hyvin useasti laitteiden ja tapahtumaketjujen optimointiin (Huttunen 2015).

2.3 Data-analytiikka teollisuudessa

Data-analytiikka voidaan hyödyntää teollisuudessa moniin eri kohteisiin. Käyttökohteina voi olla esimerkiksi päästöjen mittaaminen ja analysointi, huollon tarpeen ennakointi, logistiikan optimointi, laadun tarkkailu sekä automaatiojärjestelmän kehittäminen (TTY Pori 2018).

Vaikka data-analytiikka on ollut jo pitkään käytössä esimerkiksi IT-laitteistoissa ja suuremmissa automaatiojärjestelmissä, ei se teollisuuden ohjelmoitavissa logiikoissa ole vielä kovin yleistä. Tätä puoltaa se, että maailman suurimmat automaatiolaitteiden

toimittajat ovat vasta lähivuosina heränneet tarjoamaan omia välineitään data-analytiikan hyödyntämiseen. Esimerkiksi tässäkin työssä käsiteltävä Beckhoffin TwinCAT Analytics on julkaistu kokonaisuudessaan vasta vuonna 2020.

3 TwinCAT Analytics

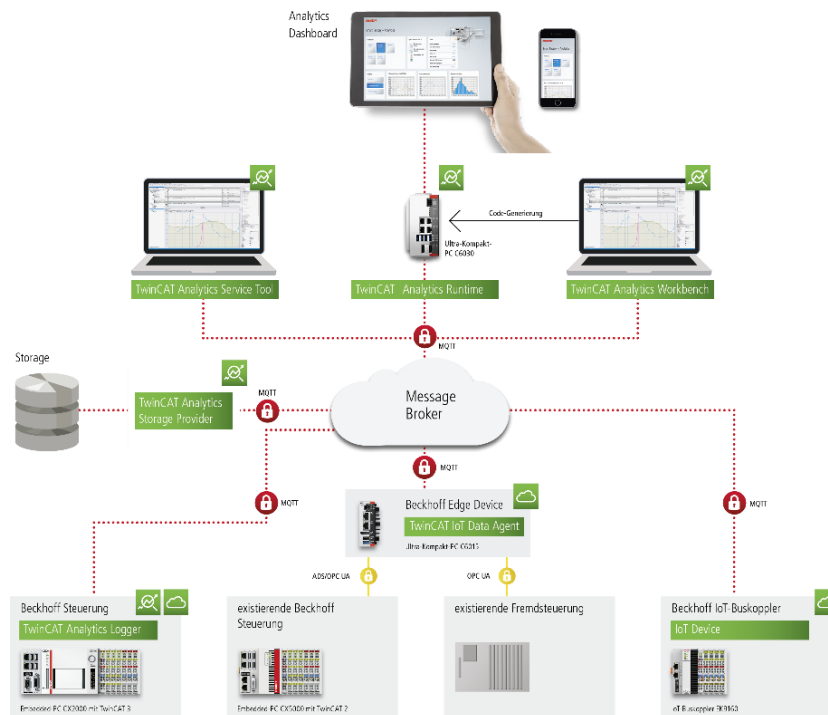
TwinCAT Analytics on Beckhoffin tarjoama tietokoneohjelmisto, jolla voidaan kerätä automaatiojärjestelmästä dataa ja analysoida se. TwinCAT Analytics ei siis ole yksittäinen ohjelma vaan ohjelmistokokonaisuus, jonka eri osia hyödyntämällä voidaan tuotantolinjalta kerätä, tallentaa ja analysoida dataa. Myös analysoinnin tulokset voidaan esittää käyttäjille Beckhoffin omalla TwinCAT HMI -valvomosovelluksella (Beckhoff Automation GmbH & Co. KG 2018).

3.1 Beckhoff Automation GmbH

Beckhoff Automation GmbH saksalainen automaatiolaitetoimittaja, jonka pääkonttori sijaitsee Saksan Verlässä. Beckhoffin päätuote on PC-pohjaiset ohjelmoitavat logiikat. Näitä ohjelmoitavia logiikoita on mahdollista ohjelmoida yrityksen tarjoamalla TwinCAT- tuoteperheellä, johon tässäkin opinnäytetyössä käsiteltävä TwinCAT Analytics kuuluu. Kaikki TwinCAT -ohjelmistot ovat ladattavissa ilmaiseksi heidän kotisivuiltaan. (Beckhoff Automation GmbH & Co. KG 2020).

3.2 TwinCAT Analytics

Kuviossa 4 on esitetty TwinCAT Analyticsin topologia. Kuvan topologiassa on esitetty myös kaikki mahdolliset komponentit, jotka eivät ole välttämättömiä analysoinnin suorittamisessa. Tässä luvussa esitellään topologian komponentit yksilöllisesti.



Kuvio 4. Analytics topologia (Beckhoff Automation GmbH & Co. KG 2018).

3.2.1 Message Broker

TwinCAT Analyticsin keskeinen komponentti on MQTT Message Broker, jota edustaa pilvi keskellä kuvion 6 topologiaa. Message Brokerin tehtävä on välittää MQTT-protokollan mukaista dataa topologian laitteiden välillä. Valitettavasti Beckhoff ei tarjoa omaa Message Brokeria, mutta internetistä on saatavilla usean eri valmistajan tarjoamia vaihtoehtoja ilmaiseksi. Message Broker voi sijaita millä tahansa topologian PC:llä tai vaihtoehtoisesti omalla itsenäisellä PC:llä (Beckhoff Automation GmbH & Co. KG 2018).

3.2.2 Analytics Logger

Analytics Loggerin tehtävänä on kerätä ja lähettää MQTT-protokollan mukaista dataa ohjelmitavilta logiikoilta Message Brokerille, josta ne välitetään eteenpäin analysoitavaksi. Analytics Logger konfiguroidaan tuotantolinjaa ohjaaviin logiikoihin. Konfiguroinnissa määritellään käytettävä Message Broker sekä muuttujat, jotka halutaan lä-

hettää analysoitavaksi. Konfiguroinnissa on myös mahdollista määrittellä lähetettävien datapakettien koko ja puskurit. Eli käytännössä voidaan todeta, että TwinCAT Analytics Logger hoitaa puolet data-analytiikan prosessin toisesta vaiheesta eli datan keräämisen (Beckhoff Automation GmbH & Co. KG 2018).

3.2.3 Analytics Workbench

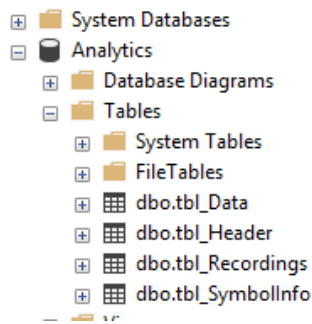
Analytics Workbench on sovellus, johon luodaan analyysin algoritmit. Ohjelma tarjoaa kymmenittäin valmiita algoritmeja, joita on helppo käyttää ilman vahvempaa ohjelmoinnin osaamista. Analytics Workbenchilla hoidetaan data-analytiikan prosessin neljäs vaihe eli Datan mallintaminen ja algoritmien määrittäminen. Kun mallintaminen on valmis, voidaan Workbenchilla tehdyt algoritmit generoida ohjelmaksi ja ladata se tietokoneelle tai ohjelmoitavalla logiikalle. Tätä analysointia suorittavaa laitetta kutsutaan kuvion 6 topologiassa nimellä Analytics Runtime (Beckhoff Automation GmbH & Co. KG 2018).

3.2.4 Analytics Storage Provider

Analytics Storage Provider eli ASP on palvelu, joka hoitaa datan tallennuksen data-analytiikan prosessissa eli puolet vaiheesta 2. ”Datan hankinta”. Storage Provider voidaan konfiguroida tallentamaan dataa tietokantaan tai tietokoneen paikalliselle kovalevylle. Datan tallentaminen ei ole välttämätöntä, jos analyysi halutaan suorittaa vain reaaliajassa, eikä historiallisten tapahtumien analysoinnille ole tarvetta (Beckhoff Automation GmbH & Co. KG 2018).

Analytics Storage Provider on yhteensopiva vain Microsoftin SQL Serverin, sekä pilvipalveluna toimivan Microsoft Azuren kanssa. Storage Providerin etuna on, että se hoitaa itse tietokannan käsittelyn, eikä käyttäjän tarvitse, kuin luoda uusi tietokanta ja yhdistää Storage Provider siihen. Kun yhteys on luotu ja ensimmäinen tallennus on käynnistetty Recorderista, ilmestyy luotuun tietokantaan neljä taulua, jotka sisältävät Recorderin tallentaman datan (Kuvio 5). Storage Providerin huonona ominaisuutena

voidaan pitää sen kykyä tallentaa dataa vain binäärimuodossa, joka tarkoittaa käytännössä, että dataa voidaan käsitellä vain Beckhoffin omilla työkaluilla (Beckhoff Automation GmbH & Co. KG 2018).

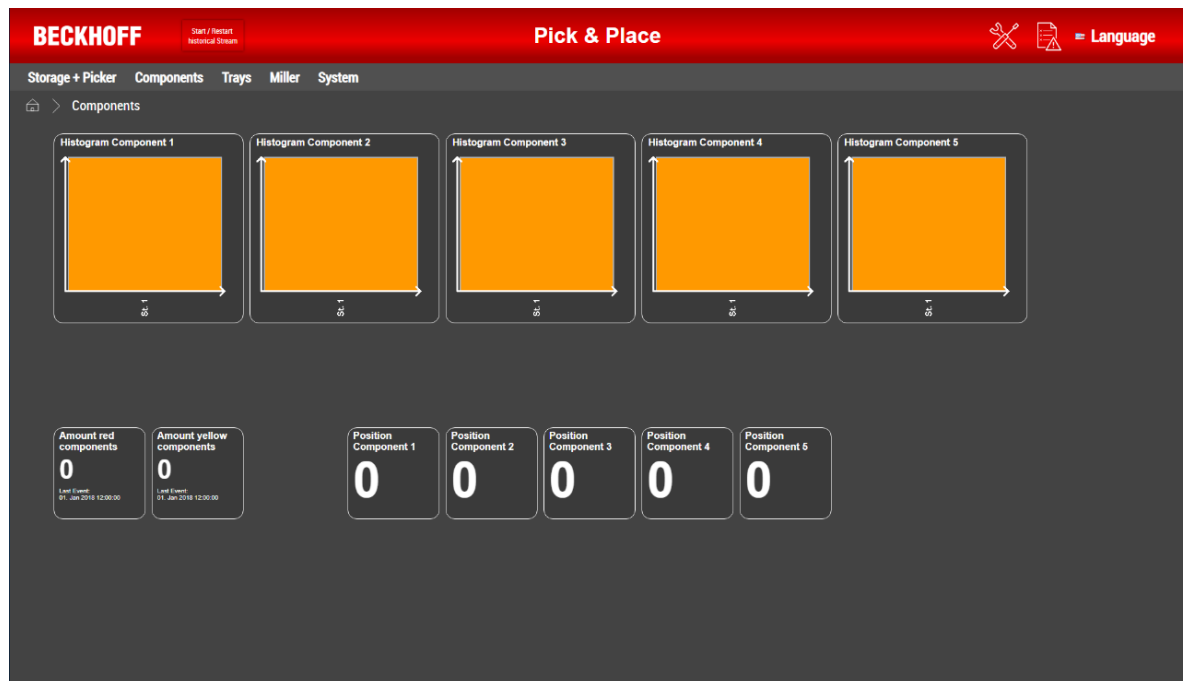


Kuvio 5. Tietokannan rakenne.

3.2.5 Analytics Dashboard

Analysoinnin tulokset voidaan esittää Analytics Dashboardissa, joka on käytännössä valvomo ja samalla analysoinnin tulosten esityspaikka. Valvomo voidaan luoda käyttämällä Beckhoffin omaa valvomosovellusta TwinCAT HMI. TwinCAT HMI on valvomosovellus, joka hyödyntää HTML5-merkintäkieltä. HTML5 on yleisesti verkkosivujen luomisessa hyödynnetty merkintäkieli, mikä tarkoittaa käytännössä, että valmis valvomo voidaan niin sanotusti julkaista tehtaan sisäiseen automaatioverkkoon. Julkaisun jälkeen millä tahansa verkkoon kytkeytyvällä laitteella voidaan operoida valvomoa. Valvomo tarvitsee toimiakseen myös TwinCAT HMI Server -ohjelmiston, joka ylläpitää valvomon verkkosivua (Beckhoff Automation GmbH & Co. KG 2018).

Valvomoon voidaan luoda trendejä, histogrammeja ja muita visuaalisia elementtejä kuvailemaan analysoinnin tuloksia. Valvomoon on myös mahdollista liittää monen eri analysoinnin tulokset. Tämä mahdollistaa kaikkien tehtaan tuotantolinjojen lisäämisen samalle valvomonäytölle. Valvomon kuvat ja sisällön voi luoda joko itse manuaalisesti tai käyttämällä automaattista generointia (Kuvio 6). Tämä onnistuu samalla toimenpiteellä, kuin luvussa 3.2.3 esitelty ohjelman generoiminen (Beckhoff Automation GmbH & Co. KG 2018).



Kuvio 6. Generoitu Analytics Dashboard.

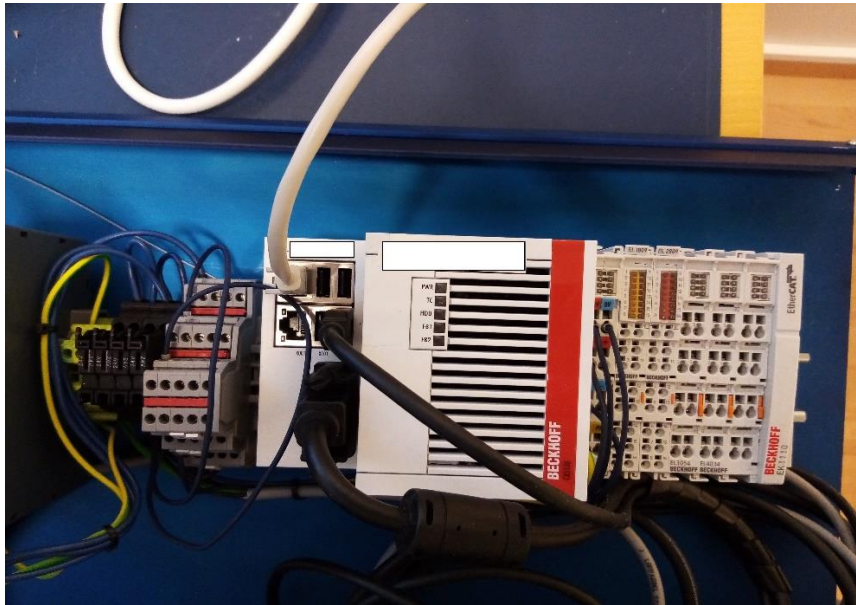
4 Ohjelmiston toimintojen ja ominaisuuksien testaus

Tässä luvussa käydään läpi testiprojektin vaiheet, jossa TwinCAT Analyticsia testattiin toimistotiloihin luodulla testikokoonpanolla. Testauksen tarkoituksena oli selvittää ohjelmiston ominaisuuksien toimivuus sekä tutkia sen soveltuvuutta tulevaisuuden asiakasprojekteihin.

4.1 Laitteisto

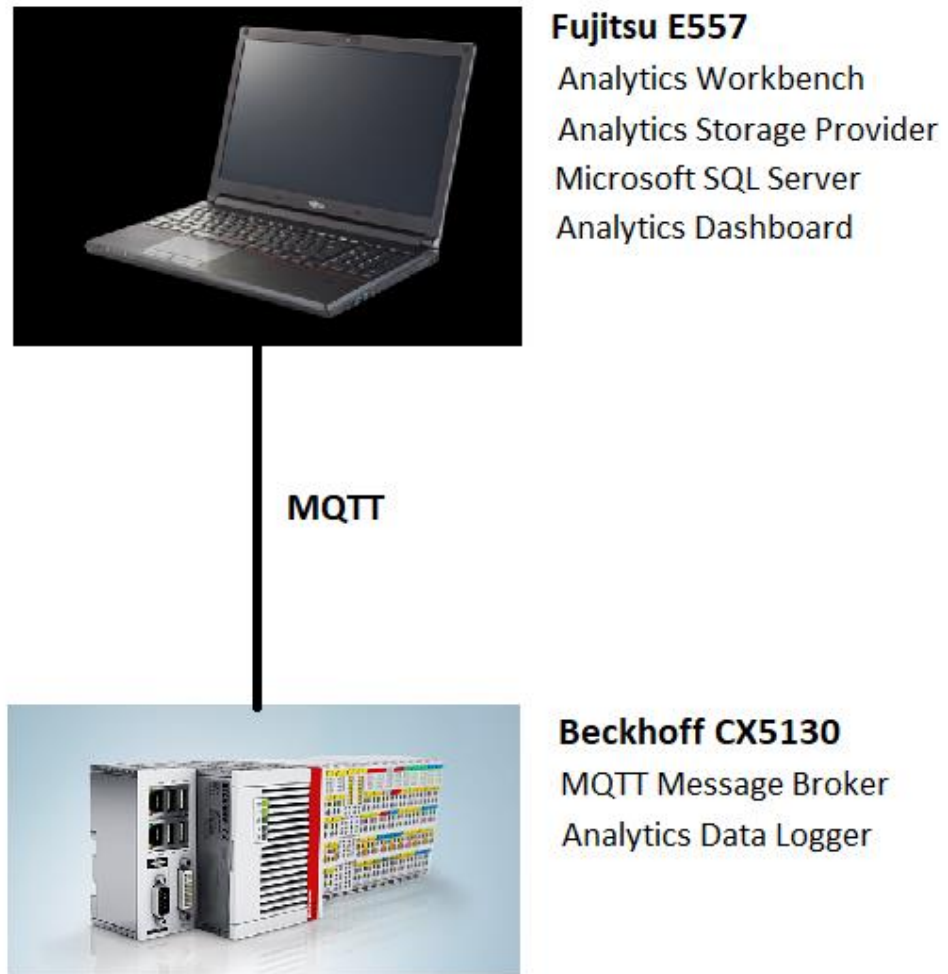
Ennen varsinaista testausta täytyi valita käytettävä laitekanta eli analysoinnin topologian komponentit. Ensimmäisenä täytyi päättää mihin Analytics Workbenchilla luotu analysointiohjelma generoitaisiin. Tähän tarkoitukseen sopivaksi osoittautui ohjelmointi käytössä ollut Windows 10 -käyttöjärjestelmällä toimiva Fujitsun kannettava tietokone. Toiseksi pääkomponentiksi valikoitu toimiston yleisessä testauskäytössä ollut CX5130 ohjelmoitava logiikka (Kuvio 7). Tämä logiikka konfiguroitiin lähettä-

mään simuloituja mittauksia ja tilatietoja analysointiohjelmalle. Tähän samaan laitteeseen asennettiin myös MQTT Message Broker, joka välittäisi datan verkon yli. Tiedon tallentamiseen tarkoitettu Storage Provider ja sen vaatima tietokanta sijoitettiin analysointia suorittavalle kannettavalle tietokoneelle.



Kuvio 7. CX5130 ohjelmoitava logiikka.

Kuviossa 8 on nähtävillä periaatekuva analysoinnin topologiasta. Kuvan alaosassa on näkyvillä CX5130 -ohjelmoitava logiikka, jonka voidaan ajatella mallintavan tehtaän tuotantolinjaa ohjaavaa ohjelmoitavaa logiikkaa. Logiikka välittää Message Brokerin avulla MQTT -protokollan mukaista mittaustietoa tuotantolinjan tilasta analysointia suorittavalle tietokoneelle, joka on tässä kokoonpanossa kuviossa 8 yläosassa sijaitseva Fujitsun E557 kannettava tietokone. Samassa tietokoneessa analysoinnin tulokset esitetään visuaalisesti Analytics Dashboardissa. Topologiaan olisi voinut lisätä useammankin ohjelmoitavan logiikan, jotka olisivat lähettänyt dataa analysoitavaksi, mutta testauksen yksinkertaisena pitämiseksi päädyttiin vain yhteen logiikkaan.



Kuvio 8. Testikokoonpanon periaatekuva.

4.2 Datan kerääminen ja tallentaminen

4.2.1 Muuttujien valinta

Ennen varsinaista datan keräämistä ja tallentamista täytyi valita muuttujat, joita halutaan analysoida. Tämä vaihe on periaatteessa jo aiemmin esitellyn data-analytiikan prosessin ensimmäinen vaihe eli kysymysten asettelu sekä osittain myös saman prosessin vaihe 2 eli datan hankinta. Tässä vaiheessa normaalissa analysointiprojektissa määritellään se mitä analysoinnilla halutaan saavuttaa, mutta koska testaustilanteissa ei ole käytössä todellista prosessia tai tuotantolinjaa, täytyy testaajan miettiä millaisilla muuttujilla saisi mallinnettua todellista tilannetta.

Tässä testauksessa haluttiin mallintaa kappaletavaruotannon tuotantolinjaa, joten testausta varten luotiin neljä erilaista muuttujaa, jotka ovat tyypillisiä tuotantolinjalta saatavaa mittausta tai tilatietoa. Taulukkoon 1 on kerätty lista simuloituista muuttujista. Taulukon vasemmalla puolella on näkyvissä muuttujan nimi ja oikealla puolella sen tietotyyppi.

Taulukko 1. Analysoitavat muuttujat.

Muuttujan nimi	Tietotyyppi
Vikatila	Boolean
Kappalelaskuri	Boolean
Moottorin lämpötila	Real
Linjan nopeus	Real

Taulukon ensimmäisenä muuttujana on vikatila, jonka tietotyyppinä on boolean.

Tällä muuttujalla haluttiin simuloida tuotannon vikatilaa. Vikatilasta haluttiin selvittää kokonaisaika, jolloin se on aktiivisena esimerkiksi yhden työvuoron aikana.

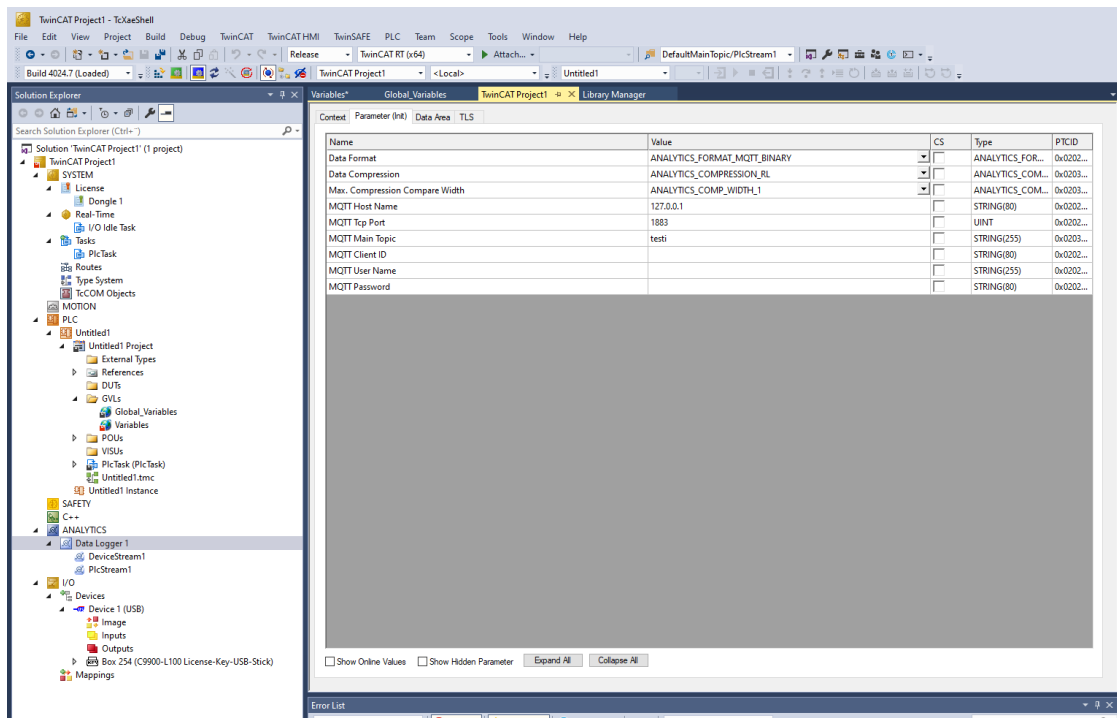
Toiseksi muuttujaksi valittiin kappalelaskuri, jolla laskettiin tuotettujen kappaleiden kokonaismäärä. Moottorin lämpötilasta haluttiin selvittää aika, jolloin moottorin lämpötila on liian korkea. Linjan nopeudesta tarkasteltiin sen maksimi, minimi ja keskiarvonopeutta. Näitä neljää muuttujaa varten luotiin simulaatio-ohjelma ohjelmoitavalle logiikalle, joka yrittää mallintaa niiden käyttäytymistä todellisessa tuotantolinjassa.

4.2.2 MQTT Message Broker

Testikokoonpanon MQTT Message Brokeriksi valittiin Eclipse Foundationin tarjoama Mosquitto -sovellus, koska Beckhoffin tekninen tuki kertoi heidän todenneen sen toimivaksi omissa testauksissaan. Mosquitto Message Broker on täysin ilmainen sovellus ja se on ladattavissa heidän omilta kotisivuiltaan. Sovelluksen käyttöönotto on tehty yksinkertaiseksi ja nopeaksi. Asennuksen jälkeen ohjelma pitää vain käynnistää Windowsin komentokehotteesta yhdellä komennolla. Käynnistyksen jälkeen sovellus kuuntelee oletuksena tietoliikenneporttia 1883. Portti on vaihdettavissa, mutta tässä testausympäristössä käytettiin vain kyseistä oletusporttia.

4.2.3 Data Loggerin konfigurointi

Data Loggerin konfigurointi ohjelmoitavaan logiikkaan tapahtuu TwinCAT 3 -ohjelmointityökalulla. Data Loggerin saa lisättyä ohjelmaan valitsemalla hiiren oikealla painikkeella Solution Explorerista ANALYTICS ja Add Data Logger. Toimiakseen Data Logger vaatii oikeat parametrit. Parametrit ovat määriteltävissä Data Loggerin sivulta Parameter (Init). Kuviossa 9 on nähtävillä tässä testausprojektissa käytetyt parametrit. Huomioitavaa on, että kun Message Broker sijaitsee samalla laitteella kuin Data Logger, on MQTT Host Name -valintaan sijoitettava paikallinen IP-osoite eli 127.0.0.1.



Kuvio 9. Data Loggerin parametrit.

Ennen kuin Data Logger alkaa lähettämään dataa Message Brokerille on halutut muuttujat ensin valittava lähetettäväksi. Valinta tapahtuu Solution Explorerista löytävästä PlcStream -valikosta. Valikosta löytyvät kaikki muuttujat, jotka ovat määritelty muuttujalistassa analysointimuuttujiksi komennolla {attribute 'TcAnalytics'} (Kuvio 10).

```

{attribute 'qualified_only'}
VAR_GLOBAL

{attribute 'TcAnalytics'}
Vikatila :BOOL;

{attribute 'TcAnalytics'}
Kappalelaskuri :BOOL;

{attribute 'TcAnalytics'}
Moottorin_lampotila :REAL;

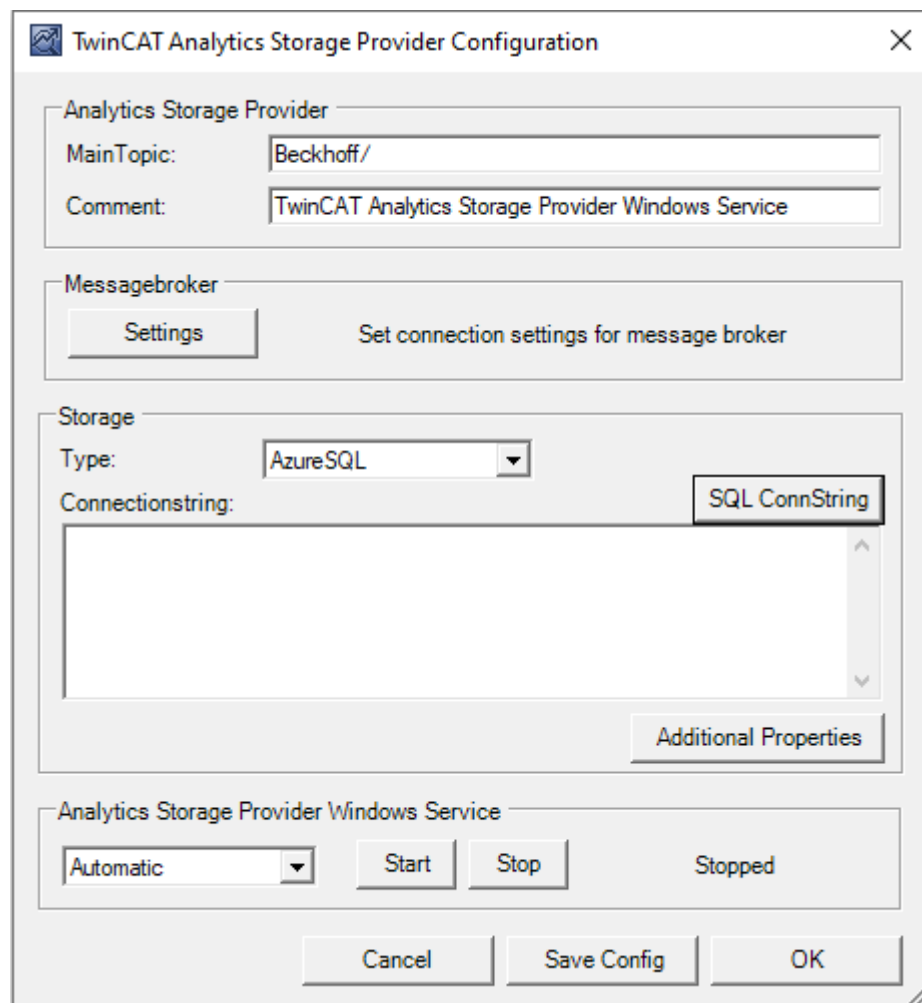
{attribute 'TcAnalytics'}
Linja_nopeus :REAL;

```

Kuvio 10. Analysointimuuttujien määrittely.

4.2.4 Storage Providerin konfigurointi

Tiedon tallentamiseen tarkoitettu Storage Provider konfiguroidaan ja otetaan käyttöön omassa sovelluksessaan. Kuviossa 11 on kuvakaappaus sovelluksen käyttöliittymästä. Käyttöliittymän Messagebroker -osion sisällä olevasta Settings -painikkeesta saadaan valittua käytettävä MQTT Message Broker, josta tallennettavaa dataa kerätään. SQL ConnString -painikkeesta voidaan määritellä haluttu sijoituspaikka tallennettulle datalle.



Kuvio 11. Analytics Storage Providerin käyttöliittymä.

Storage Provider ei automaattisesti tallenna kaikkea saatavilla olevaa dataa, vaan käyttäjän on käytävä asettelemassa Storage Provider Recorderista halutut muuttujat tallennukseen. Recorderin käyttöliittymä on löydettävissä Analytics Workbenchin etusivulta. Datan tallentamisessa täytyy ottaa huomioon tietokannan tai kovalevyn

koko, koska ne voivat täyttyä liiasta datan määrästä. Tämän ongelman voidaan ratkaista määrittelemällä data tallentumaan vain tietyksi ajanjaksoksi. Esimerkiksi pus-kuri voidaan määritellä 30 päiväksi, joka tarkoittaa käytännössä, että kaikki yli 30 päivää vanha data poistetaan muistista automaattisesti.

4.3 Datan mallintaminen

TwinCAT Analytics -ympäristössä datan mallintaminen tapahtuu hyödyntäen Analytics Workbenchissa olevia valmiita algoritmeja. Tässä alaluvussa käymme läpi Workbenchin tärkeimpiä ominaisuuksia.

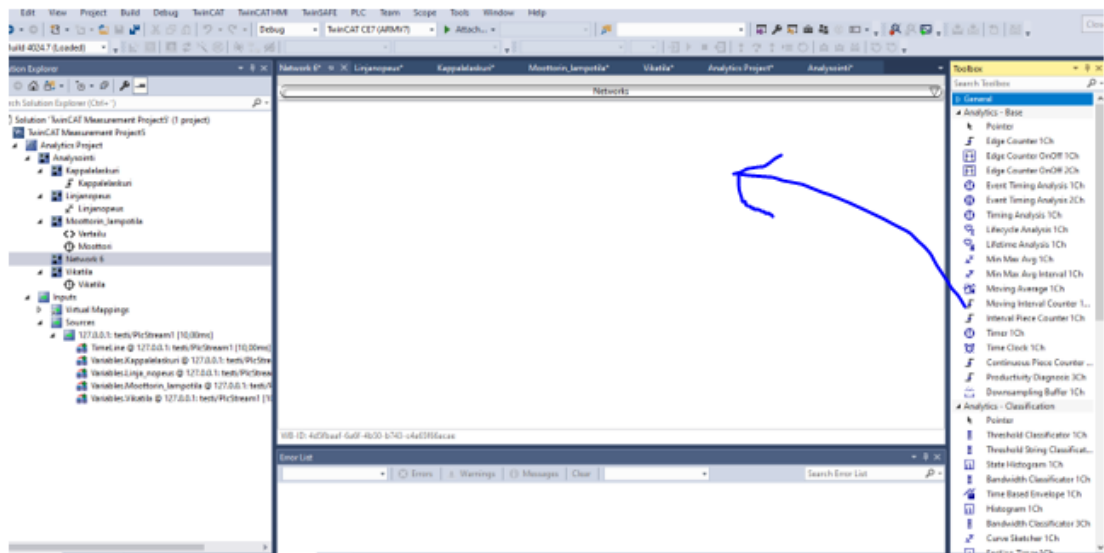
4.3.1 Muuttujien tuominen Workbenchiin

Analytics Workbenchilla voidaan käsitellä reaaliaikaisia tai historiallisia muuttujia. Molempien tuonti analysoitavaksi onnistuu TwinCAT Target Browserista. Target Browserin TcAnalytics -välilehdeltä löytyvät muuttujat voidaan yksinkertaisesti hiirellä vetämällä lisätä projektin Inputs -valikkoon. Lisäämisen jälkeen niitä voidaan käyttää analysointialgoritmien sisääntuloina.

4.3.2 Algoritmien lisääminen analysointiprojektiin

Analytics Workbenchissa on tarjolla useita kymmeniä valmiiksi luotuja analysointialgoritmeja. Algoritmien käyttö on luotu helpoksi eikä niiden käyttö vaadi analysoinnin tekijältä vankkaa ohjelmoinnin osaamista. Pääsääntöisesti tarjotut algoritmit ovat yksinkertaisia kuvailevan analytiikan malleja, mutta niitä yhdistelemällä on mahdollista saada hyvinkin arvokasta tietoa analysoitavien muuttujien käyttäytymisestä.

Kaikki käytettävät algoritmit ovat näkyvillä Workbenchin Toolboxissa. Sieltä algoritmien lisääminen projektiin onnistuu hiirellä vetämällä haluamansa Networkin päälle (Kuvio 12).



Kuvio 12. Algoritmien lisääminen analysointiprojektiin.

4.3.3 Testausprojektin algoritmit

Testausta varten analysointiprojektiin luotiin jokaiselle muuttujalle omat yksilölliset algoritmit.

Vikatila

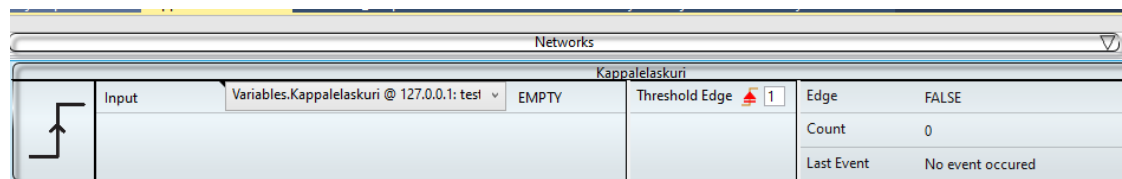
Vikatila -muuttujan analysoinnissa haluttiin ensisijaisesti selvittää kokonaisaika, jolloin muuttuja on aktiivisena. Tähän tarkoitukseen sopivaksi algoritmiksi osoittautui Event Timing Analysis, joka kertoo kokonaisajan lisäksi myös minimi-, maksimi ja keskiarvoajat (Kuvio 13).

Input		Variables.Vikatila @ 127.0.0.1: testi/PlcSti	EMPTY	Vikatila	
	Threshold Edge On		1	Is On	FALSE
	Threshold Edge Off		1	Current Interval	00:00:00:000
				On Min	00:00:00:000
				On Max	00:00:00:000
				On Avg	00:00:00:000
				On Total	00:00:00:000
				Off Min	00:00:00:000
				Off Max	00:00:00:000
				Off Avg	00:00:00:000
				Off Total	00:00:00:000
			Count On	0	

Kuvio 13. Event Timing Analysis -algoritmi Analytics Workbenchissa.

Kappalelaskuri

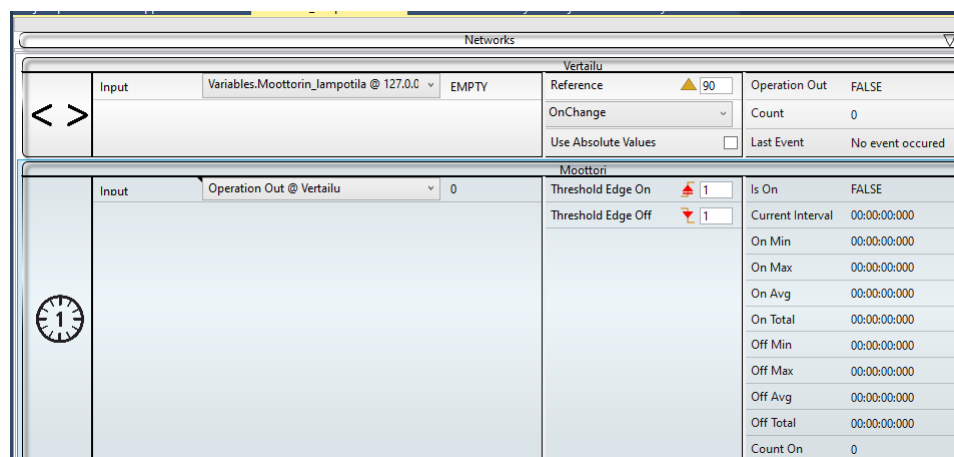
Kappalelaskurin tavoitteena oli mallintaa tuotantolinjan kappaleiden laskentaa. Kyseessä on hyvin yksinkertainen matemaattinen laskutoimitus, mutta silti todella hyödyllinen väline kappalevaratuotannon analysoimisessa. Tähän tarkoitukseen sopivaksi algoritmiksi valikoitui Edge Counter (Kuvio 14). Kyseinen algoritmi laskee kokonaislukuna, kuinka monta kertaa valittu muuttuja on mennyt aktiiviseksi.



Kuvio 14. Edge Counter -algoritmi Analytics Workbenchissa.

Moottorin lämpötila

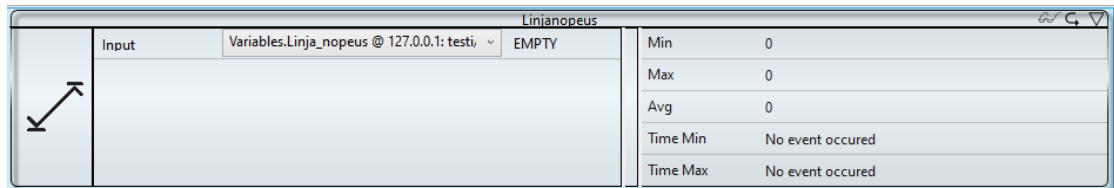
Moottorin lämpötilasta haluttiin selvittää kokonaisaika, jolloin moottorin lämpötila ylittää asetellun raja-arvon. Ennen kokonaisajan analysointia tarvittiin tehdä vertailu, joka kertoo milloin lämpötila ylittää asetellun raja-arvon. Vertailun ulostulo analysoitiin Event Timing Analysis -algoritmilla, jolla saatiin selville ylityksen kokonaisaika. Kuviossa 15 on nähtävillä, että vertailun ulostuloa on käytetty suoraan Event Timing Analysis -algoritmin sisääntulona.



Kuvio 15. Moottorin lämpötilan analysoinnin algoritmit.

Linjanopeus

Linjanopeus -muuttujalla mallinnettiin mittausta, joka kertoo tuotantolinjan liukuhihnan nopeuden. Tästä nopeudesta haluttiin analysoinnilla selvittää sen maksimi-, mini- ja keskiarvonopeudet. Kyseiseen tarkoitukseen täydellinen algoritmi löytyi Workbenchin Toolboxista nimellä Min Max Avg (Kuvio 16).



Kuvio 16. Min Max Avg -algoritmi Analytics Workbenchissa.

4.3.4 Analysointimallinnuksen generoiminen

Kun halutut algoritmit on saatu lisättyä projektiin ja mallinnuksen toimivuus on testattu, voidaan tehdystä mallinnuksesta generoida ohjelma. Generoitu ohjelma voidaan ladata TwinCAT 3 -ohjelmointityökalulla Analytics Runtimelle, joka jää suorittamaan analysointia tuotannon taustalle. Ohjelma generoidaan Deploy Runtime -työkalulla, joka on avattavissa Analytics Workbenchin etusivulta. Ohjelman generoinnin yhteydessä luodaan myös Analytics Dashboard, jossa analysoinnin tulokset näytetään. Dashboard sisältää jokaisesta algoritmista luodun oman yksilöllisen Controllerin, jotka esittävät algoritmien tulokset visuaalisesti. Ennen ohjelman generoimista on analysoinnin tekijän määriteltävä algoritmien asetuksista haluamansa Controller. Vaikka Dashboard luodaan automaattisesti, on se generoinnin jälkeen vielä täysin käyttäjän muokattavissa TwinCAT HMI Engineering -työkalulla.

4.4 Analysoinnin tulosten esittäminen

Testausprojektissa Analytics Dashboard luotiin generoinnin avulla. Kuviossa 17 on esitetty testiprojektiin valittujen algoritmien Controllerit. Kuvion oikeassa reunassa on nähtävillä Event Timing Analysis -Controller, joka näyttää käyttäjälleen visuaali-

sesti algoritmin tulokset. Kuvan keskellä on Edge Counter, joka yksinkertaisesti ilmaisee laskettujen kappaleiden määrän. Linjanopeuden esittämiseen käytetty Min Max Avg -algoritmin Controller on kuvion oikeassa reunassa.



Kuvio 17. Algoritmien Controllerit Analytics Dashboardissa.

4.5 Testaaminen

Testiprojektia testattiin simuloimalla logiikalta lähetettävien muuttujien arvoja vastaan todellista tuotantolinjan ajotilannetta. Simuloinnin aikana todettiin, että luodut algoritmit näyttävät oikein ja toimivat muutenkin luvutulla tavalla Analytics Dashboardissa. Myös Storage Providerilla tietokantaan tallennettua dataa analysoitiin ja sekin todettiin toimivaksi.

5 Tulokset

Opinnäytetyön tavoitteena oli tutustua TwinCAT Analytics -analytiikkatyökalun ominaisuuksiin sekä selvittää sen soveltuvuus tulevia asiakasprojekteja varten. Alkuperäisenä suunnitelmana oli testata työkalua myös asiakkaan tuotantolinjan analysoimisessa, mutta aikataulullisista syistä ohjelmiston ominaisuuksien testaus kerrettiin suorittamaan vain toimisto-olosuhteissa. Onnistuneen testausprojektin ansiosta työkalua tullaan tulevaisuudessa kokeilemaan oikean tuotantolinjan analysoimisessa, jolloin työkalun hyödyllisyydestä voidaan tehdä laajempia arvioita.

Opinnäytetyön tietoperustan keräämisen aikana saatiin luotua vahva ymmärrys siitä mitä data-analytiikan sovelluksella täytyy pystyä tekemään sekä mitä ominaisuuksia siltä voidaan vaatia. Tätä ymmärrystä hyödyntäen voitiin päätellä, että TwinCAT Analytics on soveltuva ainakin yksinkertaiseen kuvailevaan tuotantolinjojen analytiikkaan, koska sen ominaisuuksia hyödyntäen voidaan tässä opinnäytetyössä esitelty data-analytiikan prosessi suorittaa teollisuuden tuotantolinjoille.

Työn konkreettisena tuloksena saatiin myös laadittua opinnäytetyön toimeksiantaja Apex Automation Oy:lle kattava operointiohje työkalun käytöstä sekä sen tarjoamista ominaisuuksista. Operointiohjeen tarkoituksena on vähentää tuleviin analysointiprojekteihin menevää aikaa sekä parantaa toimeksiantajan kykyä kertoa asiakkailleensa yksityiskohtaisempaa tietoa ohjelmiston tarjoamista mahdollisuuksista.

6 Pohdinta

Onnistumisten lisäksi opinnäytetyön tekemisessä oli myös paljon haasteita. Muutamit tekniset ongelmat hidastivat työn suorittamista merkittävästi, mutta Beckhoffin tarjoaman tuen ansiosta ongelmista selvittiin ja ohjelmiston testaus saatiin suoritettua onnistuneesti.

Vaikka TwinCAT Analytics on toimiva ja soveltuva ohjelmisto tuotantolinjojen analytiikassa, on siinä silti yksi merkittävä ongelma. Ongelmana on datan tallennus tietokantaan, koska data voidaan tallentaa vain binäärimuodossa, joka käytännössä tarkoittaa, ettei kerättyä dataa tai analysoinnin tuloksia voida käsitellä missään muualla kuin Analytics Workbenchissa tai Dashboardissa. Mutta täytyy kuitenkin muistaa, että TwinCAT Analytics on vasta uusi ohjelmisto, joten sitä tullaan todennäköisesti päivittämään ja parantamaan todella paljon ensimmäisten käyttövuosien aikana. Mielenkiintoinen lisä ohjelmiston ominaisuuksiin olisi myös mahdollisuus luoda Excel-raportteja analysoinnin tuloksista. Raporttien avulla tulosten käsittely olisi helpompaa, koska niihin voitaisiin kerätä selvemmin tietoa esimerkiksi yhden työvuoron aikana tapahtuneista asioista.

TwinCAT Analyticsin tarkempi soveltuvuuden tarkastelu vaatisi ehdottomasti laajempien ja pidemmän aikavälin testauksien suorittamista oikeassa teollisuuden tuotantolinjassa sekä mahdollisesti myös käyttäjältään syvempää data-analytiikan ymmärtämistä. Mutta yhteenvetona uskallan kuitenkin todeta, että TwinCAT Analytics on varsin hyvä lisä Beckhoffin laajaan ja monipuoliseen TwinCAT -tuoteperheeseen sekä pienten parannusten jälkeen oivallinen väline tuotantolinjojen analysoimiseen.

Lähteet

- Beckhoff Automation GmbH & Co. KG. 2018. Creating value: IoT and Data Analytics from Beckhoff. Viitattu 8.5.2020.
https://download.beckhoff.com/download/document/catalog/Beckhoff_IoT_Data_Analytics_en.pdf
- Beckhoff Automation GmbH & Co. KG. 2020. The New Automation Technology Advantage. Viitattu 8.5.2020.
https://download.beckhoff.com/download/document/catalog/The_New_Automation_Technology_Advantage_e.pdf
- Bertolucci, J. 2013. Big Data Analytics: Descriptive Vs. Predictive Vs. Prescriptive. *InformationWeek*. Viitattu 10.5.2020. <https://www.informationweek.com/big-data/big-data-analytics/big-data-analytics-descriptive-vs-predictive-vs-prescriptive/d/d-id/1113279>
- Huttunen, S. 2015. Kehittynyt data-analytiikka vähittäiskaupan alalla. Pro gradu-tutkielma. Itä-Suomen Yliopisto, Tietojenkäsittelytieteen laitos. Viitattu 22.4.2020.
https://epublications.uef.fi/pub/urn_nbn_fi_uef-20160089/urn_nbn_fi_uef-20160089.pdf
- Kananen, J. 2015. Kehittämistutkimuksen kirjoittamisen käytännön opas: miten kirjoittaa kehittämistutkimuksen vaihe vaiheelta. Jyväskylä: Jyväskylän ammattikorkeakoulu.
- Laney, D. 2015. The Gartner Analytic Continuum. Viitattu 15.5.2020.
https://twitter.com/doug_laney/status/611172882882916352.
- Rouse, M. 2019. data analytics (DA). Viitattu 12.4.2020.
<https://searchdatamanagement.techtarget.com/definition/data-analytics>
- Shung, K. P. 2014. Stages in the Analytics/Data Science Process. Viitattu 22.4.2020.
<https://www.linkedin.com/pulse/20140822130955-4710606-stages-in-the-analytics-data-science-process>.
- TTY Pori. 2018. Selvitys data-analytiikan nykytilasta ja data-analytiikan hyödyntämisestä satakunnassa. Viitattu 12.4.2020. <http://www.datatiede.fi/wp-content/uploads/2018/10/Data-analytiikan-selvitys-Julkaisuversio-2-2018-10-30.pdf>
- Turunen, W. 2016. Datatiede verkko-opetuksessa. Pro gradu-tutkielma. Itä-Suomen Yliopisto, Tietojenkäsittelytieteen laitos. Viitattu 22.4.2020.
https://epublications.uef.fi/pub/urn_nbn_fi_uef-20161048/urn_nbn_fi_uef-20161048.pdf
- Wayne W. Eckerson. 2007. Predictive Analytics: Extending the Value of Your Data Warehouse Investment. Viitattu 11.5.2020.
https://www.researchgate.net/profile/Waranpong_Boonsiritomachai/publication/307571066_Exploring_business_intelligence_and_its_depth_of_maturity_in_Thai_SME

s/links/585017a708ae4bc8993b6696/Exploring-business-intelligence-and-its-depth-of-maturity-in-Thai-SMEs.pdf.