

2017 • 2018
Faculteit Industriële ingenieurswetenschappen
master in de industriële wetenschappen: energie

Masterthesis

Praktische IIoT case

PROMOTOR :

ing. Geert LEEN

PROMOTOR :

ing. Stijn VAN UYTFANGHE

Glen Thoelen

Scriptie ingediend tot het behalen van de graad van master in de industriële wetenschappen: energie,
afstudeerrichting automatisering

De transnationale Universiteit Limburg is een uniek samenwerkingsverband van twee universiteiten in twee landen: de Universiteit Hasselt en Maastricht University.



Universiteit Hasselt | Campus Diepenbeek | Agoralaan Gebouw D | BE-3590 Diepenbeek
Universiteit Hasselt | Campus Hasselt | Martelarenlaan 42 | BE-3500 Hasselt



2017 • 2018

Faculteit Industriële ingenieurswetenschappen
master in de industriële wetenschappen: energie

Masterthesis

Praktische IIoT case

PROMOTOR :

ing. Geert LEEN

PROMOTOR :

ing. Stijn VAN UYTFANGHE

Glen Thoelen

Scriptie ingediend tot het behalen van de graad van master in de industriële wetenschappen: energie,
afstudeerrichting automatisering



KU LEUVEN

Woord vooraf

De masterproef is afgelegd bij Reditech engineering gedurende de periode van 11 september 2017 tot 2 februari 2018. Als masterproef werd een onderzoek gedaan in opdracht van Cooltech naar de haalbaarheid van een centraal platform waaruit Cooltech alle remote installaties kan monitoren en bijsturen.

Ten eerste wil ik mijn externe promotor de heer ing. S. Van Uytfanghe willen bedanken voor de goede begeleiding en feedback doorheen de hele stage. Verder zou ik graag de heren W. Van Soom en W. Franssen willen bedanken om mij de kans te geven dit eindwerk af te leggen binnen hun bedrijf.

Ten tweede zou ik graag de heren S. Poortmans en J. Vanharen van Cooltech willen bedanken om me de kans te geven het project voor hun bedrijf uit te voeren en voor de goede begeleiding en ondersteuning doorheen het eindwerk.

Vervolgens wil ik mijn interne promotor de heer ing. G. Leen bedanken voor de nodige bijstand en advies tijdens het eindwerk. Ook E. Hermans wil ik graag bedanken voor het ontwerpen van de nodige grafische afbeeldingen en poster voor het eindwerk.

Ten slotte wil ik nog het team van Reditech engineering bedanken. Zij hebben me enorm geholpen mijn vragen en bepaalde adviezen gegeven.

Inhoudsopgave

Woord vooraf	1
Lijst van Tabellen	7
Lijst van Figuren	9
Verklarende woordenlijst	11
Abstract	13
Abstract in English	15
1 Inleiding	17
1.1 Situering	17
1.2 Probleemstelling	17
1.3 Doelstellingen.....	18
1.4 Methode.....	19
2 Literatuurstudie	21
2.1 Tijdslijn veranderingen in de industrie.....	21
2.2 IoT	22
2.2.1 IoT architectuur	23
2.3 IIoT	24
2.3.1 OT + IT	25
2.4 IoT vs. IIoT	26
2.5 MES.....	27
2.6 IIoT vs. MES.....	28
2.7 Cloud/Edge computing.....	29
2.8 IIoT platformen.....	29
2.9 IIoT-protocollen	30
2.9.1 Client/Server vs. Publish/Subscribe	30
2.9.2 OPC UA.....	31
2.9.3 HTTP (REST/JSON)	32
2.9.4 MQTT.....	32
2.9.5 CoAP	33
2.9.6 DDS.....	33
2.9.7 AMQP.....	33

2.9.8	Conclusie.....	33
2.10	Mogelijke toepassingen voor IIoT	34
2.11	EU GDPR.....	37
3	Studie huidige installatie Cooltech.....	39
3.1	Controllers	39
3.1.1	Carel.....	40
3.1.2	Green & Cool.....	42
3.1.3	Danfoss	43
3.1.4	Wurm	44
3.1.5	Dixell.....	45
3.1.6	Overzicht mogelijkheden controllers	46
3.2	Tijdelijke oplossing Cooltech.....	47
4	Mogelijke oplossingen	49
4.1	Stap 1: connectie met de controllers	49
4.1.1	Mogelijkheden van de controllers.....	50
4.1.2	IO-Link sensoren	50
4.1.3	“eWON” routers	53
4.1.4	OPC UA server	54
4.2	Stap 2: data opslaan in een database.....	55
4.2.1	Cloud database	56
4.2.2	On-premise database.....	57
4.3	Stap 3: rapportage van data in Microsoft Excel.....	58
4.4	Stap 4: vb.net applicatie op de database	58
4.5	Gekozen totaaloplossing	59
5	Simulatie.....	61
5.1	Werking koelinstallatie.....	61
5.2	Verloop temperaturen en openen expansieventiel	62
5.2.1	Normale werking.....	63
5.2.2	Ontdooi cyclus	64
5.3	OPC UA en database	66
5.3.1	Configuratie OPC UA server.....	66
5.3.2	Configuratie datalogger en database	69
5.4	Rapportage in Microsoft Excel.....	71
5.5	Vb.net applicatie	72

5.6	Automatisch rapport	74
6	Testen werkelijke installaties.....	75
6.1	Carel	75
6.1.1	Configuratie controllers	76
6.1.2	Configuratie OPC UA server.....	77
6.2	Green & Cool	80
6.2.1	Configuratie OPC UA server.....	80
6.3	Danfoss.....	81
6.4	Wurm.....	81
6.5	Dixell	82
7	Besluit.....	85
	Literatuurlijst	87

Lijst van Tabellen

Tabel 1: Overzicht communicatie controllers	46
Tabel 2: Prijzen Standard SQL Cloud database	56
Tabel 3: PLC variabelen simulatie.....	67
Tabel 4: Omvorming Modbus adressen Carel	79
Tabel 5: Kepsverex Modbus adressering.....	79
Tabel 6: Parameterlijst Green & Cool	80

Lijst van Figuren

Figuur 1: Logo Reditech	17
Figuur 2: Logo Cooltech	17
Figuur 3: IoT architectuur	23
Figuur 4: IIoT automatisatie piramide	24
Figuur 5: OT en IT	25
Figuur 6: Automatisatiepiramide	27
Figuur 7: OPC UA netwerk	31
Figuur 8: Toepassingen IIoT	34
Figuur 9: PlantWatchPro	40
Figuur 10: PlantVisorPro	41
Figuur 11: Schneider M241 PLC	42
Figuur 12: AK SM800	43
Figuur 13: Wurm HKS module	44
Figuur 14: C2C-MOD-S-MAN	44
Figuur 15: XWEB 500 EVO	45
Figuur 16: Overzicht monitoring huidige situatie	47
Figuur 17: IO-Link signaal	50
Figuur 18: Oplossing IO-Link Sensoren	52
Figuur 19: "eWon Flexy" router	53
Figuur 20: Logo Kepware	54
Figuur 21: Illustratie gekozen totaaloplossing	59
Figuur 22: Werking koelmeubel	61
Figuur 23: Illustratie verdamper	62
Figuur 24: Verloop temperaturen en opening expansieventiel normale werking	63
Figuur 25: Verloop temperaturen en opening expansieventiel ontdooicyclus	64

Figuur 26: Print screen Kepware simulatie	66
Figuur 27: Print screen selectie tags datalogger	69
Figuur 28: Print screen data in database	70
Figuur 29: Print screen data uit database Excel.....	71
Figuur 30: Print screen rapportage Excel met filtermogelijkheden.....	71
Figuur 31: Dashboard vb.net applicatie	72
Figuur 32: Monitoring installatie vb.net applicatie.....	73
Figuur 33: Voorbeeld automatisch rapport	74
Figuur 34: PlantWatchPro activeren plug-in.....	76
Figuur 35: PlantWatchPro virtueel apparaat	76
Figuur 36: PlantWatchPro Modbus variabelenlijst.....	77
Figuur 37: Voorbeeld API Dixell	83

Verklarende woordenlijst

AMQP = Advanced Message Queuing Protocol

API = Application Programming interface

CoAP = Constrained Application Protocol

DDS = Data Distribution Service

DPO = Data Protection Officer

DSN = Data Source Name

DTU = Database Transaction Unit

ERP = Enterprise Resource Planning

EU GDPR = The European Union General Data Protection Regulation

FTP = File Transfer Protocol

HACCP = Hazard Analysis and Critical Control Points

HTTP = Hypertext Transfer Protocol

IETF = Internet Engineering Task Force

IIoT = Industrial Internet Of Things

IoT = Internet Of Things

IT = Information Technology

JSON = JavaScript Object Notation

M2M = Machine to Machine

MES = Manufacturing Execution System

MQTT = Message Queuing Telemetry Transport

OEE = Overall Equipment Effectiveness

OGM = Object Management Group

OPC UA = OLE for Process Control Unified Architecture

OT = Operational Technology

PLC = Programmable Logic Controller

R&D = Research and Development

REST = Representational State Transfer

SCADA = Supervisory Control And Data Acquisition

TLS = Transport Layer Security

VPN = Virtual Private Network

XML = Extensible Markup Language

Abstract

In opdracht van Reditech Engineering is er voor Cooltech een IoT haalbaarheidsstudie uitgevoerd. Cooltech, een bedrijf dat koel- en vriesinstallaties plaatst en onderhoud, gebruikt vijf type controllers voor de aansturing van de koelinstallaties. Waardoor er geen uniforme manier is voor het communiceren met en het opvolgen van de installaties. Het doel van deze masterproef was om een universeel platform te ontwikkelen waar Cooltech de procesvariabelen en alarmen van verschillende koelinstallaties in kan opvolgen.

In een eerste fase werd er een connectie gemaakt met de controllers, waarvoor een OPC UA server software werd gebruikt. Deze software kan de verschillende controllers op de verschillende locaties bereiken, door middel van VPN-tunnels tussen elke koelinstallatie en het hoofdkantoor van Cooltech. Een datalogger, een functie van de OPC UA server, haalt vervolgens gegevens en alarmen uit de controllers. Deze datalogger is gekoppeld aan een database waarin alle gegevens en alarmen worden gestockeerd. Ten slotte werd een vb.net-applicatie gemaakt die de gegevens van deze database omzet naar een universeel visueel dashboard. Om de werking van dit totaalconcept als oplossing aan te tonen werd ook een simulatie van een koelinstallatie ontwikkeld.

De simulatie heeft aangetoond dat de totaaloplossing werkt indien er een connectie tussen de OPC UA server en de controller gemaakt kan worden. Deze totaaloplossing dient nog bevestigd te worden door de implementatie en veelvoudige testen op werkelijke installaties.

Abstract in English

An IoT feasibility study is commissioned by Reditech Engineering for Cooltech, a company that implements and maintains cooling and freezing installations. Within Cooltech five types of controllers are used to remotely manage the different installations. In addition, there is no universal way to communicate with and to monitor the cooling installations. The main objective of this “master thesis” was to develop a universal platform that allows Cooltech to monitor and remotely manage all the process variables and alarms of all their different installations.

First a connection was made with the controllers, for which an OPC UA server software was used. This software is able to reach each controller through a VPN-tunnel, initiated from the remote installations towards the headquarter of Cooltech. Using an embedded datalogger from the Kepware software suite, alarms and data can be extracted from the controllers. This datalogger is linked with a database where the alarms and data will be stored. Finally, a vb.net-application is made to convert the data from a database to a visual dashboard. As part of the thesis a simulation of a cooling installation is also made to demonstrate the operation of this solution.

The simulation has proven that this solution works in the majority of the cases, except for those rare situations where a connection between the OPC UA server and the controllers couldn't be made. This solution has yet to be confirmed by actually implementing and testing it on real installations.

1 Inleiding

1.1 Situering

De masterproef is een project bij Reditech Engineering, dat opgericht is in 1995 en momenteel bestaat uit vijf vestigingen over heel de wereld. Reditech is gespecialiseerd in industriële automatisatieprojecten. De firma heeft projecten gerealiseerd voor o.a.: Aquafin, Febelco, Mercedes-Benz, Continental foods, MARS Foods,...

Eén van de mogelijk toekomstige klanten is Cooltech, gevestigd in Alken. Cooltech is gespecialiseerd in het ontwerpen en plaatsen van diepvries- en koelcellen, commerciële koel- en vriesmeubels en gekoelde werkruimtes. De producten die Cooltech aanbiedt zijn geschikt voor vele toepassingen en zijn op maat gemaakt. Verder bieden ze ook service en onderhoud aan voor geplaatste en bestaande installaties (jaarlijks onderhoud, herstellingen en het keuren van de installaties) [1].



Figuur 1: Logo Reditech [2]



Figuur 2: Logo Cooltech [1]

1.2 Probleemstelling

Wanneer een klant een onderhoudscontract tekent, verbindt Cooltech zich ertoe de betreffende koelinstallaties te monitoren. Dat wil zeggen dat het alle alarmen en de nodige procesvariabelen opvolgt van deze koelinstallaties. Deze alarmen treden op als er bepaalde limieten worden bereikt waardoor het systeem niet meer zoals behoren kan functioneren.

Cooltech heeft vele verschillende klanten zoals Aldi, Carrefour, Esso, ... , maar de wensen omtrent de eigenschappen van de koelinstallaties zijn voor elke klant verschillend. Hierdoor heeft Cooltech verschillende installaties ontworpen met controllers van verschillende fabrikanten.

Deze verschillende controllers zorgen ervoor dat de installaties van Cooltech niet uniform zijn. Hier schuilt het uiteindelijke probleem. Omdat ze met verschillende controllers werken, verschilt de manier om te communiceren met de controllers en de procesgegevens en alarmen uit de controllers te halen. Dit zorgt voor een inefficiënte manier van werken voor de personen die de installaties dienen op te volgen bij eventuele problemen.

Om toch te kunnen verbinden met de controllers, heeft Cooltech voor een tijdelijke oplossing gezorgd. Het verbindt de controllers met een Lancom 4G router. Hierdoor kan de technische dienst op de controllers inloggen vanop een afstand en zo de alarmen en systeemvariabelen van de koelinstallaties raadplegen. Het inloggen op een installatie dient per installatie opnieuw te gebeuren, wat zeer tijdrovend is. Omdat de techniekers elke installatie enkel apart dienen te bekijken, bestaat er ook de mogelijkheid dat er een alarm wordt gemist. Het missen van alarmen kan tot gevolg hebben dat Cooltech pas gecontacteerd wordt wanneer de koelinstallatie al uitgevallen is, wat de interventiekost verhoogd. De huidige mogelijkheden zijn dus vooral gericht op reactief onderhoud op het moment dat de installatie al uitgevallen is. De tijdelijke oplossing wordt momenteel voornamelijk gebruikt voor reactieve acties als er zich een defect voordoet. Zodat de servicedienst vanop een afstand kan kijken wat er precies aan de hand is.

1.3 Doelstellingen

De hoofddoelstelling van deze masterproef bestaat eruit een universeel platform te ontwikkelen waarin alle alarmen en procesvariabelen van de verschillende koelinstallaties samenkomen. Zodat Cooltech in de toekomst op een eenvoudigere en efficiëntere manier de koelinstallaties kan monitoren en ingrijpen bij alarmen of events en het uitvallen van de koelinstallaties kan vermijden. Het wil af van het reactief reageren op defecten en overgaan naar een predictieve/proactieve stijl van reageren op defecten. Verder als uitbreiding hierop dient er een visueel dashboard ontworpen te worden, waarin Cooltech alle installaties visueel kan opvolgen.

Aangezien hier vele verschillende aspecten bij komen kijken, is het noodzakelijk de hoofddoelstelling in vier kleinere deeldoelstellingen op te delen.

Een eerste deeldoelstelling is het onderzoeken van een mogelijke manier om te kunnen communiceren met elk type controller. Hier zal er ten eerste gekeken worden naar de functies waarover de controllers zelf beschikken in verband met het naar buiten brengen van de procesdata en alarmen. Vervolgens wordt er onderzocht wat de mogelijkheden zijn om met de verschillende type controllers te communiceren en wat de hiervoor noodzakelijke protocollen zijn.

De tweede deeldoelstelling is vinden van een communicatiesysteem dat met alle controllers kan communiceren. Dit systeem zal alle data en alarmen van de verschillende controllers verzamelen en vervolgens in een database plaatsen. Het is noodzakelijk dat het een systeem is dat met elke controller kan communiceren zodat er uiteindelijk één uniform platform is waar de alarmen en de data in getoond worden.

De derde doelstelling is een tussenstap. Hier dient er een kleine applicatie gebouwd te worden die enkel de alarmen van de verschillende installaties oplijst. In deze applicatie moet er de mogelijkheid zijn om de nodige filtering- en zoekmogelijkheden uit te voeren. Deze tussenstap is noodzakelijk zodat de technische dienst, die instaan voor het onderhoud, het management kan overtuigen dat dergelijk systeem het investeren waard is met behulp van een ROI.

Een vierde doelstelling is het visueel maken van alle data. Als alle data van de verschillende koelinstallaties in een database staat, dan gaat dit om zeer veel data en wordt dit onoverzichtelijk. Waardoor het moeilijk wordt om conclusies te trekken. Daarom is het noodzakelijk om een overzichtelijk dashboard te maken waar in één oogomslag de meest kritische alarmen in zichtbaar zijn. In dit overzicht staat alle informatie over het alarm, wat het probleem is en op welke installatie het zich voordoet.

1.4 Methode

Om één universeel en visueel platform voor alarmen van de verschillende koelinstallaties te realiseren, is er ten eerste een studie gebeurt over wat de huidige controllers al dan niet kunnen in verband met het exporteren van data en alarmen. Dit staat beschreven in “3 Studie huidige installatie Cooltech”.

Vervolgens kan er uit de studie besloten worden voor welke controllers een communicatiesoftware al dan niet noodzakelijk is. Ook is hieruit gebleken met welke communicatieprotocollen de controllers communiceren. Dit is een noodzakelijke factor voor het kiezen van een communicatiesoftware, aangezien er één gekozen dient te worden die alle protocollen van de verschillende type controllers ondersteunt.

Eenmaal de communicatie werkt met de verschillende type controllers, kunnen de nodige procesgegevens en alarmen uit de controllers gehaald worden. Deze gegevens en alarmen dienen dan gestockeerd te worden in een database. De verschillende opties voor een database worden aangehaald in paragraaf 4.2.

Wanneer alle gegevens beschikbaar zijn in een database, volgt de tussenstap. Voor deze stap wordt er een kleine applicatie gebouwd. De applicatie zal alle alarmen van de verschillende installaties verzamelen en in een Excel oplijsten. Waar achteraf de nodige filtering en zoekmogelijkheden in kunnen gebeuren.

Als laatste volgt de uitbreiding van een visueel dashboard. Aangezien een lijst met waardes in Excel, wanneer er zeer veel gegevens instaan het niet altijd even eenvoudig te lezen is. In het dashboard is het noodzakelijk dat men in één oogomslag de meest cruciale alarmen met alle bijhorende informatie kan zien, zodat de problemen in de toekomst snel en efficiënt opgelost kunnen worden. Het is ook belangrijk dat er een lijst van alle alarmen geraadpleegd kan worden waarbij de nodige filter- en zoekmogelijkheden zoals in de Excel applicatie mogelijk zijn.

2 Literatuurstudie

Vooraleer de praktische case van Cooltech besproken kan worden, is het noodzakelijk om de voorkennis van IIoT te kennen. Dit werd tijdens de literatuurstudie bekeken en beschreven. In dit hoofdstuk wordt verklaard wat IoT, IIoT en MES is en legt een aantal veel gebruikte begrippen rond de IoT uit. De meest courante protocollen die de dag van vandaag gebruikt worden in de IoT wereld worden ook beschreven. Ten slotte wordt ook de mogelijke impact van de nieuwe EU GDPR wetgeving beschreven.

2.1 Tijdslijn veranderingen in de industrie

De wereld zoals we ze nu kennen heeft al een aantal industriële revoluties doorstaan, welke telkens een grote impact hebben gehad en de wereld gemaakt hebben hoe ze nu is. Zonder deze revoluties zou het leven en de wereld er totaal anders uitzien. In totaal zijn er al 3 industriële revoluties geweest.

De 1ste industriële revolutie was een werkelijke revolutie. Het was de uitvinding van stoommachines en het gebruiken van stoom en waterkracht om allerlei machines aan te drijven. Dit zorgde voor een industriële transformatie van de samenleving door middel van treinen en de mechanisering van ontwikkeling van producten [3].

De 2de revolutie is begonnen in de periode waar elektriciteit nieuwe manieren mogelijk maakte om producten te ontwikkelen, zoals productielijnen. Dit leidde tot een 1ste vorm van massaproductie en automatisatie [3].

De 3de industriële revolutie heeft alles te maken met het opkomen van computers, netwerken, industriële robots, connectiviteit en de geboorte van het internet. Wat de manier van hoe informatie behandeld en gedeeld werd, sterk heeft veranderd. Hierdoor heeft de automatisatie van productieprocessen een grote stap vooruit genomen [3].

De 4^{de} industriële revolutie ofwel industrie 4.0 zijn we nu aan het meemaken. Deze is nog volop bezig en is zeker nog niet voorbij. Industrie 4.0 wil zeggen dat we gaan werken met “cyber physical systems”, IT en OT gaan laten samenwerken, de opkomst van IIoT (Industrial Internet of Things), Big Data, cloud, enz. . Ook de opkomst van geavanceerde machines die uitgerust zijn met cognitieve IO, zijn factoren die industrie 4.0 mogelijk maken. Dit zorgt dat de automatisatie en optimalisatie van machines en productielijnen op een volledig nieuwe manieren kunnen gebeuren, wat leidt tot nieuwe kansen om te innoveren en de industrie werkelijk te automatiseren. Het zal de industrie op een nieuw niveau zal brengen [3].

Omdat deze 4^{de} revolutie nog volop bezig is, komen er aan een zeer hoog tempo nieuwe technologieën op de markt. Wat het zeer moeilijk maakt om al de verschillende mogelijkheden in deze scriptie te beschrijven. De beschreven oplossingen, zijn oplossingen die op de markt verschenen zijn voor december 2017.

2.2 IoT

In deze scriptie zal het vooral gaan over IIoT “industrial internet of things”. Waardoor er niet diep op IoT zal worden ingegaan. Het is echter wel noodzakelijk om de grote lijnen van IoT uit te leggen. Zo kan achteraf ook de vergelijking gemaakt worden tussen IoT en IIoT.

Binnen IoT zijn er vele termen die vaak terug komen. Termen zoals Cloud, Big data, predictive maintenance, machine learning, Het verklaren van al deze termen gebeurt via onderstaande tekst, waarin ook wordt aangekaart wat IoT juist is.

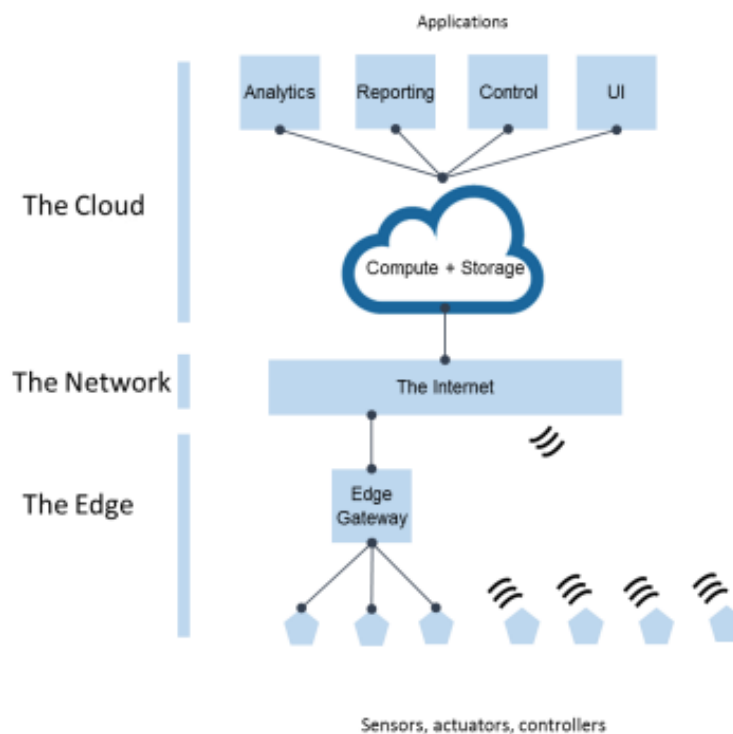
The “internet of things”, kortweg IoT wil zeggen dat alles van auto's, gsm's tot eenvoudige huis-tuin-en-keukenapparaten verbonden zijn met het internet. Wat vele nieuwe mogelijkheden biedt voor de manier waarop we handelingen doen, door het beschikbaar stellen van zeer veel informatie via het internet. Echter enkel het verbinden van deze toestellen aan het internet is niet voldoende om interessante en innovatieve manieren van het dagelijkse leven mogelijk te maken [4].

Deze toestellen dienen ten 1^{ste} data te uploaden via het internet naar de zogenaamde “cloud”. Dit is een provider van services, maar voornamelijk een grote dataopslagplaats waar je veel gegevens online kan opslaan. Het gaat om grote hoeveelheden gegevens ofwel “Big-data” en kan veel informatie bevatten zoals temperatuur, energie verbruik, eventuele storingen, Omdat al deze data opgeslagen is, beschikt men ten alle tijden van de historische data. Uit deze historische data kan men het gedrag van de apparaten leren kennen, door de opgetreden storingen te bestuderen. Zo kan er een patroon worden gevonden, dat vooraleer een storing optreedt dit detecteert waardoor men op voorhand kan ingrijpen. Je gaat als het ware storingen voorspellen en kunnen ingrijpen vooraleer storingen optreden. Dit noemen ze binnen de IoT termen “predictive maintenance”. Het leren kennen van de apparaten en het gedrag voordat storingen optreden wordt dan weer “machine learning” genoemd.

IoT-apparaten verzamelen enorme hoeveelheden gegevens over een heleboel zaken. Wat mogelijkheden biedt voor het analyseren van fenomenen welke vooraf niet mogelijk waren. Zo kan men bijvoorbeeld kijken naar het verbruik van een motor van een auto. Als dit verbruik na verloop van tijd een bepaald limiet overschrijdt, weet men dat er iets mis is. Zo kan men dan bijvoorbeeld aan de hand van dit verbruik en het aantal uren in gebruik bepalen dat er een onderdeel defect aan het gaan is. Dit en nog veel meer fenomenen kunnen met behulp van IoT worden geanalyseerd. Voor dit analyseren van de data is het ten 2^{de} is noodzakelijk dat er platformen, tools en algoritmes bestaan welke al de data die de “IoT apparaten” verzamelen kunnen verwerken [4]. Dan pas kan men werkelijk voordeel halen uit IoT.

2.2.1 IoT architectuur

De architectuur van IoT is geïllustreerd op onderstaande figuur. Hier zijn er 3 onderverdelingen gemaakt: “The cloud”, “The network” en “The Edge”.



Figuur 3: IoT architectuur [4]

De Cloud is de grote opslagplaats waar de data opgeslagen wordt. Het kan beschikken over applicaties die de analyse, reportage en controle van de data kunnen uitvoeren en een userinterface om dit allemaal zichtbaar te maken voor de gebruiker. Verder is er ook de mogelijkheid om aan “Cloud computing” of aan “Edge computing” te doen. Meer hierover staat beschreven in paragraaf “2.7 Cloud/Edge computing”.

Het netwerk is de manier waarop de toestellen verbonden zijn. Deze connectie wordt gemaakt door gebruik te maken van de juiste protocollen. Een aantal van deze protocollen zijn bekend zoals HTTP, maar er bestaat ook protocollen die speciaal voor IoT toepassingen ontworpen zijn. Een aantal van deze meest courante protocollen die gebruikt worden binnen IoT staan beschreven in paragraaf 2.9.

“The Edge” bestaat uit de “things” in internet of things. Dit zijn de sensoren, actuatoren, machines, Deze apparaten kunnen rechtstreeks verbonden zijn met het netwerk via 3G/4G of Wifi. In het andere geval is er een hulpmiddel nodig: een edge gateway. Dit zal 1 of meerdere apparaten met het netwerk verbinden [4].

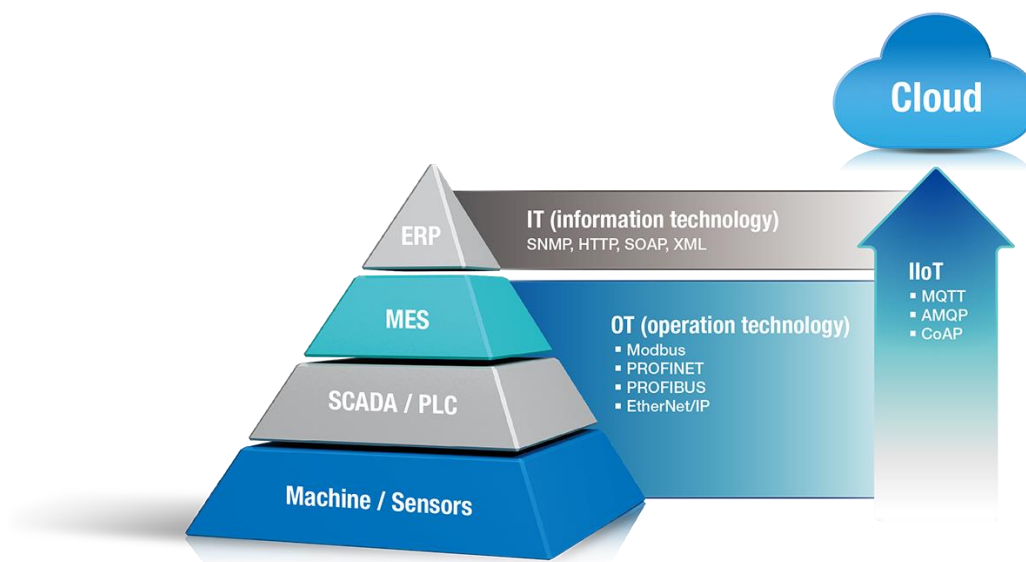
2.3 IloT

IloT en “IoT in manufacturing” zijn beide The Internet of Things in industriële toepassingsgebieden. IoT in manufacturing is echter specifiek gericht op productiebedrijven, terwijl IloT ook gebruikt kan worden door niet productiebedrijven.

Hier gaat het niet enkel om “Things” zoals machines of controllers te verbinden via het internet met de Cloud, maar ook over procesgegevens naar de Cloud te sturen. Het is dus niet altijd noodzakelijk is om een volledige machine met het internet te verbinden. Meestal is het voldoende om slechts enkele gegevens via een software of gateway uit te lezen en via het internet naar de Cloud te sturen. Hierdoor beschik je enkel over de noodzakelijke gegevens, die voldoende zijn om in de Cloud analyses te doen.

Binnen de industrie wordt dataverlies ook niet geaccepteerd. Waardoor IloT toepassing altijd gebruik maken van een store en forward principe. Dit zorgt ervoor dat bij communicatieproblemen de data tijdelijk gebufferd wordt totdat de communicatie terug functioneert. Vervolgens zal het systeem de gebufferde data mee doorsturen zodra de communicatie terug functioneert. Hierdoor is dataverlies uitgesloten.

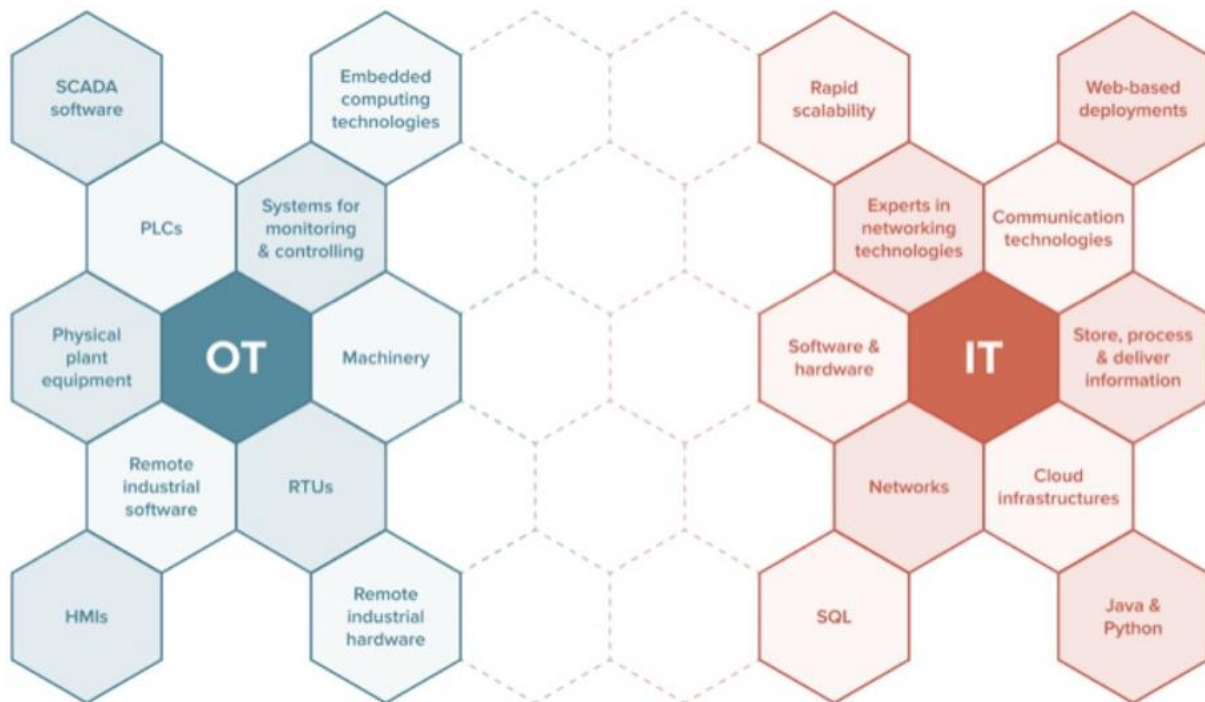
IloT heeft vele toepassingen, waaronder een aantal van de belangrijkste uitgelegd zijn in het topic “2.10 Mogelijke toepassingen voor IloT”. Het gaat procesgegevens van OT en IT laag via het internet naar de Cloud sturen. Dit is afgebeeld op Figuur 4. In de Cloud kunnen vervolgens al deze gegevens dan geanalyseerd en gevisualiseerd worden, zodat deze altijd en overal toegankelijk zijn voor de bevoegde personen.



Figuur 4: IloT automatisatie piramide [5]

2.3.1 OT + IT

Vooraleer men IIoT kan implementeren dient men de manier waarop men omgaat met de procesgegevens in de industrie te wijzigen. De manier waarop men nu voornamelijk omgaat met procesgegevens is om ze binnen de productieomgeving te houden. Wanneer men op deze manier gaat denken, dan kan men echter nooit IIoT gaan implementeren. Er dient afgestapt te worden van het voornamelijk focussen op OT(=Operational Technology) en IT (=Information Technology) mee implementeren in de productieomgeving [5]. Op Figuur 5 is te zien wat OT en IT inhouden.



Figuur 5: OT en IT [6]

Zoals op de Figuur 5 te zien is, heeft OT voornamelijk te maken met alles wat zich in de industriële wereld afspeelt. Het gaat hier over machines, PLC's, HMI's, SCADA, OT-systemen zijn meestal ook volledig afgeschermd van het internet om de veiligheid te garanderen. IT-systemen heeft vooral te maken met informatie verwerking in al dan niet grote hoeveelheden data. Binnen IT is men sterk in (SQL) databases, web gebaseerde applicaties, het connecteren van apparaten, internet security, Het is dus noodzakelijk om deze twee technologieën te combineren om IIoT mogelijk te maken.

Door het laten samenwerken van OT en IT zorgt er ten eerste voor dat een bedrijf de toegankelijkheid van gegevens vergoot op een veilige manier met de huidige IT-beveiligingsprotocollen. Want IT is sterk in het beveiligen tegen internet aanvallen , waar tegenover OT tot op het heden nog geen ervaring mee heeft. Vervolgens kunnen door de Cloud technologieën de servers naar de Cloud verplaatst worden. Wat ervoor zorgt dat er minder apparatuur nodig is en de systemen automatisch up-to-date gehouden worden [6].

2.4 IoT vs. IIoT

Een groot verschil tussen IoT en IIoT is, dat bij IIoT toepassingen men zeker moet zijn dat altijd alle data binnen komt. Want voor IoT toepassingen waar huis-tuin-en-keukenapparaten verbonden zijn met het internet, zal het geen ramp zijn als er 15 minuten geen data gelogd is. Bij IIoT toepassingen echter kunnen deze 15 minuten echter wel cruciale informatie bevatten over waarom een installatie gefaald heeft.

Een ander groot verschil is dat IIoT gebruikt wordt in een industriële omgeving waar veiligheid zeer belangrijk is. Hier gaat het niet enkel over beveiliging om het lekken van informatie tegen te gaan, maar ook de beveiliging van personen. In een industriële omgeving zijn er namelijk machines die een mens kunnen verwonden [4]. In een IIoT omgeving wil men ten alle tijden vermijden dat onbevoegden is toegang krijgen tot het recept van hun producten of controle over de machines krijgt.

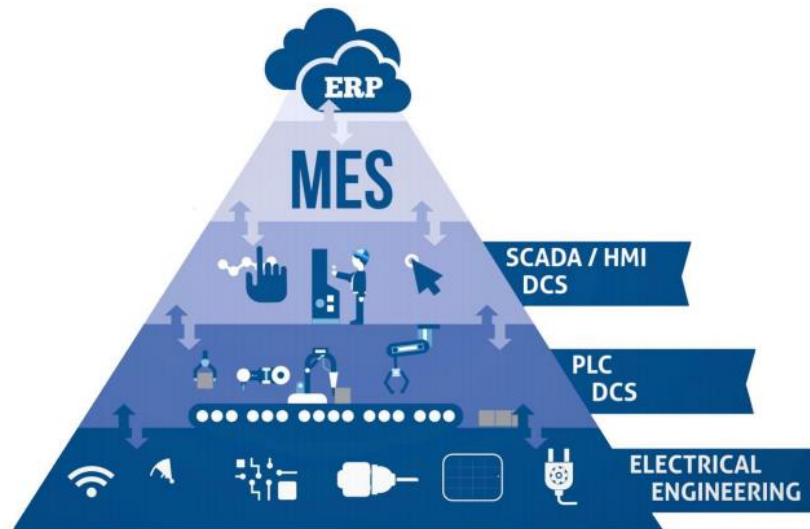
Nog een ander verschil is dat als er een storing optreedt er bij IIoT het systeem waar mogelijk de operaties dienen voortgezet te worden. Tenzij er deterministisch besloten wordt het proces te stoppen op een veilige manier [4]. Waartegenover IoT, wat meer gericht is op toepassingen voor consumenten. Zoals het verbinden van een vaatwasser of een wasmachine via het internet, welke hierdoor zelf kunnen bepalen aan de hand de kostprijs van de elektriciteit wanneer ze best hun cyclussen starten. Bij dit soort toepassingen heeft het niet veel gevolgen als er een storing voordoet en de cyclus niet start wordt. In de industrie kan dit wel zware economische gevolgen hebben.

Nog een belangrijk verschil is dat industriële toepassingen een zeer lange levensduur hebben tegenover toepassingen voor consumenten. Consumenten zijn ook vlugger bereid om een nieuw toestel te kopen dat over nieuwe en verbeterde functionaliteiten beschikt. In de industrie laten ze liefst alles zolang mogelijk werken omdat hier de kostprijzen voor het vervangen van een installaties enorm zijn. Dit wil zeggen dat de IIoT apparaten moeten kunnen blijven communiceren met oude protocollen en soms speciale middelen nodig hebben om ze te integreren in een grote IIoT omgeving [4].

2.5 MES

Manufacturing Execution System (MES) wordt even kort besproken zodat we deze achteraf kunnen vergelijken met IIoT omdat men deze soms als hetzelfde beschouwd. Vervolgens gaan we kijken naar de verschillen, de gelijkenissen en de voordelen van het ene boven het andere in “2.6 IIoT vs. MES”.

Binnen een productiebedrijf is de automatiseringspiramide altijd van toepassing. Deze staat op Figuur 6 afgebeeld.



Figuur 6: Automatisatiepiramide [7]

Bovenaan deze piramide staat het ERP wat staat voor Enterprise Resource Planning. Deze laag houdt zich voornamelijk bezig met de kosten en facturatie, de leveringen, de planning van resources, de productieplanning,

De volgende laag is de MES-laag wat staat voor Manufacturing Execution System. Dit is een systeem, waarmee productiebedrijven hun productieprocessen en procedures sturen en monitoren gebaseerd op data en operationele inputs welke ze krijgen van machines, mensen en van een software.

Een MES-systeem focust zich vooral op het productie uitvoerende gedeelte. Het zal data beheren en is hierdoor eigenlijk een vergaande digitalisering van een productieproces. Het gaat de arbeider daadwerkelijk helpen en begeleiden doorheen het productieproces op een zo efficiënt en gebruiksvriendelijke mogelijke manier. Alle stilstanden en defecten worden ook in het systeem geregistreerd. Hierdoor komen er veel gegevens beschikbaar van het systeem welke men kan gaan analyseren om zo het productieproces te verbeteren.

2.6 IIoT vs. MES

MES en IIoT kunnen allebei over evenveel procesdata beschikken, maar typisch beschikt een MES-systeem over meer en gedetailleerdere data. Dit omdat de bandbreedte om alles via het internet naar de Cloud te sturen anders te groot dient te zijn. Ook omdat de meeste bedrijven niet al hun gegevens online in de Cloud beschikbaar willen stellen.

Een MES-systeem dient voornamelijk voor het digitaliseren, aansturen en monitoren van een productieproces. Het kan ook beschikken over sterke analytische tools, maar de functionaliteiten beperken zich meestal tot één enkele bedrijfslocatie. Waartegenover IIoT, wat voornamelijk focust op predictive maintenance, voorraadbeheer en het monitoren van processen. Wel over de mogelijkheid beschikt om informatie van meerdere locaties te bundelen.

Dit zorgt ervoor dat IIoT-toepassingen voornamelijk gebruikt worden voor nieuwe toepassingen en het bundelen van de gegevens van meerdere locaties. Een aantal van deze IIoT-toepassingen staan beschreven in topic “2.10 Mogelijke toepassingen voor IIoT”.

Een ander voordeel van IIoT tegenover een MES-systeem is dat een IIoT oplossing ook geïmplementeerd kan worden in een niet productiebedrijf. Want een MES-systeem biedt enkel een oplossing voor productiebedrijven. IIoT kan daarom ook zeer interessant zijn voor servicebedrijven en leveranciers van grondstoffen, waarvoor MES geen oplossing kan bieden.

Het is niet zo dat IIoT zal zorgen dat MES verdwijnt of omgekeerd. Ze hebben beiden hun voordelen. Als men wilt zorgen dat een productiebedrijf volledig industrie 4.0 klaar is dan dient men zowel IIoT als MES te implementeren. Typisch zal men ook aanraden een MES-systeem eerst te implementeren omdat dit zorgt dat alles wordt gedigitaliseerd. Dit maakt vervolgens alle gegevens toegankelijker voor een IIoT oplossing.

2.7 Cloud/Edge computing

Data computing wil gegevensverwerking zeggen. Dit wil zeggen dat de gegevens herkend, geschaald, gefilterd of geanalyseerd worden indien nodig. Edge computing binnen IoT betekent dat de data computing zich afspeelt dicht bij de gegevensbronnen en niet in een centrale Cloud [8].

Om in industriële bedrijven IIoT volledig te kunnen benutten moeten edge en cloud computing samenwerken. Voor een aantal toepassingen waar er maar een kleine vertraging is toegestaan, een slechte verbinding is of waar de bandbreedte beperkt is, is edge computing beter. Want de gegevensverwerking gebeurt dan aan de bron. Voor toepassingen waar er veel rekenkracht nodig is, data van meerdere locaties dient geanalyseerd te worden, machine learning, ... kiest men beter voor Cloud computing [8].

Dit zorgt ervoor dat beide edge en Cloud computing belangrijk zijn in een industriële omgeving en men moet kijken wanneer men best welke manier van gegevensverwerking kiest.

2.8 IIoT platformen

IIoT is sterk in populariteit aan het stijgen en dat gaat voor vele bedrijven niet onopgemerkt voorbij. Het is ook nog relatief nieuw waardoor er nog geen standaard IIoT platform bestaat. Om dit standaard platform te worden, zijn een aantal grote bedrijven aan het strijden. Bedrijven zoals: Microsoft, Amazon, GE, etc, ... willen allemaal het standaard IIoT platform worden.

Wat houdt een IIoT platform juist in? Het hoofddoel van een IIoT platform is een communicatiesoftware zijn om alle hardware te verbinden met de Cloud. Maar omdat bedrijven de oorlog willen winnen van beste IIoT platform, zorgen ze ervoor dat hun platform nog net iets meer kan. Waardoor vele platformen een volledige end-to-end oplossing bieden. Deze beschikken dan over plug-ins die de data-analyse, visualisatie en nog veel meer kunnen doen.

2.9 IIoT-protocollen

Om alle gegevens naar de Cloud te sturen zijn er specifieke communicatieprotocollen noodzakelijk. Hiervoor bestaan echter vele protocollen en elk protocol heeft zijn eigen voordelen. Het ene focust meer op snelheid, het andere meer op bandbreedte en dus het versturen van grote pakketten data. Waardoor het vinden van het juiste IIoT-protocol niet altijd even eenvoudig is. Hieronder staan in het kort de meest voorkomende protocollen voor IIoT beschreven.

2.9.1 Client/Server vs. Publish/Subscribe

Vooraleer een aantal protocollen besproken kunnen worden, is het noodzakelijk om te protocollen te groeperen in twee categorieën: Client/Server en Publish/Subscribe.

Bij de Client/Server protocollen is het noodzakelijk dat de Client verbindt met de Server en een aanvragen doet. Dit zorgt ervoor dat de Client op voorhand op de hoogte van de Server moet zijn en ermee kunnen verbinden. In dit model beschikt de server over de data en antwoord op aanvragen van de Clients [9].

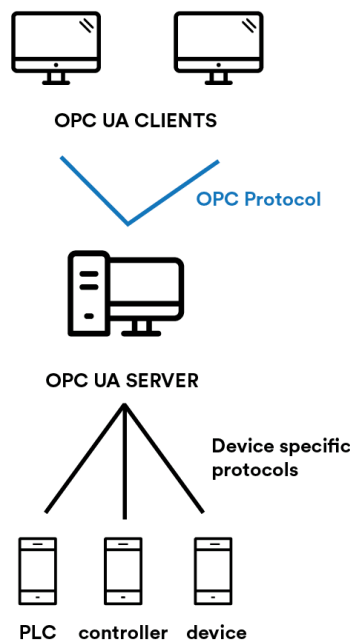
Bij de Publish/Subscribe protocollen dienen de apparaten te verbinden en data publiceren via een topic een broker. Een broker is een soort van tussenpersoon waar gebruikers mee kunnen verbinden. Deze gebruikers kunnen zich dan vervolgens abonneren op het topic waar data op gezet wordt [9].

De verschillen tussen de twee is dat in eerste instantie zijn Client/Server protocollen veiliger zijn omdat ze gebaseerd zijn op een point-to-point connectie. Ze zijn echter wel minder schaalbaar wat de Publish/Subscribe protocollen wel zijn. Client/Server protocollen zijn in het algemeen beter geschikt als de infrastructuur gekend is en Publish/Subscribe protocollen zijn beter als infrastructuur niet voldoende gekend is [9].

2.9.2 OPC UA

OPC UA staat voor OLE for Process Control Unified Architecture. Het is een Client/Server protocol, waarbij Clients: verbinden, browsen, lezen en schrijven naar industriële apparaten. UA staat voor de communicatie van de apparaten naar de transport laag en is zeer compatibel tussen verschillende fabrikanten. Het is ook een zeer veilig protocol omdat het gebruik maakt van tweerichtings encryptie. Het wordt voornamelijk gebruikt om PLC en sensor data in een SCADA of MES-systeem te krijgen, maar heeft ook zijn toepassingen binnen IIoT [9].

OPC UA is de opvolger van OPC DA, wat in de industrie nog altijd veel in gebruik is. Echter wordt OPC UA aangeraden voor nieuwe installaties. Dit omdat OPC UA platform onafhankelijk is, wat het veel toegankelijker maakt. Een ander groot voordeel is dat de OPC UA servers gebruikt kunnen worden voor embedded platformen. Hierdoor kunnen er apparaten gemaakt worden die hun eigen OPC UA server aan boord hebben [10].



Figuur 7: OPC UA netwerk

Op Figuur 7 is te zien hoe een OPC-netwerk er typisch uitziet. Deze illustreert hoe er meerdere apparaten, controllers en PLC's, met elk hun specifiek protocol, verbonden worden met één OPC UA server. Deze server kan met al deze protocollen communiceren waardoor er eenvoudig gewerkt kan worden met apparaten van verschillende fabrikanten.

Een OPC client dient enkel het OPC-protocol om te kunnen communiceren met de server. Zo kan het met elk apparaat dat verbonden is met de server communiceren. Wat voor de fabrikanten een grote besparing in kostprijs is want ze dienen op deze manier enkel nog de licentie van het OPC-protocol aan te schaffen.

2.9.3 HTTP (REST/JSON)

Hypertext Transfer Protocol (HTTP) is een client/server protocol dat voornamelijk bekend is in de IT en web wereld. Dit omdat er zeer veel open source tools bestaan die HTTP gebruiken, wat het zeer toegankelijk maakt. Voor IIoT toepassingen is HTTP niet altijd even geschikt omdat het een zwaar protocol met veel overhead is. Overhead zijn bits die nodig zijn voor het communicatieprotocol, maar geen nuttige informatie bevatten [9]. Veel overhead is nadelig voor IoT en IIoT toepassingen omdat er meer “onnodige” data verstuurd dient te worden. Dit zorgt ervoor dat het versturen van een pakket meer energie en data vereisen omdat het grotere pakketten data zijn. Dit is nadelig is voor batterijtoepassingen en het dataverbruik.

HTTP kan binnen IIoT wel gebruikt worden voor het verzenden van blokken data zoals elk uur een blok van data met elke minuut het volume van een tank. Op deze manier is de overhead relatief klein ten opzichte van de grote van het pakket. Gebruik dus HTTP niet voor data uitwisseling aan een hoge snelheid zoals om de 100ms [9].

Om de veiligheid te verzekeren gebruik je best altijd HTTPS, welke ervoor zorgt dat de informatiestroom versleuteld wordt. [9]. Dit omdat de veiligheid in een industriële omgeving altijd dient gegarandeerd te worden. Niet enkel voor persoonsveiligheid, maar ook om het lekken van informatie tegen te gaan. Dit lekken van informatie door het gebruiken van een onveilige communicatie wil men te allen tijde vermijden.

In IIoT toepassingen is het noodzakelijk om met de server applicaties te communiceren. Om ervoor te zorgen dat de clients en de server dezelfde taal spreken kan Representational State Transfer (REST) gebruikt worden. REST is een platform onafhankelijke standaard van communiceren, waardoor bijvoorbeeld een JAVA client met een .NET server kan communiceren. Om de uitgewisselde gegevens dan weer te geven in een voor zowel de client als voor de server verstaanbaar formaat, wordt JavaScript Object Notation (JSON) gebruikt. JSON is een gelijkaardig aan XML, maar is lichter en flexibeler. Waardoor het vaker in IoT toepassingen wordt gebruikt.

2.9.4 MQTT

Message Queuing Telemetry Transport (MQTT) is een publish/subscribe protocol dat in eerste instantie ontworpen is voor SCADA. Het focust zich op een minimale overhead en een betrouwbare communicatie. MQTT heeft een typische overhead van 2 bytes. Het is ook een zeer eenvoudig protocol zoals HTTP en meestal wordt een JSON of binair formaat gebruikt [9].

Gebruik MQTT wanneer bandbreedte prioriteit is en je de infrastructuur niet kent. Je moet wel nog zorgen dat je over een MQTT broker beschikt waar je de data op kan publiceren. Beveilig je communicatie door gebruik te maken van een Transport Layer Security (TLS).

2.9.5 CoAP

Het Constrained Application Protocol (CoAP) is een client/server protocol, dat ontwikkeld is om een protocol te bekomen dat de interoperabiliteit van HTTP heeft met een minimale overhead. Het is daarom ook vergelijkbaar met HTTP, maar gebruikt UDP/multicast in plaats van TCP. Ook heeft CoAP een vereenvoudigde header ten opzichte van HTTP, wat de overhead en de grootte van elke aanvraag van data verkleint [11]. CoAP gebruik je als HTTP te intensief is voor de bandbreedte.

2.9.6 DDS

Data Distribution Service (DDS) is een publish/subscribe. Het is ontworpen voor M2M communicatie. Het is een open standaard ontworpen door de Object Management Group (OGM). Het is een zeer snel protocol, ideaal voor betrouwbare real-time data en wordt voornamelijk gebruikt bij M2M communicatie [12].

2.9.7 AMQP

Advanced Message Queuing Protocol (AMQP) is een publish/ subscribe protocol, dat uit de financiële sector ontstaan is. Het heeft zijn toepassingen in de IT, maar nog niet zozeer in de industrie. Het grootste voordeel van AMQP is de robuuste communicatie. Het kan garanderen dat een datatransactie voltooid is. Het grootste nadeel echter van dit protocol is dat het een zeer zwaar protocol is. Waardoor het niet geschikt is voor elke toepassing [9].

2.9.8 Conclusie

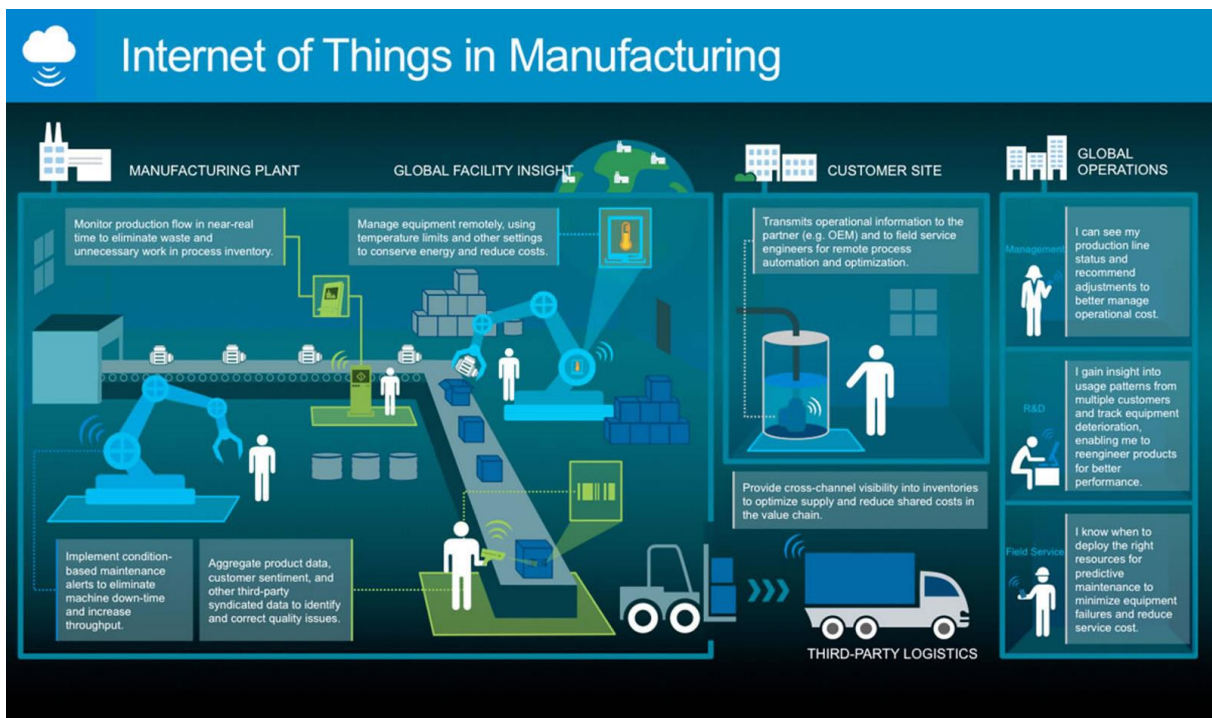
OPC UA, HTTP, CoAP, DDS, AMQP kunnen allemaal gebruikt worden voor IIoT applicaties. Hier is geen protocol dat tot nu toe de standaard is, maar ze hebben wel allemaal hun voor- en nadelen. Het is dus afwegen welk protocol het beste is voor een bepaalde toepassing. Het is altijd mogelijk dat één van deze protocollen, een volledig nieuw protocol of een niet vernoemd protocol in de toekomst het standaard IIoT-protocol wordt, maar dit zal de toekomst moeten uitwijzen [9].

2.10 Mogelijke toepassingen voor IIoT

IIoT-toepassingen zijn voornamelijk bedoeld om te monitoren, analyses uit te voeren en rapportage doeleinden. Niet om een proces of operationele activiteiten aan te sturen. Want IoT is 'near real-time' en afhankelijk van een internetverbinding, indien de toepassing in de cloud draait. Als deze internetverbinding weg zou vallen, dan vallen de stuurfunctionaliteiten en dus het proces stil. Dit is een risico dat een productiebedrijf meestal niet wil nemen.

Dit topic legt een aantal toepassingen die met IIoT verwezenlijkt kunnen worden kort uit. De verschillende toepassingen zijn onderverdeelt in groepen die aangeven waar ze toepasbaar zijn in een productieomgeving. Er zijn 3 grote groepen, welke ook afgebeeld staan op Figuur 8, namelijk:

- De productieomgeving,
- De site van de klanten,
- Globale operaties.



Figuur 8: Toepassingen IIoT [13]

Voor elke groep bestaan er een aantal IIoT-toepassingen waaruit men voordeel kan halen in productiviteit en of efficiëntie. Enkele van deze toepassingen zullen in deze scriptie besproken worden, maar het is onmogelijk om alle mogelijke toepassingen te bespreken.

Toepassingen productieomgeving

Voor de productieomgeving zijn er vele toepassingen. Eén van de toepassingen is het zo goed als real-time monitoren van een productieproces. Dit zorgt ervoor dat de productiestroom goed opgevolgd kan worden en men exact weet hoeveel onderdelen er geproduceerd zijn en hoeveel afkeur er is. Uit deze gegevens kan de Overall equipment effectiveness (OEE) van een machine automatisch berekend worden. Aan de hand van deze OEE kunnen de nodige beslissingen genomen worden, welke ervoor zorgen dat de machine efficiënter wordt.

Voor een andere toepassing van IIoT in een productieomgeving gaat het energieverbruik en de temperatuur van een component zoals een motor opgevolgd worden. Aan de hand van deze waarden kan men ervoor zorgen dat de motor nooit boven zijn optimale temperatuur werkt. Dit zorgt ervoor dat de motor minder verbruikt en hij energie uitspaart.

Aan de hand van het energieverbruik en het aantal machine uren van een motor kan men ook bepalen wanneer een motor vervangen dient te worden. Want als een motor meer energie nodig heeft om een bepaalde snelheid met een bepaalde belasting te halen, dan weet men dat er componenten van de motor aan het verouderen zijn. Om te bepalen welke componenten van de motor verouderd zijn, gaan analytische modellen toegepast worden op de data van de motor die beschikbaar gesteld is via IIoT. Hierdoor is het mogelijk om te voorspellen wanneer er componenten vervangen dienen te worden vooraleer ze defect zijn. Waardoor de downtime van de machines sterk daalt en er aan predictive maintenance gedaan wordt. Het uitvoeren van analyses op data waardoor het mogelijk is om defecten van een bepaalde machine te voorspellen wordt ook wel machine learning genoemd.

Toepassingen klantzijde

Op de site van klanten zijn er twee toepassingen die veel impact kunnen hebben. Voor deze twee toepassingen is het noodzakelijk dat de klant toestaat om data te delen via het internet ofwel zelf bezig is met IIoT.

Een eerste toepassing is voor de leveranciers van grondstoffen. Hier zal de klant het huidige niveau van grondstoffen ter beschikking stellen van de leveranciers. De leveranciers dienen dan zelf de bewaking te doen zodat er ten alle tijden voldoende grondstoffen aanwezig zijn. Deze bewaking gebeurt meestal aan de hand van een alarm dat naar de leveranciers verzonden wordt. Hierdoor dient de klant nooit meer tijd te investeren in het monitoren en bestellen van deze grondstoffen. De leverancier haalt hier het voordeel uit dat hij de levering van de grondstoffen beter kan plannen. Wat ervoor zorgt dat er in totaal minder menselijke handelingen gebruikt worden, wat het tijdsefficiënter maakt.

Een tweede belangrijke toepassing is voor machinebouwers. Zij kunnen hun machines voorzien van vele sensoren, welke allerlei gegevens kunnen doorsturen via het internet. Deze gegevens kunnen allerlei informatie bevatten van hoe de machine gebruikt wordt en waar slijtage optreedt. Zo kunnen ze aan de hand van deze gegevens, toestellen ontwerpen die in de toekomst beter bestand zijn tegen dit gebruik.

Ook kunnen deze machinebouwers betere onderhoudsdiensten aanbieden op basis van een onderhoudscontract. Zo hoeft de klant bijvoorbeeld geen wisselstukken meer in te slaan. De machinebouwer kan uit de data afleiden dat een bepaald onderdeel binnenkort dient vervangen te worden. Hij kan zo op voorhand met de klant een onderhoud inplannen en een servicetechniker laten langs gaan met het nodige wisselstuk.

Toepassingen globale operaties

De globale operaties van een bedrijf kunnen opgedeeld worden in 3 groepen:

- Management,
- R&D,
- Field service.

Het management zal door IIoT veel informatie zien over hun processen. Aan de hand van deze informatie kan het verbeteringen gaan implementeren om kosten te besparen. Zo kan het aan de hand van al deze data blijven verbeteren en de verbeteringen visueel waarnemen.

Wat R&D aan IIoT heeft staat al beschreven in de tweede toepassing voor op de site van de klanten. Het is de toepassing waarbij ze doormiddel van vele sensoren het gebruik en de slijtage van de machine kunnen bestuderen. Om zo betere ontwerpen te maken die voorzien zijn op dat bepaald gebruik.

De Field service kan door middel van de gegevens die hun geplaatste machines ter beschikking stellen, aan predictive maintenance gaan doen. Waardoor ze het onderhoud van aan de geplaatste installaties kunnen plannen.

2.11 EU GDPR

De Europese General Data Protection Regulation (EU GDPR) is de nieuwe wetgeving rond gegevens privacy die ingaat vanaf 25 mei 2018. Deze wet zal de oude Data Protection Directive 95/46/EC vervangen. De nieuwe wet is bedoeld om alle wetten rond gegevens privacy binnen Europa te harmoniseren en de gegevens privacy te versterken en het beschermen van alle EU-burgers. Ten slotte omvat deze wet ook hoe bedrijven dienen om te gaan met persoonlijke gegevens van EU-burgers [14].

Dit is belangrijk om even aan te halen in deze scriptie want wanneer een bedrijf bezig is met IoT of IIoT, dan is het met veel data aan het werken. Als deze data waarover de bedrijven beschikken persoonsgegevens bevatten of er gegevens zijn die gerelateerd kunnen worden aan één of meerdere personen, dan zal het bedrijf vanaf 25 mei 2018 zich moeten houden aan de EU GDPR wetgeving. Dit zorgt ervoor dat er best gezorgd wordt dat alle IoT en IIoT toepassingen die men de dag van vandaag ontwerpt, voldoen aan de regels van de EU GDPR. Want op het overtreden van de regels van de EU GDPR staat hoge geldboetes.

Het is belangrijk om te weten wat er verstaan wordt als persoonlijke gegevens. Dit gaat om alle informatie gerelateerd aan een persoon of een data object dat men aan een persoon kan relateren. Dit kan om allerlei zaken gaan van een foto, een naam, een post op een sociaal netwerk, een IP adres,... [14].

Hieronder staan de belangrijkste veranderingen met de vorige wetgeving kort beschreven.

Data lek melding

Volgens de GDPR is het verplicht om een inbreuk te melden aan alle lidstaten waar een data lek een risico vormt op de privacy van één of meer van de inwoners. Dit dient te gebeuren binnen de 72 uur na de gewaarwording van het data lek. De klanten waarvan er data gelekt is, dienen onmiddellijk op de hoogte gebracht te worden na de gewaarwording van het lek [14].

Recht op toegang

Door de GDPR krijgen personen het recht om ten alle tijden bevestiging te krijgen van een gegevensbeheerder, als er persoonsgegevens van desbetreffende gebruiker verwerkt worden en voor welke reden. Verder moet de gegevensbeheerder van de data ten alle tijden een gratis kopie van al de persoonsgegevens die zij beschikken in een elektronische vorm kunnen aanbieden [14].

Recht op vergeten

Een ander belangrijk recht wat personen krijgen is het recht om volledig vergeten te worden. De gegevensbeheerder dient dan op aanvraag van de gebruiker al zijn/haar gegevens te wissen en verdere verspreiding van data stop te zetten [14].

Privacy bij ontwerp

Privacy bij ontwerp bestaat al jaren, maar is nooit onderdeel geweest van een wettelijke eis. Door de GDPR gebeurt dit nu wel. Dit betekent concreet dat er rekening gehouden dient te worden met de gegevensbescherming vanaf het ontwerp van een systeem [14].

Data Protection Officers (DPO)

Nu moeten gegevensbeheerders voor de wetgeving hun activiteiten van gegevensverwerking melden aan lokale Data Protection Authorities (DPAs). Wat voor grote multinationals een hel kan zijn omdat ze dit voor elk EU-land afzonderlijk dienen te doen. Elk land heeft dan ook weer verschillende meldingsvereisten [14].

Dit zal door de GDPR veranderen. Hierdoor is het niet meer nodig om aan elke DPA-activiteiten van gegevensverwerking te melden. In plaats daarvan zullen er interne vereisten voor het bijhouden van gegevens. Voor grote bedrijven welke op een grote schaal met veel met persoonsgegevens bezig zijn moet een DPO aangesteld worden. Wie de taak heeft van het voor het opleiden en toezien dat het personeel de regels van de GDPR nastreven [14].

Boetes

Als een bedrijf een overtreding begaat op 1 van de regels van de GDPR, kan er een boete opgelegd worden tot 4% van de jaarlijkse wereldwijde omzet of maximaal 20 miljoen. Deze regels en boetes gelden ook voor “clouds”, wat binnen de IoT vaak van toepassing is [14].

Toepasbaarheid

Dit is één van de belangrijkste en grootste veranderingen. Deze wil zeggen dat de GDPR-regels voor alle bedrijven gelden die over persoonsgegevens van personen in de EU beschikken, ongeacht de vestiging van het bedrijf [14].

3 Studie huidige installatie Cooltech

Door de literatuurstudie zijn termen zoals Cloud, big data, OPC UA, ... en de mogelijkheden van IIoT gekend. Hierdoor kan er efficiënter en gericht naar een mogelijke IIoT oplossing voor Cooltech gezocht worden. Vooraleer dit zoeken van oplossingen daadwerkelijk kan gebeuren, is noodzakelijk om de huidige situatie goed te bestuderen.

3.1 Controllers

De koelinstallaties van Cooltech hebben elk één controller die de installaties aanstuurt en regelt. De controllers die Cooltech gebruikt zijn van volgende fabrikanten:

- Green & Cool,
- Carel,
- Danfoss,
- Wurm,
- Dixel.

De algemene werking van de controllers is gelijkaardig, maar ze verschillen op een aantal vlakken zoals communicatie en visualisatie. In de onderstaande paragrafen is uitgelegd wat elke controller kan in verband met het naar buiten brengen van data en alarmen en het communicatieprotocol dat ze gebruiken.

3.1.1 Carel

Bij Carel zijn er twee modellen die Cooltech gebruikt: PlantWatchPro en de PlantVisorPro. Dit zijn twee gelijkaardige controllers, maar de PlantVisorPro over een aantal functies meer dan de PlantWatchPro omdat deze als de grote broer beschouwd wordt. De functionaliteiten van beide controllers over het extraheren van alarmen en procesvariabelen staan in de twee volgende topics uitgelegd.

3.1.1.1 PlantWatchPro

De PlantWatchPro, afgebeeld op Figuur 9, biedt enkele mogelijkheden voor het exporteren van rapporten en alarmen naar een USB-opslagapparaat [15]. Om deze gegevens te exporteren, dient er een manuele handeling uitgevoerd te worden op de PlantWatchPro. De manuele handeling houdt in dat er op een knop gedruwd dient te worden om de bestanden te exporteren.

Een USB-opslagapparaat of een SD-kaart dient alvorens het gebruik met de PlantWatchPro geformatteerd te worden met het FAT32 bestandssysteem [15].



Figuur 9: PlantWatchPro [15]

De PlantWatchPro beschikt ook over een web-interface. Deze interface ziet er exact hetzelfde uit als de interface, te zien op het scherm van het toestel [15]. Via de web-interface is het ook mogelijk om de alarmen en rapporten te downloaden, maar ook hiervoor is een manuele handeling noodzakelijk.

De alarmen die via de interface manueel gedownload kunnen worden, is een lijst van enkel alle actieve alarmen. Deze lijst wordt opgeslagen in een “.CSV” file, wat compatibel is met Excel. De rapporten die gedownload kunnen worden, zijn HACCP-rapporten of gelogde rapporten [15].

Ook beschikt de PlantWatchPro over een “scheduler” functie, waarmee er vier systeem notificatie kanalen geconfigureerd kunnen worden, namelijk [15]:

- Sms,
- E-mail,
- RemotePro,
- Relais.

De scheduler kan een automatisch rapport genereren en verzenden via één van de vier notificatiekanalen. In de scheduler kunnen regels gekoppeld worden wanneer het een rapport dient te maken en versturen. Deze regels kunnen tijdsgebonden zijn, maar ook getriggerd worden via een situatie zoals een alarm [15].

Ten slotte is het communicatieprotocol van de PlantWatchPro Modbus TCP/IP en kan deze controller geconfigureerd worden als Modbus slave. Waardoor het kan communiceren met 3th party apparaten. De PlantWatchPro kan maximaal 500 Read/Write variabelen beschikbaar stellen. Er is een Safety plug-in nodig om de controller als Modbus slave in te stellen [16].

3.1.1.2 PlantVisorPro

De PlantVisorPro te zien op Figuur 10, heeft dezelfde mogelijkheden voor het exporteren van alarmen en rapporten als de PlantWatchPro. Het grote verschil schuilt in het aantal te besturen koelsystemen en hun grootte. De PlantVisorPro kan hier veel grotere en eventueel meerdere koelinstallaties monitoren en managen [17].



Figuur 10: PlantVisorPro [17]

Modbus TCP/IP is ook het communicatieprotocol van de PlantVisorPro en kan ook geconfigureerd worden als Modbus slave. Deze controller kan echter tot maximaal 1000 of 3000 Read/Write variabelen beschikbaar stellen, afhankelijk van het model. Ook hier is de Safety plug-in noodzakelijk voor de controller in te stellen als Modbus slave. Ten slotte beschikt de PlantVisorPro ook over het FTP, waardoor er bestanden met de controller uitgewisseld kunnen worden [18].

3.1.2 Green & Cool

Green & Cool is een bedrijf waarmee Cooltech samenwerkt voor het realiseren van hun CO₂ koelinstallaties. Dit is de toekomst voor de koelinstallaties omdat deze tot nu toe de meest economische en ecologische oplossing biedt en 100% milieuvriendelijk is.

In de koelinstallaties van Green & Cool wordt een Schneider PLC gebruikt. Deze PLC is binnen de industrie een welgekende controller, welke vaak meerdere mogelijkheden biedt voor communicatie met andere apparaten. Op welke manieren er gecommuniceerd kan worden hangt af van het type. In de koelinstallaties van Cooltech gebruiken ze de M241 PL, afgebeeld op Figuur 11.



Figuur 11: Schneider M241 PLC [19]

De M241 PLC beschikt over meerdere manieren om data beschikbaar te stellen. Manieren zoals een Webserver, FTP client/server en OPC UA client server, Modbus slave TCP/IP [20].

De Webserver zorgt ervoor dat de visualisatie van de installatie vanop een afstand beschikbaar is via een Web browser. Door middel van VPN-modems wordt het systeem beveiligd zodat niet zomaar iedereen de visualisatie kan raadplegen [20].

Het File Transfer Protocol (FTP) zorgt ervoor dat bestanden uit de controller kunnen worden gehaald via een FTP client. Ook kunnen bestanden of programma's op de controller geplaatst worden [20].

OPC UA server is zoals beschreven in paragraaf 2.9.2. De Schneider PLC kan hier namelijk ingesteld worden als een OPC UA server. Zo kan men met behulp van een OPC UA client variabelen en alarmen uitlezen.

De controller staat ook toe om via Modbus TCP/IP-gegevens uit te lezen. Hiervoor kan er een I/O tabel in de controller aangemaakt worden welke aangeeft welke variabelen beschikbaar zijn via Modbus TCP/IP. Meer over hoe zulke I/O tabel en hoe de gegevens via modbus uitgelezen dienen te worden, staat beschreven in 6.2.

3.1.3 Danfoss

In de koelinstallaties van Cooltech wordt de AK SM800 van Danfoss gebruikt. Deze controller is te zien op Figuur 12. Ook bij dit type controller is er gekeken wat de mogelijkheden zijn voor het naar buiten brengen van data en het gebruikte communicatieprotocol.



Figuur 12: AK SM800 [21]

Ten eerste beschikt de AK SM over een ethernetpoort. Deze poort kan enkel gebruikt worden om de controller te verbinden met het internet of een lokaal netwerk [21]. Dit zorgt ervoor dat de visualisatie, zichtbaar op het scherm van de controller, toegankelijk is op het netwerk. Op deze visualisatie is het mogelijk op te variabelen te raadplegen en de procesparameters aan te passen.

De AK SM beschikt ook over het FTP. Hiermee kan er op een afstand verbinding gemaakt worden en kunnen er systeembestanden geladen en geopend worden. Verder is het ook mogelijk om via het FTP datapunten op te halen [21].

Het communicatieprotocol dat gebruikt wordt, is Modbus RTU [21]. Echter is er geen documentatie gevonden om deze controller te gebruiken als Modbus slave. Een andere mogelijkheid is via HTTP-communicatieprotocol, met XML gecodeerde aanvragen [22]. Het HTTP protocol staat beschreven in paragraaf 2.9.3 en is een alternatief is voor REST.

Ten slotte bestaat er binnen Danfoss een PC applicatie: de AK EM. Met deze applicatie is het mogelijk om meerdere installaties te managen en alle gegevens beschikbaar te stellen via API's [21]. Hierdoor kunnen alle alarmen opgevraagd worden vanuit een applicatie door “/alarmlist” te pollen.

3.1.4 Wurm

De controllers van Wurm die Cooltech gebruikt zijn HKS-modules, zoals te zien op Figuur 13. Dit zijn relatief gesloten controllers, maar hebben toch enkele mogelijkheden om procesvariabelen naar buiten te brengen.



Figuur 13: Wurm HKS module [23]

Het communicatieprotocol dat de HKS-modules gebruiken met onderliggende modules is CAN-bus [23]. Om gegevens uit de controller te kunnen halen is de C2C-MOD-S-MAN (afgebeeld op Figuur 14) noodzakelijk. Dit is een C2C module die de gebruikte CAN-bus converteert naar Modbus-RTU. Deze module werkt als een Modbus slave en zorgt ervoor dat er enkele procesvariabelen toegankelijk worden via Modbus RTU [24]. De nieuwere koelinstallaties van Cooltech gebruiken een nieuwere C2C module, namelijk de C2C-MOD-SGT. Deze module is volledig gelijkaardig aan de C2C-MOD-S-MAN en heeft voornamelijk dezelfde functionaliteiten. Hoe deze gebruikt dienen te worden staat beschreven in 6.4.



Figuur 14: C2C-MOD-S-MAN [24]

3.1.5 Dixell

Er zijn twee controllers van Dixell die Cooltech gebruikt, de XWEB en de XWEB EVO. De XWEB is het oudere model en werd geproduceerd en verkocht tussen 2007 tot 2016. Na 2016 is er overgegaan naar de XWEB EVO, het nieuwe toegankelijker model met meer functionaliteiten [25].



Figuur 15: XWEB 500 EVO [26]

De XWEB en de XWEB EVO zijn toestellen met een webserver gebaseerd op Linux, waardoor er geen speciale programma's nodig zijn voor de monitoring. De monitoring gebeurt via een internetbrowser. Dit is voor de EVO en niet EVO-toestellen [25].

Beide de XWEB en de XWEB EVO beschikken over een scheduler functie waarmee commando's op tijd gestuurd kunnen worden. Commando's zoals het exporteren of mailen van een gegevens. Het is ook bij beide toestellen mogelijk om alarmen en gegevens te exporteren in een formaat compatibel met Excel. Dit dient echter te gebeuren door handmatig op een exportknop te duwen [26].

Beide Dixell regelaars hebben ook het Modbus RTU-protocol om te kunnen communiceren met onderliggende apparaten.

De XWEB heeft echter geen mogelijkheid om te kunnen communiceren met bovenliggende systemen. Dixell heeft echter wel een PLC-programma geschreven dat dienst doet als verkeersagent op de Modbus RTU-bus. Hierdoor is het mogelijk om 2 masters op dezelfde bus te plaatsen om toch een beperkt aantal gegevens te kunnen lezen.

Bij de XWEB EVO kan er daarentegen wel zeer eenvoudig data uitgelezen worden via het HTML protocol. Deze controller beschikt over een aantal API's om informatie uit de controller te halen.

3.1.6 Overzicht mogelijkheden controllers

Op Tabel 1 staan alle belangrijke besproken onderwerpen in verband met het communiceren met de verschillende type controllers opgelijst. De eerste 2 kolommen “FTP” en “Data zelf exporten” zijn bekeken om een zo kost efficiënt mogelijke oplossing te kunnen vinden. Want als één van de controllers zelf op een efficiënte manier data kan exporteren, dient hiervoor geen oplossing gezocht te worden. Dit is helaas niet het geval. Een aantal controllers hebben echter wel de mogelijkheid om gegevens te exporteren via een scheduler, maar dit is niet betrouwbaar genoeg omdat deze enkel na een vooringesteld tijdsplan van enkele uren gegevens zullen beschikbaar stellen, wat veel te traag is voor cruciale alarmen.

Met de communicatieprotocollen waarover de controllers beschikken kan wel iets gedaan worden. Ze hebben allemaal Modbus TCP/IP of RTU en de meeste kunnen geconfigureerd worden als Modbus slave. De mogelijk onderzochte oplossingen om communicatie met alle controllers te maken staan uitgelegd in paragraaf 4.1.

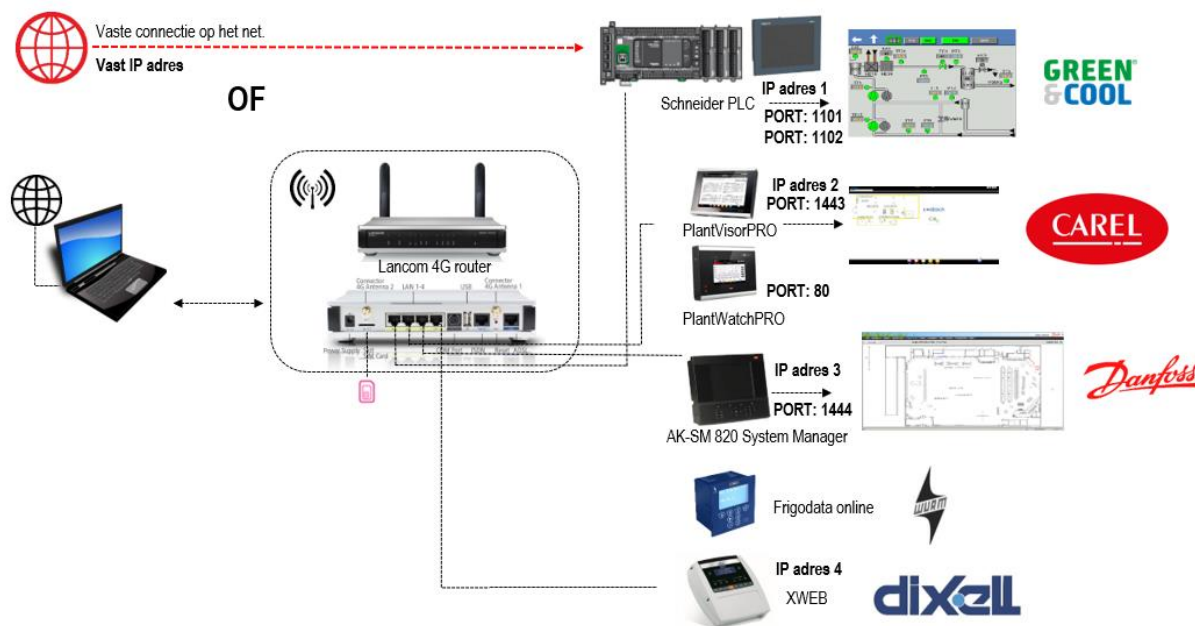
Tabel 1: Overzicht communicatie controllers

Controller	FTP	Data zelf exporteren	Protocol	Modbus slave mogelijkheid
Green & Cool	Ja	Ja, manueel	Modbus TCP/IP	Ja
Carel-PlantWatchPro	Nee	Ja, manueel of scheduler	Modbus TCP/IP	Ja
Carel-PlantVisorPro	Ja	Ja, manueel of zelf via scheduler	Modbus TCP/IP	Ja
Danfoss	Ja	Ja, manueel of zelf via scheduler	Modbus RTU, HTTP	Nee
Wurm	Nee	Ja, manueel	Canbus → Modbus RTU via C2C	Ja
Dixel	Nee	Ja, manueel	Modbus RTU, HTTP	Nee

3.2 Tijdelijke oplossing Cooltech

Figuur 16 illustreert de huidige infrastructuur die Cooltech gebruikt voor het monitoren van de koelinstallaties. Op deze figuur staat centraal de Lancom 4G router waardoor ze vanop een afstand kunnen inloggen op de netwerken van de koelinstallaties. Dit is als het een business VPN-router met een goede beveiliging en mogelijkheid tot 4G connectiviteit [27]. Door deze router kan Cooltech veilig op elke installatie inloggen en de schermen van de controllers op een afstand raadplegen. Bij deze oplossing dient echter op elke koelinstallatie apart ingelogd te worden. Dit is een tijdrovend proces, waardoor er niet efficiënt gewerkt kan worden en zorgt ook niet voor een uniform platform voor alle installaties.

Op de figuur is ook te zien dat de controllers van Wurm niet verbonden zijn met de Lancom router. Dit komt omdat deze niet met de router via TCP/IP verbonden kunnen worden. Enkel voor de laatste versies van de Wurm controllers samen met een C2C module is het mogelijk om ze vanop een afstand op te volgen via een analoge modem. Dit zorgt ervoor dat Cooltech van een aantal van de koelinstallaties, uitgerust met Wurm controllers, niet kan opvolgen.



Figuur 16: Overzicht monitoring huidige situatie [28]

4 Mogelijke oplossingen

Nu dat de mogelijkheden van de verschillende type controllers dat Cooltech gebruikt gekend zijn en er voldoende voorkennis over IIoT is, kunnen er oplossingen gezocht worden voor het probleem dat Cooltech heeft. Het probleem dat ze geen universeel platform hebben voor het monitoren van hun koelinstallaties. Dit probleem staat uitgebreid beschreven in paragraaf 1.2 .

In dit hoofdstuk zal de aanpak van het probleem opgesplitst worden in vier verschillende stappen, namelijk:

- Stap 1: connectie met de controllers
- Stap 2: data opslaan in een database
- Stap 3: rapportage van data in Excel
- Stap 4: vb.net applicatie op de database

Deze stappen zijn ook de volgorde in welke Cooltech het probleem wou zien opgelost worden.

In paragrafen 4.1 tot 4.4 staan de verschillende stappen uitgelegd met de mogelijk onderzochte oplossingen. Paragraaf 4.5 vat alle gekozen oplossingen samen in één grote totaaloplossing en geeft dit grafisch weer.

4.1 Stap 1: connectie met de controllers

Deze stap is meteen de meest uitdagende stap. Hier dient er een manier gevonden te worden om met elk type controller een connectie te kunnen maken en zo vervolgens data uit de controllers te kunnen lezen. Hiervoor zijn er vier verschillende pistes bekeken, welke uitgelegd staan in de volgende paragrafen van deze scriptie. De verschillende bestudeerde pistes voor het connecteren met de controllers zijn:

- Bestuderen van de mogelijkheden waarover de controllers zelf beschikken;
- Plaatsen van IO-Link sensoren;
- Huidige routers vervangen door “eWON” routers;
- Gebruiken van een OPC UA server.

4.1.1 Mogelijkheden van de controllers

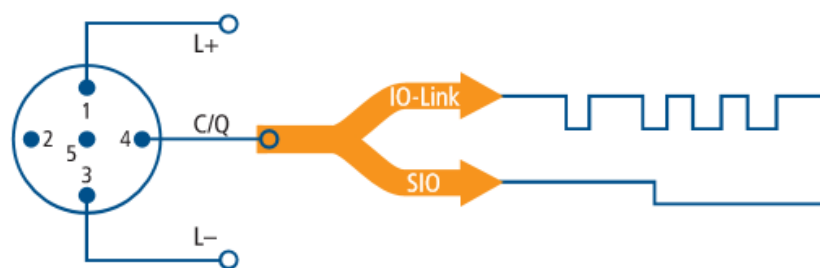
De mogelijkheden van alle verschillende type controllers zijn reeds besproken in het vorige hoofdstuk. Hier is er dan gebleken dat er geen uniforme betrouwbare manier is voor het exporteren van procesvariabelen en alarmen. Een aantal controllers beschikken echter wel over een scheduler of het FTP protocol, maar via deze manieren is het mogelijk dat cruciale alarmen vertraagd binnen komen omdat hier enkel na een bepaalde tijd data gepubliceerd kan worden.

4.1.2 IO-Link sensoren

Een tweede mogelijke oplossing voor het connecteren met de controllers is het bijplaatsen van IO-Link sensoren. Deze sensoren zijn verbonden met een IO-Link master welke de functionaliteiten van een IoT gateway bezit. Vooraleer deze oplossing besproken wordt, zal kort uitgelegd worden wat IO-Link is en wat hiervan de voordelen zijn. Deze oplossing is gevonden tijdens de voorstudie van IIoT.

4.1.2.1 Wat is IO-Link

IO-Link is de eerste I/O technologie die toegekend is als een internationale standaard (IEC 61131-9). IO-Link is een manier om te communiceren met het sensor- en actuatorniveau. De communicatie verloopt via een 3-draadse kabel, dewelke de huidige sensoren momenteel gebruiken. Echter zal over deze kabel zoals bij standaard sensoren geen analoog of digitaal signaal over gestuurd worden, maar een gemoduleerd signaal zoals te zien op Figuur 17 [29]. Het voordeel van zulk gemoduleerd signaal is dat één signaal meer data kan bevatten dan één enkele procesvariabele.



Figuur 17: IO-Link signaal [29]

IO-Link sensoren hebben ook telkens een IO-Link master nodig, die instaat voor de communicatie tot 8 verschillende sensoren en/of actuatoren. Deze IO-Link master kan vervolgens aan een bestaand industrieel netwerk geconnecteerd worden zoals Profinet om zo te communiceren met eventuele PLC's.

4.1.2.2 Voordelen IO-Link

Een eerste voordeel van IO-link is dat er meer dan één procesvariabele gemeten en beschikbaar gesteld kan worden via één enkele sensor. Want er kan door middel van het gemoduleerde signaal nog een tweede procesvariabele over dezelfde kabel gestuurd worden. Waardoor één enkele sensor de druk en temperatuur kan meten, wat een grote kost in hardware uitspaart [29].

Een tweede voordeel is dat bovenop de gemeten data, er ook diagnostische informatie van de sensor of actuator aan het signaal kan worden toegevoegd. Hierdoor kent men de gezondheid van de sensoren en actuatoren en kan men voorspellen wanneer de hardware zal gaan falen en vervangen dient te worden. Zo wordt de kans op falen van de hardware geminimaliseerd [29].

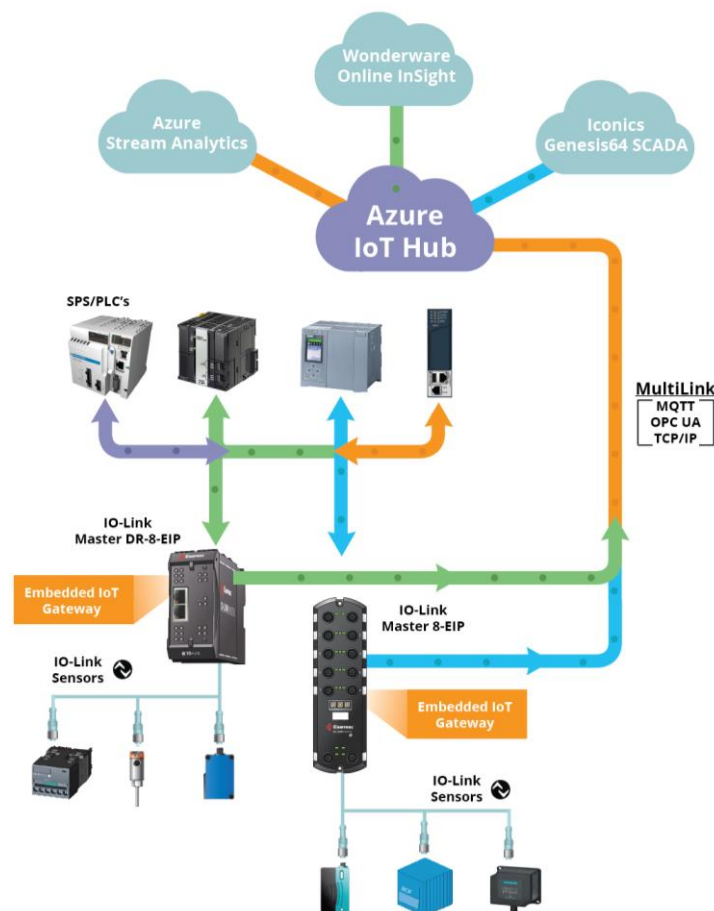
De mogelijkheid om sensor of actuator parameters dynamisch te kunnen veranderen, is een derde voordeel. Dit veranderen van de parameters kent vele voordelen, maar een van de belangrijkste is dat de omsteltijd sterk gereduceerd wordt en de productdiversiteit van één enkele machine stijgt. De parameters kunnen veranderd worden via een controller of een HMI [29].

Een vierde en laatste belangrijk voordeel is dat parameters automatisch ingesteld worden bij het vervangen van een sensor of actuator. Dit zorgt ervoor dat de downtime geminimaliseerd wordt en er geen foute instellingen gedaan kunnen worden. Ook zorgt dit ervoor dat er geen opgeleid personeel nodig is voor het vervangen van IO-Link sensoren [29].

4.1.2.3 Oplossing met IO-Link sensoren

Control is een bedrijf dat een IO-Link master heeft ontwikkeld die beschikken over een Microsoft Azure IoT Gateway SDK. Hierdoor kan de sensordata rechtstreeks geïntegreerd worden in de Azure IoT Hub. Dit terwijl het diezelfde sensordata ook naar de PLC's stuurt [30].

De oplossing hier is door enkele sensoren, die de belangrijkste procesvariabelen meten te vervangen door IO-link sensoren of het bij plaatsten van nieuwe IO-link sensoren in de bestaande koelinstallaties. Deze sensoren worden dan vervolgens verbonden met een IO-Link master en kunnen dan zonder enige controller data naar de Cloud sturen voor verder gebruik.



Figuur 18: Oplossing IO-Link Sensoren [31]

Na een gesprek met Cooltech was echter duidelijk dat deze oplossing niet realistisch is als oplossing voor het probleem van Cooltech. Dit omdat het bijplaatsen van IO-Link sensoren op alle koelinstallaties samen met een IO-Link master zeer kostelijk. Echter kan deze oplossing voor andere toepassingen die een IIoT oplossing zoeken wel een zeer interessant zijn omdat deze oplossing meestal eenvoudig bijgeplaatst kan worden op een proces zonder aanpassingen aan het proces te moeten doen. Deze oplossing werkt ook onafhankelijk van het proces, wat voor een aantal situaties een groot voordeel is .

4.1.3 “eWON” routers

Een derde mogelijke oplossing is gebaseerd op het gebruik maken van de industriële protocollen waarover de controllers beschikken. In Tabel 1 staan alle protocollen van de verschillende controllers samengevat en hieruit blijkt dat alle controllers communiceren met Modbus TCP/IP ofwel Modbus RTU.

Bij deze oplossing dienen de huidige Lancom routers vervangen te worden door routers die rechtstreeks met de controllers kunnen communiceren op basis van een industrieel protocol. Een voorbeeld van zulke routers zijn de routers van eWON, met name de “eWON Flexy” routers zoals afgebeeld op Figuur 19.



Figuur 19: “eWON Flexy” router [32]

De “eWON Flexy” routers zijn speciale routers waarbij modules met specifieke functionaliteiten op geplaatst dienen te worden. Dit zijn functionaliteiten die variëren van 4G mogelijkheid en seriële poorten tot digitale in en uitgangen. Voordelen van zulke routers zijn [32]:

- Ze kunnen data bufferen bij connectieverlies met de centrale;
- De controllers moeten niet extern toegankelijk gesteld worden;
- Data kan beschikbaar gesteld worden via API's;
- Ze beschikken over OPC-server mogelijkheden;
- Een veilige SSL-gebaseerde VPN-tunnel wordt gebruikt voor het dataverkeer;
- ...

Een ander groot voordeel van de “eWON Flexy” is dat er binnen eWON een “Danfoss IO server” bestaat. Dit is een voorbeeld over hoe er met de Danfos AK controllers gecommuniceerd dient te worden door gebruik te maken van Modbus RTU [32].

Deze oplossing is uiteindelijk toch niet gekozen omdat het vervangen van elke router te kostelijk is. Echter is deze oplossing wel belangrijk om in het achterhoofd te houden omdat deze wel interessant kan zijn voor controllers waarbij de connectie niet lukt via de gekozen manier.

4.1.4 OPC UA server

De laatste oplossing die onderzocht en na overleg ook gekozen is om connectie te kunnen maken met alle type controllers, is het gebruiken van een OPC UA server. Een OPC UA server is een communicatiesoftware die met meerdere apparaten, die elk een verschillend protocol hebben, gelijktijdig kan communiceren. Om gegevens van alle geconnecteerde apparaten met de server te kunnen lezen, is er een OPC UA Client nodig welke enkel het OPC UA protocol dient te kennen. Een uitgebreidere beschrijving van OPC UA staat beschreven in paragraaf 2.9.2 .



Figuur 20: Logo Kepware [33]

Er zijn zeer veel OPC UA server software applicaties, maar de gekozen software is Kepservers van Kepware. Deze software is voornamelijk gekozen omdat het zich al bewezen heeft in vele industriële toepassingen en hierdoor tot één van de marktleiders van OPC UA software behoort. Een tweede factor tot de keuze van de software van Kepware is dat Reditech hier zeer veel ervaring mee heeft. Dus eventuele vragen over de configuratie, kunnen binnen snel beantwoord worden door de aanwezige specialisten binnen Reditech.

Kepservers is een software die bijna met alle bestaande industriële protocollen kan communiceren. Ook is het uitgerust met geavanceerde veiligheidsfuncties en is zeer schaalbaar. De software beschikt ook een aantal optionele “Advanced Plug-ins” zoals een datalogger, een Alarm en Events functie, een IoT gateway, ... [33].

De gekozen oplossing met een OPC UA server vereist minstens de datalogger plug-in. De Datalogger functie zorgt ervoor dat data van de OPC UA server in eender welke ODBC-conforme database gelogd kan worden [33]. ODBC is een standaard manier om toegang te krijgen tot een standaard database [34]. Bij het kiezen van een geschikte database dient er dus rekening gehouden te worden dat de database voldoet aan deze standaard.

Om de oplossing zo kost efficiënt mogelijk te maken dienen er een VPN-tunnels tussen elke koelinstallatie en het hoofdkantoor van Cooltech gelegd te worden. Deze VPN-tunnels zorgen ervoor dat de controllers die op verschillende locaties staan niet enkel toegankelijk zijn op het lokale netwerk, maar ook op het netwerk van Cooltech. Hierdoor is er maar één enkele OPC UA server noodzakelijk is op het hoofdkantoor van Cooltech om elke controller te kunnen bereiken.

4.2 Stap 2: data opslaan in een database

De oplossing van stap 1 zorgt ervoor dat er zeer veel data beschikbaar komt die dient opgeslagen te worden in een database. Zo beschikt men ten alle tijden over de huidige en historische data van de koelinstallaties in een gestructureerde database.

Door analyses uit te voeren op de huidige en historische data van een koelinstallatie is het mogelijk om het gedrag van de koelinstallaties te leren. Zo kunnen er bepaalde defecten voorspeld worden, waardoor men kan ingrijpen vooraleer een defect zich voordoet. Gegevens opslaan in een database en het bijhouden van historische data is met andere woorden noodzakelijk als men functies zoals predictive maintenance mogelijk wil maken.

Er zijn verschillende soorten databases, maar één van de bekendste is een SQL-database. Dit is een zeer toegankelijke soort database, waar het beschikbaar stellen van data gebeurt via query's. Een voorbeeld van een query is: "SELECT * FROM AlarmDB WHERE value=1". Met deze query zullen alle gegevens van de alarmen die hoog zijn of waren getoond worden. Met een SQL-database is het ook mogelijk om gegevens in de database te wijzigen door een "UPDATE"-commando uit te voeren. Dit opvragen en wijzigen van gegevens kan ook gebeuren via een relatief eenvoudige applicatie.

In deze scriptie zal enkel de SQL-database aangehaald worden.

De locatie waarop de data wordt opgeslagen bepaalt of het om een Cloud of een on-premise database gaat. Beide databases worden afzonderlijk in de volgende paragrafen uitgelegd.

4.2.1 Cloud database

Een Cloud database wil zeggen dat de locatie van de server, waarop de database draait niet gekend is. Cloud databases hebben enkele voordelen ten opzichte van on-premise databases. Een aantal van deze voordelen zijn:

- Ze staan online, wat zorgt dat ze zeer toegankelijk zijn;
- Men moet zelf niet over een servers beschikken;
- Ze zijn eenvoudig uitbreidbaar;
- Ze hebben geen onderhoud nodig;
- Veilig.

Om een Cloud database te gebruiken dient er een bedrag betaald te worden per keer dat er gegevens van of naar de database geschreven worden. Dit bedrag hangt af van: het aantal datatransacties, de gewenste grootte van de database en de snelheid van dataverwerking. Op Tabel 2 is een deel van een prijslijst van een Microsoft Azure SQL database te zien. Deze tabel geeft de grote orde van de kostprijs voor het gebruiken van een Cloud database.

Tabel 2: Prijzen Standard SQL Cloud database [35]

	DTU's	INBEGREPEN OPSLAG	MAX. OPSLAGRUIMTE	PRIJS VOOR DTU'S EN INBEGREPEN OPSLAG
S0	10	250 GB	250 GB	€0,0171/uur
S1	20	250 GB	250 GB	€0,0341/uur
S2	50	250 GB	250 GB	€0,0851/uur
S3	100	250 GB	1 TB	€0,1701/uur

Eén van de bepalende kolommen voor het kiezen van een geschikte prijscategorie in bovenstaande tabel, zijn het aantal "Database Transaction Units" (=DTU's). DTU is een gecombineerde meting van CPU, geheugen en aantal gegevenstransacties [36]. Deze waarde is dus afhankelijk van het aantal transacties en de hoeveelheid data van de transacties van en naar de database.

Wat nog opvalt in Tabel 2 is dat niet overal de maximale opslagruimte gelijk is aan de inbegrepen opslagruimte. Het verschil tussen deze twee is dat men vanaf het pakket S3 meer geheugen kan gebruiken dan de database eigenlijk beschikt, maar hiervoor wordt er een bijkomende kost aangerekend.

Er zijn een aantal bekende IoT platformen die beschikken over een Cloud database. Om van alle voordelen van IIoT te kunnen profiteren is men meestal beter af met een Cloud database, maar dit is niet noodzakelijk.

4.2.2 On-premise database

Een on-premise database wil zeggen dat de database draait op een server in het bedrijf. Het voordeel hiervan is dat een bedrijf ten alle tijden weet waar hun data staat en het zoveel veiligheidsmaatregelen als het wenst kan toepassen. Dit wil echter niet zeggen dat een on-premise database altijd veiliger is, integendeel zelfs. Meestal zijn Cloud databases het beste beveiligd tegen het lekken van data naar externe partijen.

Het voordeel van een On-premise database is dat men weet waar de data staat en het lokaal binnen het bedrijf wordt gehouden. De database is daarom ook zeer toegankelijk voor iedereen op het lokale netwerk omdat ook de server op dit lokale netwerk draait. Zo heeft iedereen met de juiste rechten en toegang tot het lokale netwerk, automatisch toegang tot de server. Voor Cloud databases moet er echter altijd voor een veilige connectie met het internet gezorgd worden.

On premise databases worden vaak als goedkoper geacht omdat er bij dit soort databases niet per datatransfer betaald dient te worden. Echter moeten de onderhoudskosten, updatekosten, hardware kosten en personeelskosten hier ook wel mee in rekening gebracht worden. Wat ervoor zorgt dat er situaties zijn waar een Cloud database wel goedkoper is.

Het uitbreiden van het geheugen van een on-premise database is meestal ook moeilijker omdat er hiervoor de nodige plaats moet zijn. Een server neemt namelijk fysieke plaatst in. Ook is het mogelijk dat bij het uitbreiden de server even uit ligt, wat niet zo is bij een Cloud database. Hier dient enkel een nieuw (duurder) abonnement gekozen te worden en er is meer data beschikbaar.

In de totaaloplossing is er gekozen voor een on-premise database. De beslissende factoren hier waren:

- Cooltech beschikt al over een server;
- Dit is goedkoper in gebruik tijdens de masterproef
- Eenvoudig opzetten van SQL database met behulp van “Microsoft SQL server express”;
- Het is voldoende om de “proof of concept” van de masterproef aan te tonen.

4.3 Stap 3: rapportage van data in Microsoft Excel

Als derde stap tot het bekomen van een totaaloplossing dient er gezorgd te worden dat de data, die in de database staat, getoond kan worden in een Excel-sheet. Deze stap dient voornamelijk om aan te tonen dat, eens de data in een database staat, het zeer eenvoudig is om een lijst van gegevens te tonen via Microsoft Excel. Op basis van deze lijst kunnen er vervolgens besluiten en eventuele grafieken of histogrammen gemaakt worden.

Om het gebruik van deze Excel-sheet gebruiksvriendelijker te maken is het noodzakelijk dat deze beschikt over filtermogelijkheden. Zo is het mogelijk om enkel de gewenste waarden te tonen, wat het overzichtelijker maakt..

4.4 Stap 4: vb.net applicatie op de database

Een Lijst in Excel is niet optimaal als er uiteindelijk data over meerdere koelinstallaties in staat. Want dit zal zorgen voor een overvloed aan informatie, waardoor er geen overzicht meer is. Daarom is het noodzakelijk om een applicatie te ontwikkelen die al deze data bundelt en visueel maakt. Zo zal het veel eenvoudiger zijn om een overzicht te verkrijgen van de vele koelinstallaties.

Als applicatie is er gekozen voor een vb.net applicatie omdat deze op een eenvoudige manier kan communiceren met een SQL database. Bij dit soort applicaties kan de data rechtstreeks uit de database opgevraagd worden door middel van query's. Een vb.net-applicatie is ook een goede manier om in het kader van de proof of concept aan te tonen wat er allemaal met de data gedaan kan worden. Achteraf kan er nog altijd gekeken worden naar meer optimale oplossingen voor een applicatie.

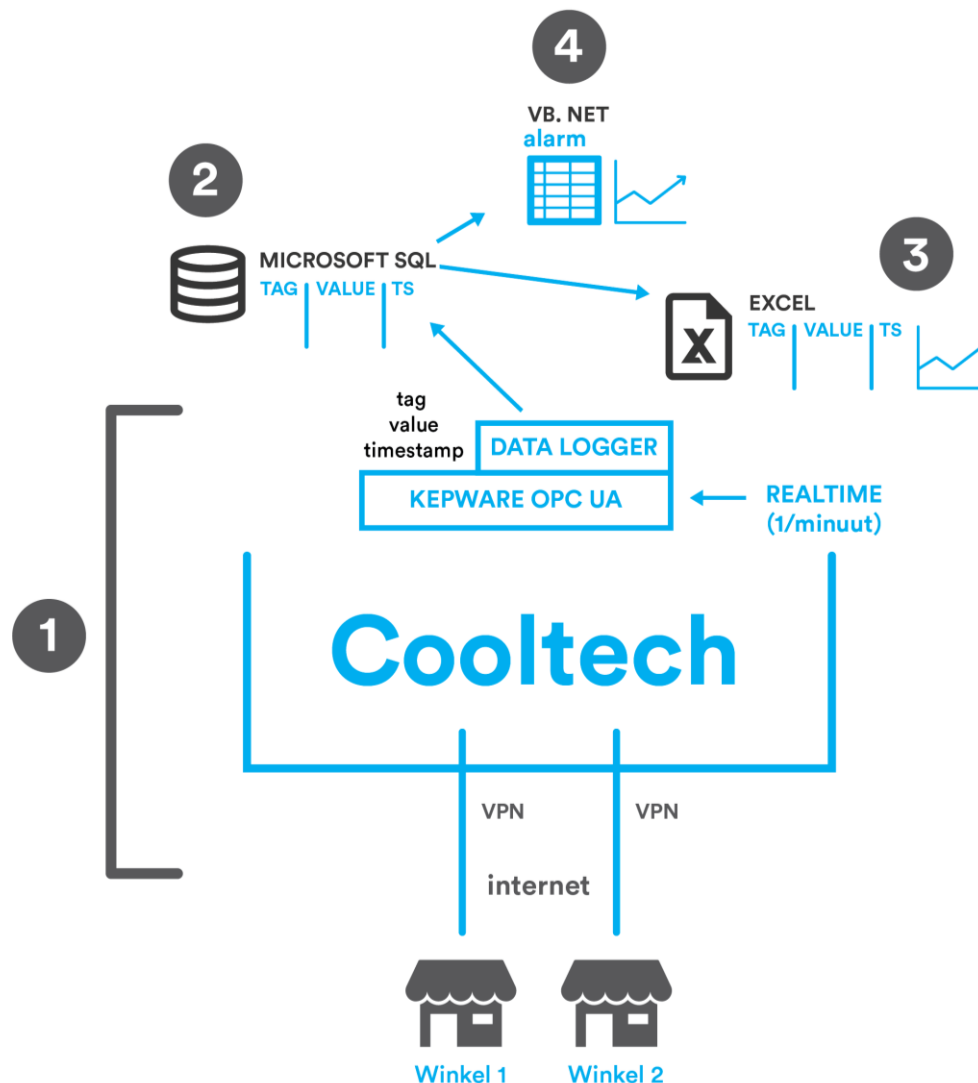
De applicatie dient zo ontwikkeld te worden dat de gebruiker een duidelijk en eenvoudig overzicht krijgt van alle gewenste gegevens.

De applicatie is volledig geprogrammeerd in "Visual Studio". Dit is een veelgebruikt programma om allerlei applicaties te programmeren, wat het zeer bekend maakt. Hierdoor kunnen er achteraf ook nog altijd bijkomende functies voor de applicatie ontwikkeld worden door andere personen met de juiste kennis.

Ten slotte kunnen bovenop de applicatie nog enkele rapporten ontwikkeld worden met Microsoft Reporting services. Dit kan net zoals de vb.net applicatie de data van de database opvragen via query's. Het voordeel van deze rapporten is dat er vlug en eenvoudig een overzicht gemaakt kan worden van bijvoorbeeld alle alarmen voorgekomen tussen een bepaalde periode opgesomd en in een histogram geplaatst of het verloop van de temperatuur over een bepaalde tijdsperiode. Deze rapporten kunnen geraadpleegd worden via een internetbrowser.

4.5 Gekozen totaaloplossing

Alle gekozen methodes voor het aanpakken van de verschillende stappen zijn samengevoegd en staat geïllustreerd op Figuur 21. Op deze figuur zijn ook alle stappen genummerd.



Figuur 21: Illustratie gekozen totaaloplossing

Bij stap 1 is te zien dat er maar één licentie van een OPC UA server noodzakelijk is om te kunnen verbinden met meerdere koelinstallaties door gebruik te maken van VPN-tunnels van Cooltech naar de locaties van de koelinstallaties. Ook dat de software van de OPC UA server van Kepware is, welke beschikt over een datalogger functie om tags in een database te plaatsen.

In stap 2 gaat het om een Microsoft SQL database. Als database is hier gekozen voor een Microsoft SQL server database, wat een on-premise database is. In deze totaaloplossing is het achteraf ook mogelijk om over te schakelen van een on-premise naar een cloud database zoals deze van Microsoft Azure. Dit kan zonder dat er aan de rest van de methodes van de totaaloplossing aanpassingen dienen te gebeuren. Enkel aan de datalogger van keppure en de applicatie dient hier de Microsoft SQL server database gewijzigd te worden naar een Microsoft Azure database.

De twee volgende stappen zijn duidelijk en kort. Hier gaat het om een tussenstap om gegevens op te lijsten in Excel en het maken van een vb.net applicatie als visueel dashboard om alle data van de database op een overzichtelijke manier zichtbaar te maken.

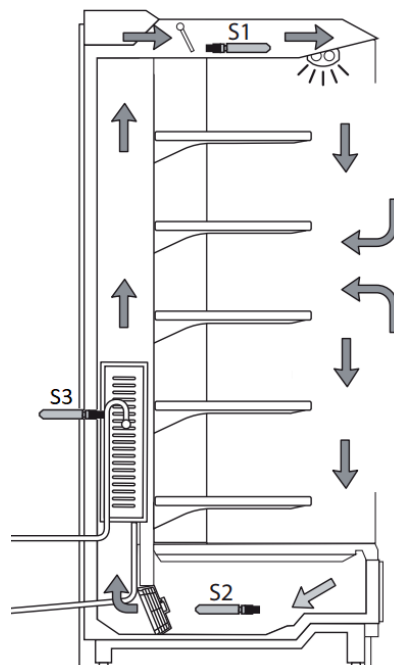
5 Simulatie

Er is een simulatie gemaakt, die de werking van een werkelijke koelinstallatie op een vereenvoudigde manier nabootst door de belangrijkste temperaturen en drukken te simuleren. Het maken van deze simulatie is om aan te tonen dat de totaaloplossing met de OPC UA server werkt indien de controller connectie en het uitlezen van gegevens toestaat. Ook helpt de simulatie met het sneller implementeren van de totaaloplossing op een werkelijke installatie. Dit omdat alle fases van de totaaloplossing uitgewerkt en getest zijn op de simulatie en dus al werkend zijn voor de werkelijke implementatie.

De simulatie is geprogrammeerd met Tia portal, wat het programma is om Siemens PLC's te programmeren. Hiervoor is in eerste instantie een gesimuleerde S7-1200 PLC gebruikt. Achteraf wanneer de simulatie klaar was, is er voor het testen van de oplossing met de OPC UA server een effectieve S7-1214C PLC gebruikt.

5.1 Werking koelinstallatie

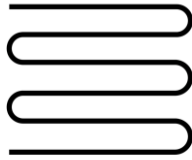
Vooraleer de simulatie van een koelinstallatie gemaakt kan worden, is het noodzakelijk om de basiswerking van een koelinstallatie te kennen. Dit zal uitgelegd worden aan de hand van Figuur 22. Dit is een vereenvoudigde weergave van een koelmeubel in dat terug te vinden is in de supermarkten, waarop de belangrijkste componenten om de basiswerking van een koelinstallatie te begrijpen op te zien zijn.



Figuur 22: Werking koelmeubel [37]

Het volledige koelcircuit is niet op de figuur weergegeven. Hier staat enkel de verdampers, wat het belangrijkste onderdeel van een koelcircuit is weergegeven.

Een verdamper is een lange buis geplooid zoals weergegeven op Figuur 23 om zoveel mogelijk oppervlak te bedekken. Doorheen deze buis stroomt een koelmiddel met een lage temperatuur. Doordat de temperatuur van het koelmiddel in de verdamper lager is dan te koelen lucht, zal de verdamper warmte onttrekken en de lucht afkoelen.



Figuur 23: Illustratie verdamper

De circulatie van de lucht doorheen het koelmeubel staat met pijlen weergegeven op Figuur 22. De extra pijlen die van buitenaf naar binnen komen, geven allemaal de warmtestroom die van buitenaf naar binnen stroomt weer. Op de figuur is er te zien dat de lucht wordt aangezogen door een ventilator, welke de lucht vervolgens langs de verdamper stuurt, waar het zal afkoelen. Deze gekoelde lucht wordt tenslotte in het meubel geblazen en zal zo producten afkoelen. Dit is de basis werking hoe een koelmeubel gekoeld wordt.

5.2 Verloop temperaturen en openen expansieventiel

Om het verloop van de temperaturen te kunnen begrijpen, moet er gekeken worden naar de drie probes (S1-S3) afgebeeld op Figuur 22 en de opening van het expansieventiel. De probes meten specifieke temperaturen en de opening van het expansieventiel geeft aan in procent hoeveel het expansieventiel geopend is. 100% wil zeggen dat de toevoer van het koelmiddel volledig geopend is en er een maximale stroom van vloeistof in de verdamper wordt geïnjecteerd. 0% wil zeggen dat de toevoer afgesloten is en er geen actieve koeling kan plaatsvinden.

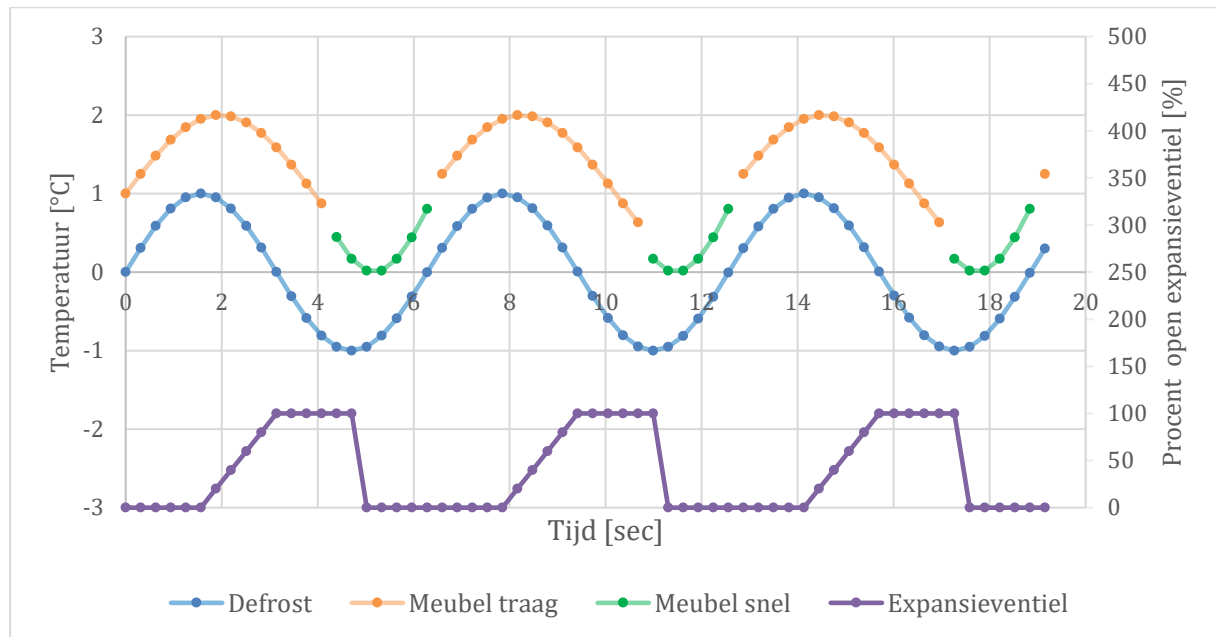
S1 meet de temperatuur van de gekoelde lucht die in het meubel geblazen wordt. S2 meet de temperatuur van de lucht die uit meubel gezogen wordt. Een weging van S1 en S2 geeft een benaderende temperatuur van het koelmeubel. Ten slotte is er S3, welke de temperatuur van de uitgaande buis van de verdamper meet.

In het verdere verloop van dit hoofdstuk zal de gemiddelde meubel temperatuur gezien worden als de effectieve meubel temperatuur. De temperatuur van de verdamper zal “defrosttemperatuur” genoemd worden.

Het verloop van de temperaturen en de hierbij gepaard gaande opening van het expansieventiel, is voor twee voorkomende werkingstoestanden ontworpen in de simulatie. Namelijk de normale werking en de ontdooicyclus. Wat deze werkingstoestanden doen en het verloop van de temperaturen en de opening van het expansieventiel is, staan in de volgende paragrafen beschreven.

5.2.1 Normale werking

Op Figuur 24 is het verloop van de variabelen tijdens de normale werking van een koelinstallatie te zien. De normale werking wil zeggen dat de installatie aan het koelen is en er geen problemen zijn. Het verloop dat te zien is, is dat van de simulatie wat fel lijkt op het werkelijke verloop. Bij het werkelijke verloop zijn er meer onregelmatigheden, maar om het basisprincipe van de werking en het verloop van de variabelen uit te leggen is eenvoudiger aan de hand van gesimuleerde waardes. Aangezien deze lopen hoe het theoretisch zou moeten.



Figuur 24: Verloop temperaturen en opening expansieventiel normale werking

Op de grafiek van het normale verloop zijn er vier gegevensreeksen te zien:

- Defrost,
- Meubel traag,
- Meubel snel,
- Expansieventiel.

De defrosttemperatuur ofwel de temperatuur van de verdamper heeft een sinusoïde vorm en geeft aan wat de temperatuur van het meubel zal moeten doen. Als deze stijgt dan zal de meubeltemperatuur ook stijgen, maar dit gebeurt vertraagd omdat temperatuurapplicaties nooit onmiddellijk kunnen reageren. Om dit vertraagde effect te kunnen simuleren zijn er 2 verschillende sinussen gebruikt voor de meubeltemperatuur. Eén is trager dan de grondfrequentie van de defrosttemperatuur en de andere is sneller. Ze wisselen onderling continue af om zo het vertraagde effect te creëren. Bij een werkelijke installatie zal deze ook nog na-ijlen op de opening van het expansieventiel. Dit is echter niet in de simulatie geïmplementeerd omdat de simulatie voornamelijk dient om de werking van de totaaloplossing aan te tonen.

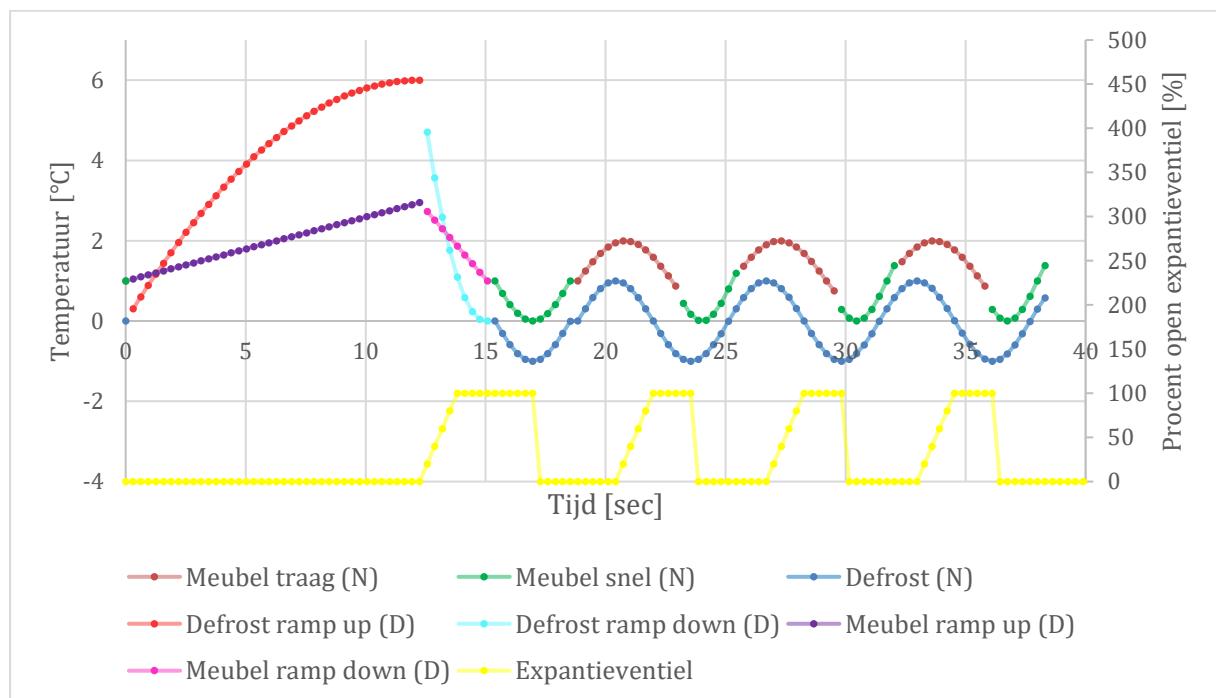
De meubeltemperatuur zal continue schommelen tussen twee waarden met in het midden de set waarde. Dit schommelen komt door het pulserend gedrag van het expansieventiel en zou niet gebeuren als de opening juist afgesteld is. Als de meubeltemperatuur zijn maximale temperatuur heeft bereikt van set waarde + ΔY , dan dient de defrosttemperatuur te dalen waardoor de meubeltemperatuur vertraagd zal volgen.

Om de defrosttemperatuur te kunnen laten dalen dient er koelmiddel doorheen de verdamper gestuurd te worden. Hier zal het expansieventiel geleidelijk meer en meer openen tot het 100% geopend is. Hierdoor gaat er meer en meer koelmiddel stromen waardoor de verdamper zal afkoelen en dus de defrosttemperatuur zal dalen.

Als de meubeltemperatuur zijn minimale waarde van set waarde - ΔY bereikt heeft, dan sluit het expansieventiel volledig en stroomt er geen koelvloeistof meer. Hierdoor zal de defrosttemperatuur weer geleidelijk gaan stijgen totdat de meubeltemperatuur weer zijn maximum bereikt heeft.

5.2.2 Ontdooi cyclus

De cyclus van de normale werking blijft zich herhalen. Echter na enige tijd zal er ijsvorming ontstaan op de onderkoelde verdamper door het vocht in de te koelen lucht. Dit zorgt ervoor dat als hier niets aan gebeurt de verdamper kan dichtvriezen. Wat ervoor zorgt dat de koelinstallatie niet meer kan koelen omdat de lucht wordt geblokkeerd. Om het dichtvriezen te voorkomen moet tijdig een ontdooicyclus uitgevoerd worden waarbij de verdamper zal ontdooien en het ijs zal smelten.



Figuur 25: Verloop temperaturen en opening expansieventiel ontdooicyclus

Figuur 25 toont het verloop van de variabelen tijdens de ontdooicyclus. Voor en na een ontdooicyclus is het verloop van de variabelen zoals tijdens de normale werking. De ontdooicyclus is enkel het gedeelte dat de temperatuur tot 6-8°C stijgt zodat het ijs op de verdamper kan ontdooien.

Op Figuur 25 zijn zowel gegevensreeksen te zien van de normale werking als van de ontdooicyclus. De werkingstoestand waartoe de gegevensreeksen behoren zijn aangegeven door middel van een letter achter de naam. Achter de gegevensreeksen van de normale werking staat een (N) en achter deze van de ontdooicyclus staat een (D). De gegevensreeksen van de ontdooicyclus en waar in deze paragraaf op gefocust wordt zijn:

- Defrost ramp down,
- Defrost ramp down,
- Meubel ramp up,
- Meubel ramp down,
- Expansieventiel.

Zoals bij de meubeltemperatuur bij de normale werking is hier de defrost- en de meubeltemperatuur opgesplitst in twee delen. Eén deel waarin ze zullen stijgen (ramp up) en een ander deel waarin ze terug dalen (ramp down).

In het deel waar de temperatuur stijgt zal de defrosttemperatuur groter worden dan deze van het meubel zodat het gevormde ijs rond de verdamper kan smelten. Het laten stijgen van de defrosttemperatuur gebeurt met behulp van verwarmingsweerstandens zodat dit proces sneller gaat. Als de defrosttemperatuur aan het stijgen is zal de meubeltemperatuur ook geleidelijk aan stijgen, maar dit gaat trager omdat hier geen verwarmingsweerstandens gebruikt worden. Het ontdooien mag nooit langer duren dan een bepaalde tijd anders kan de temperatuur van het meubel te hoog worden. Wat zeer ongewenst is voor de voedingsproducten, bewaard in de koelinstallaties. Merk ook op dat gedurende de tijd dat de temperatuur aan het stijgen is het expansieventiel volledig gesloten blijft.

In het dalende deel van de defrostcyclus, zal het expansieventiel open gaan en zal de temperatuur van de verdamper sterk dalen. Dit dalende proces gaat door totdat de meubeltemperatuur terug zijn minimale waarde (set waarde - ΔY) heeft bereikt. Waarna het de normale cyclus voortzet zoals beschreven in de vorige paragraaf. Tijdens het dalen gaat het expansieventiel weer geleidelijk aan open totdat het zijn maximale waarde bereikt heeft.

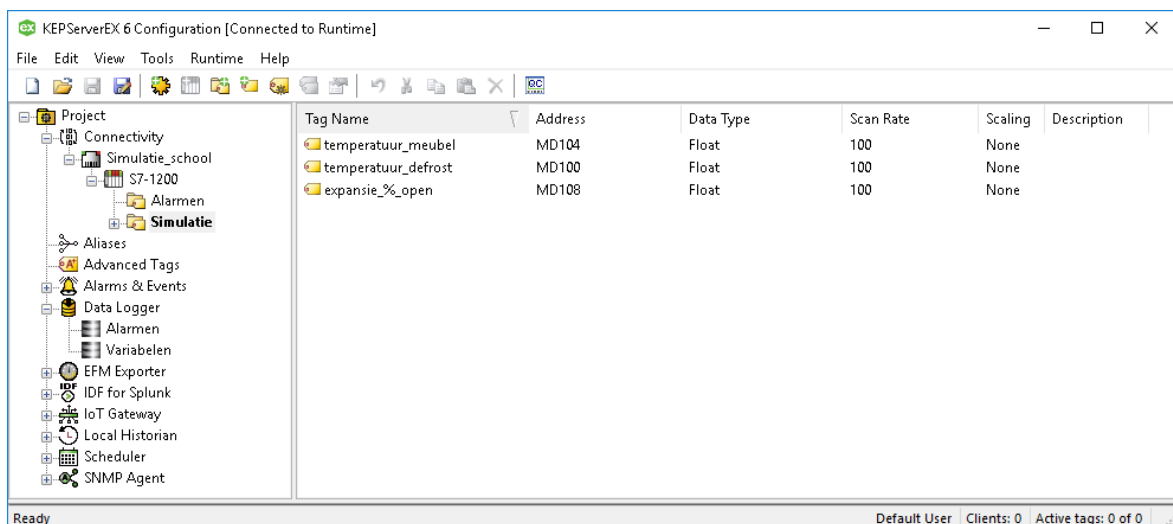
5.3 OPC UA en database

Het connecteren van een installatie met de OPC UA server is de belangrijkste stap in de gekozen totaaloplossing. Vooraleer dit kan, dient de simulatie op een effectieve PLC met een vast IP-adres te werken. Zoals al eerder beschreven is voor de OPC UA server gekozen om Kepserverex, een OPC UA server software van Kepware te gebruiken.

In deze paragraaf wordt besproken hoe de configuratie van Kepserverex en de datalogger met een database van Microsoft SQL server is gebeurd met de simulatie.

5.3.1 Configuratie OPC UA server

Op Figuur 26 is een print screen te zien van hoe de configuratie van de OPC UA server eruit ziet binnen Kepserverex. Aan de hand van deze figuur zal er uitgelegd worden hoe de configuratie is gebeurd.



Figuur 26: Print screen Kepware simulatie

In Kepserverex moet er allereerst een connectie aangemaakt te worden. Dit dient om aan te geven welk protocol de software moet gebruiken om de controller te kunnen bereiken. Aangezien de simulatie op een S7-1214 PLC van Siemens werkt, moet “Siemens TCP/IP Ethernet” als communicatieprotocol gebruikt worden. De naam van de connectie kan gekozen worden [38, pp. 134-136]. Op Figuur 26 heeft de aangemaakte connectie de naam “Simulatie_school”.

Nadat de connectie aangemaakt is, dient het apparaat waarmee de server moet communiceren toegevoegd te worden aan de connectie. De configuratie van dit apparaat hangt af van het gebruikte communicatieprotocol en het apparaat. Bij de gebruikte Siemens PLC een model en ID opgegeven te worden. Het model is hier S7-1200 en het ID is het vaste IP-adres van de controller. Ook hier kan het apparaat een gekozen naam hebben [38, pp. 136-138]. Op Figuur 26 heeft het aangemaakte apparaat de naam “S7-1200”.

De laatste stap van de configuratie is het aanmaken van “User Defined Tags”. Dit zijn de variabelen van de controller die gelogd moeten kunnen worden. Afhankelijk van het apparaat en het type variabele krijgen de tags een bepaalde codering voor het adres [38, pp. 138-144]. Bij Siemens apparaten wordt dit voor de Booleaanse variabelen met bijvoorbeeld het PLC-adres M10.7, MX10.7 als tag adres in de Kepware software. Voor woord en decimale variabele is het te gebruiken adres in Kepserverex hetzelfde als het PLC-adres.

De PLC-adressen kunnen bij Siemens PLC's terug gevonden worden in de Tag tabel van het PLC programma. Een voorbeeld van een tag tabel is te zien op Tabel 3. Merk op dat hier een kolom is waarin aangevinkt kan worden als de variabele toegankelijk zijn via OPC UA. Zo kunnen enkel de variabelen die geen schade aan de werking van de installatie kunnen verrichten toegankelijk gemaakt worden. Ook is er een kolom die aangeeft als de tags “writable” zijn via OPC UA. Zo kan er onderscheid gemaakt worden tussen “read only” en “read/write” variabelen.

Tabel 3: PLC variabelen simulatie

cooltechTagTable							
	Name	Data type	Address	Retain	Accessible from HMI/OPC UA	Writable from HMI/OPC U	
1	temp_defrost	Real	%MD100	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
2	temp_meubel	Real	%MD104	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
3	expansieventiel_%_open	Real	%MD108	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
4	start_normaleWerking	Bool	%M0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5	start_defrost_normaal	Bool	%M0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
6	start_defrost_overTijd	Bool	%M0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7	Alarm_deurOpen	Bool	%M10.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8	Alarm_ventilatie	Bool	%M10.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
9	Alarm_defrostTijd	Bool	%M10.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
10	NF_start_norm	Bool	%M90.0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
11	NF_start_defr	Bool	%M90.1	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
12	start_sim	Bool	%M1.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
13	start_ontdooi	Bool	%M1.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
14	normaleWerking_bsy	Bool	%M2.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
15	voorbereidenOntdooi_bsy	Bool	%M2.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
16	Ontdooi_bsy	Bool	%M2.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
17	NF_start_sim	Bool	%M90.2	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Scan rate

De OPC UA server biedt ook de mogelijkheid om elke tag een andere "Scan Rate" te geven. "Scan Rate" wilt zeggen om de hoeveel tijd de OPC UA server de waarde van een tag gaat updaten. Het is belangrijk dat deze juist ingesteld wordt want als deze te hoog of te laag is kan het nadelige effecten met zich meebrengen. Een hoge "Scan Rate" wil zeggen dat er tegen een hoge frequentie data geüpdatet wordt en een lage "Scan Rate" wil zeggen dat dit tegen een lage frequentie gebeurt.

Als de "Scan Rate" te laag is, dan kan het anti-aliasing effect zich voordoen. Dit zorgt ervoor dat het ontvangen signaal vervormd is en niet meer lijkt op het oorspronkelijke signaal. De interpretatie van het signaal is hierdoor meestal verkeerd, waardoor er foute conclusies genomen zullen worden. Volgens het Niquist theorema moet de sample frequentie minstens tweemaal de hoogste frequentie die voorkomt in een signaal zijn, om anti-aliasing te voorkomen. Met andere woorden dient de "Scan Rate" minstens tweemaal de frequentie van het hoogst frequente signaal dat men wil kunnen detecteren te zijn.

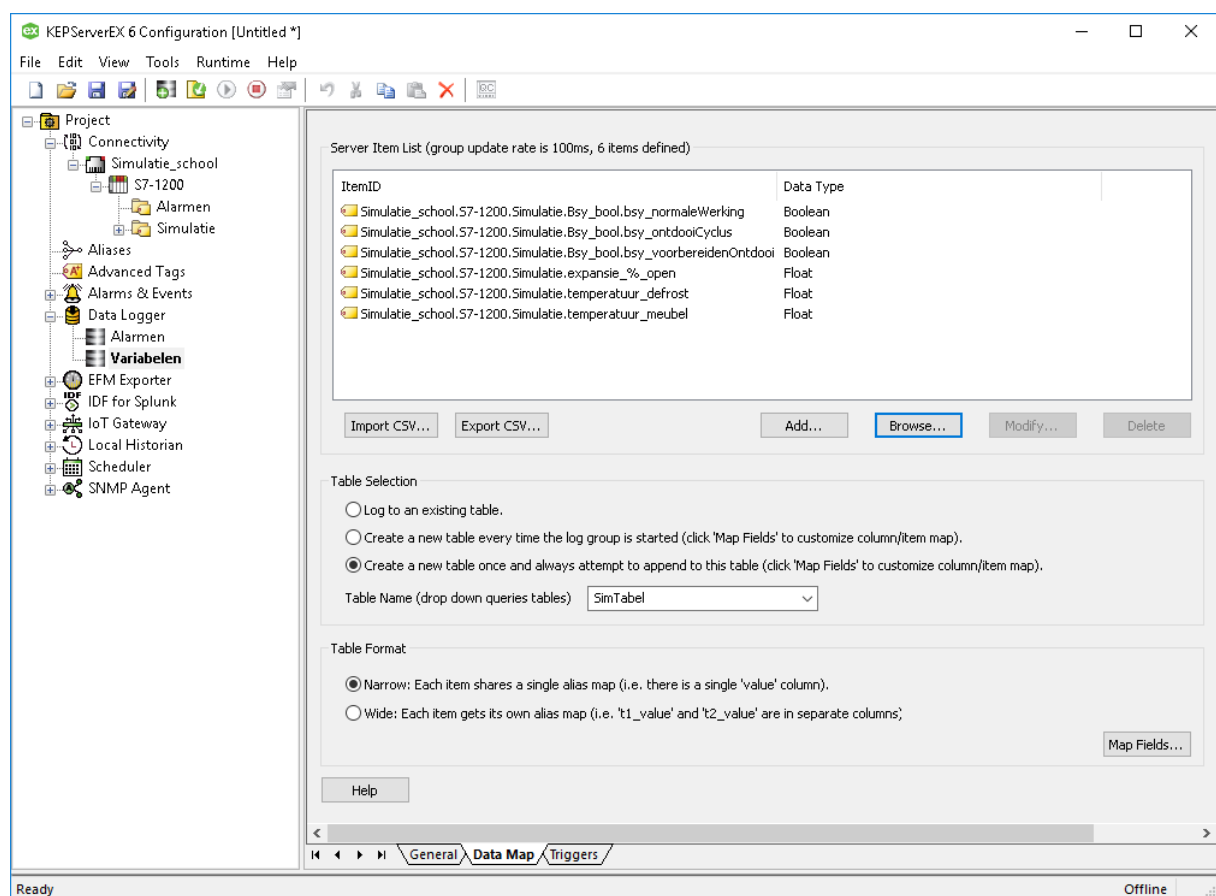
Een zeer hoge "Scan Rate" zorgt er dus voor dat er geen anti-aliasing kan voorkomen en de resolutie van het signaal zeer hoog is, maar als deze te hoog is dan kan dit ongewenste kosten met zich meebrengen. Want data versturen via het internet kost geld. Als er bijvoorbeeld om de 100 milliseconden 1000 variabelen geüpdatet worden zal de datakost snel oplopen.

Er moet dus een "Scan Rate" gekozen worden die groot genoeg is zodat er geen anti-aliasing kan voorkomen, maar niet te groot is want anders zal de datakost hoog oplopen. Voor een minimale kost kiest men best voor een "Scan Rate" die net genoeg is om geen anti-aliasing te hebben. Echter is het meestal beter om een iets snellere "Scan Rate" te nemen zodat de resolutie van het signaal vergroot wordt. Hier dient men een afweging te maken tussen kostprijs en resolutie.

5.3.2 Configuratie datalogger en database

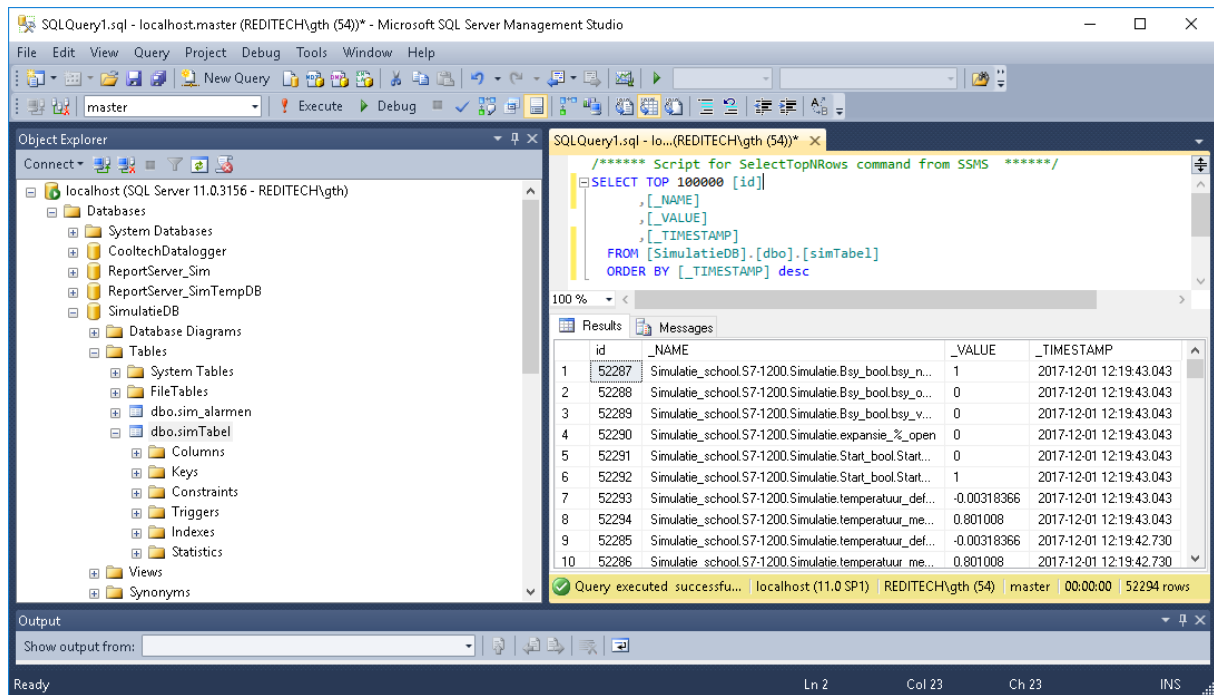
De datalogger is functie dat toelaat om variabelen te loggen in een ondersteunde database. Om deze functie te kunnen gebruiken dient er in eerste een database gekoppeld te worden. Hiervoor is het noodzakelijk om een DSN te configureren. Deze configuratie staat in voor de verbinding met een specifieke database op een server. In de DSN zit ook de login ID en het paswoord om te kunnen verbinden met de SQL-server.

Na het configureren van de DNS kan er een selectie gemaakt worden uit de aangemaakte "User Defined Tags" die de datalogger dient te loggen in een tabel van de database. Deze selectie is te zien op Figuur 27. Op deze figuur is ook te zien dat de datalogger de tags zal loggen in een tabel met de naam "simTabel". Hij zal deze Tabel zelf één keer aanmaken in de database en achteraf in deze tabel alle data dumpen.



Figuur 27: Print screen selectie tags datalogger

Als de datalogger gegevens van de simulatie in de database geplaatst heeft, dan kunnen deze gegeven op een zeer eenvoudige manier opgevraagd worden door middel van een query. Figuur 28 toont hoe door middel van een query de meest recente data van de simulatie getoond kan worden.



The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left displays the database structure for 'localhost (SQL Server 11.0.3156 - REDITECH\gth)', including the 'SimulatieDB' database and the 'dbo.simTabel' table. The SQL Query window in the center contains the following query:

```
/****** Script for SelectTopNRows command from SSMS *****/  
SELECT TOP 10000 [id]  
      ,[_NAME]  
      ,[_VALUE]  
      ,[_TIMESTAMP]  
FROM [SimulatieDB].[dbo].[simTabel]  
ORDER BY [_TIMESTAMP] desc
```

The Results pane at the bottom shows the top 10 rows of data from the 'simTabel' table, ordered by timestamp in descending order. The data is as follows:

id	_NAME	_VALUE	_TIMESTAMP
52287	Simulatie_school57-1200.Simulatie.Bsy_bool.bsy_n...	1	2017-12-01 12:19:43.043
52288	Simulatie_school57-1200.Simulatie.Bsy_bool.bsy_o...	0	2017-12-01 12:19:43.043
52289	Simulatie_school57-1200.Simulatie.Bsy_bool.bsy_v...	0	2017-12-01 12:19:43.043
52290	Simulatie_school57-1200.Simulatie.expansie_%_open	0	2017-12-01 12:19:43.043
52291	Simulatie_school57-1200.Simulatie.Start_bool.Start...	0	2017-12-01 12:19:43.043
52292	Simulatie_school57-1200.Simulatie.Start_bool.Start...	1	2017-12-01 12:19:43.043
52293	Simulatie_school57-1200.Simulatie.temperatuur_def...	-0.00318366	2017-12-01 12:19:43.043
52294	Simulatie_school57-1200.Simulatie.temperatuur_me...	0.801008	2017-12-01 12:19:43.043
52285	Simulatie_school57-1200.Simulatie.temperatuur_def...	-0.00318366	2017-12-01 12:19:42.730
52286	Simulatie_school57-1200.Simulatie.temperatuur_me...	0.801008	2017-12-01 12:19:42.730

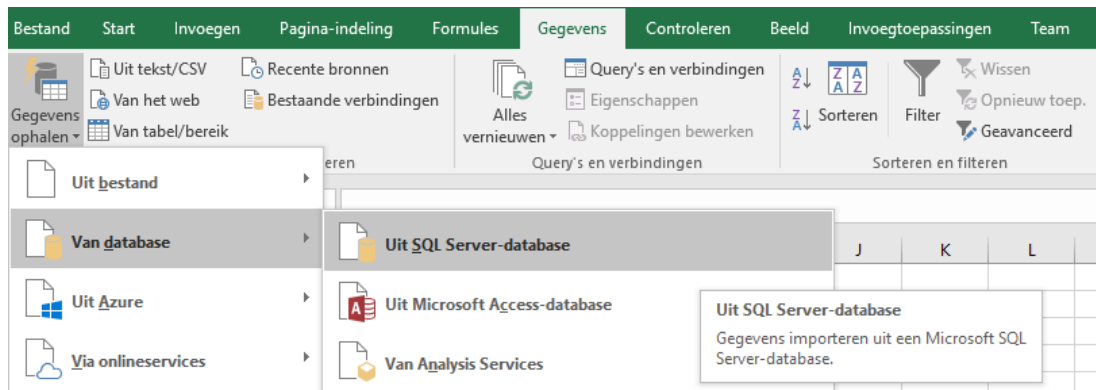
The status bar at the bottom indicates that the query was executed successfully, returning 52294 rows.

Figuur 28: Print screen data in database

5.4 Rapportage in Microsoft Excel

Deze stap dient zoals eerder besproken om aan te tonen dat er een eenvoudige rapportage van alarmen en procesgegevens gemaakt kan worden via Microsoft Excel.

De rapportage gebeurt door allereerst de gegevens van de database in te laden in een Excelsheet. Het inladen van data uit een SQL-database is zeer eenvoudig. Om dit te doen, dient er zoals getoond op Figuur 29 enkel te geklikt te worden op gegevens ophalen uit SQL server database en de naam van de server en de database in te vullen.



Figuur 29: Print screen data uit database Excel

Eens dat de naam van de server en de database zijn ingevuld, dient de naam van de tabel die men wil importeren geselecteerd te worden. Als dit gebeurd is, zal Excel automatisch een lijst genereren met alle gegevens die in de opgegeven tabel staan. Bij deze lijst zijn automatisch de nodige filtermogelijkheden om enkel de gewenste waarden zichtbaar te maken aanwezig. Op Figuur 30 is een voorbeeld van een automatisch gegenereerde lijst met filtermogelijkheden te zien, met variabelen van de simulatie.

	A	B	C	D	E
8	52293	Simulatie_school.S7-1200.Simulatie.temperatuur_defrost		0 -0.00318366	1/12/2017 12:19
9	52294	Simulatie_school.S7-1200.Simulatie.temperatuur_meubel		0 0.801008	1/12/2017 12:19
13	52280	Simulatie_school.S7-1200.Simulatie.expansie_%_open		0 0	1/12/2017 12:19
16	52285	Simulatie_school.S7-1200.Simulatie.temperatuur_defrost		0 -0.00318366	1/12/2017 12:19
17	52286	Simulatie_school.S7-1200.Simulatie.temperatuur_meubel		0 0.801008	1/12/2017 12:19
21	52272	Simulatie_school.S7-1200.Simulatie.expansie_%_open		0 0	1/12/2017 12:19
24	52283	Simulatie_school.S7-1200.Simulatie.temperatuur_defrost		0 -0.311892	1/12/2017 12:19
25	52284	Simulatie_school.S7-1200.Simulatie.temperatuur_meubel		0 0.44129	1/12/2017 12:19
29	52264	Simulatie_school.S7-1200.Simulatie.expansie_%_open		0 0	1/12/2017 12:19
32	_NAME				
33	Simulatie_school.S7-1200.Simulatie.Bsy_bool.bsy_normaleWerking				
37	Simulatie_school.S7-1200.Simulatie.Bsy_bool.bsy_ontdooiCyclus				
40	Simulatie_school.S7-1200.Simulatie.Bsy_bool.bsy_voorbereidenOntdooi				
41	Simulatie_school.S7-1200.Simulatie.Bsy_bool.bsy_voorbereidenOntdooi				
45	Simulatie_school.S7-1200.Simulatie.Bsy_bool.bsy_voorbereidenOntdooi				
48	Simulatie_school.S7-1200.Simulatie.expansie_%_open				
49	Simulatie_school.S7-1200.Simulatie.Start_bool.Start_ontdooi_cyclus				
53	Simulatie_school.S7-1200.Simulatie.Start_bool.Start_ontdooi_cyclus				
56	Simulatie_school.S7-1200.Simulatie.Start_bool.Start_simulatie				
57	Simulatie_school.S7-1200.Simulatie.Start_bool.Start_simulatie				
57	Simulatie_school.S7-1200.Simulatie.temperatuur_defrost				
61	Simulatie_school.S7-1200.Simulatie.temperatuur_defrost				
64	Simulatie_school.S7-1200.Simulatie.temperatuur_meubel				
64	Simulatie_school.S7-1200.Simulatie.temperatuur_meubel				

Figuur 30: Print screen rapportage Excel met filtermogelijkheden

5.5 Vb.net applicatie

Ten slotte is de vb.net applicatie gemaakt, welke dient als universeel platform. Hier komen alle alarmen en een aantal belangrijke procesvariabelen van verschillende koelinstallaties in samen. De applicatie is zodanig ontwikkeld dat de gezondheid en status van de koelinstallaties op een snelle en visuele manier geraadpleegd kunnen worden. Er zijn twee verschillende schermen ontwikkeld om de mogelijkheden te tonen wat mogelijk is met de gekozen totaaloplossing.

Eén van de ontwikkelde schermen is afgebeeld op Figuur 31. Dit is een dashboard dat algemene informatie toont over 15 installaties. Dit kan later uiteraard uitgebreid worden naar veel meer installaties. Op dit scherm zijn per installatie twee dingen te zien: de status van de installatie en of er al dan niet een alarm aanwezig is. De status geeft aan als een installatie in bedrijf, ontdooicyclus of niet actief is. Hieraan kan bijvoorbeeld later ook nog een status van onderhoud aan toegevoegd worden. Het vakje van alarm van een installatie zal rood kleuren wanneer er één of meerdere alarmen actief zijn van de betreffende installatie. Het is een indicatie van een algemeen alarm.

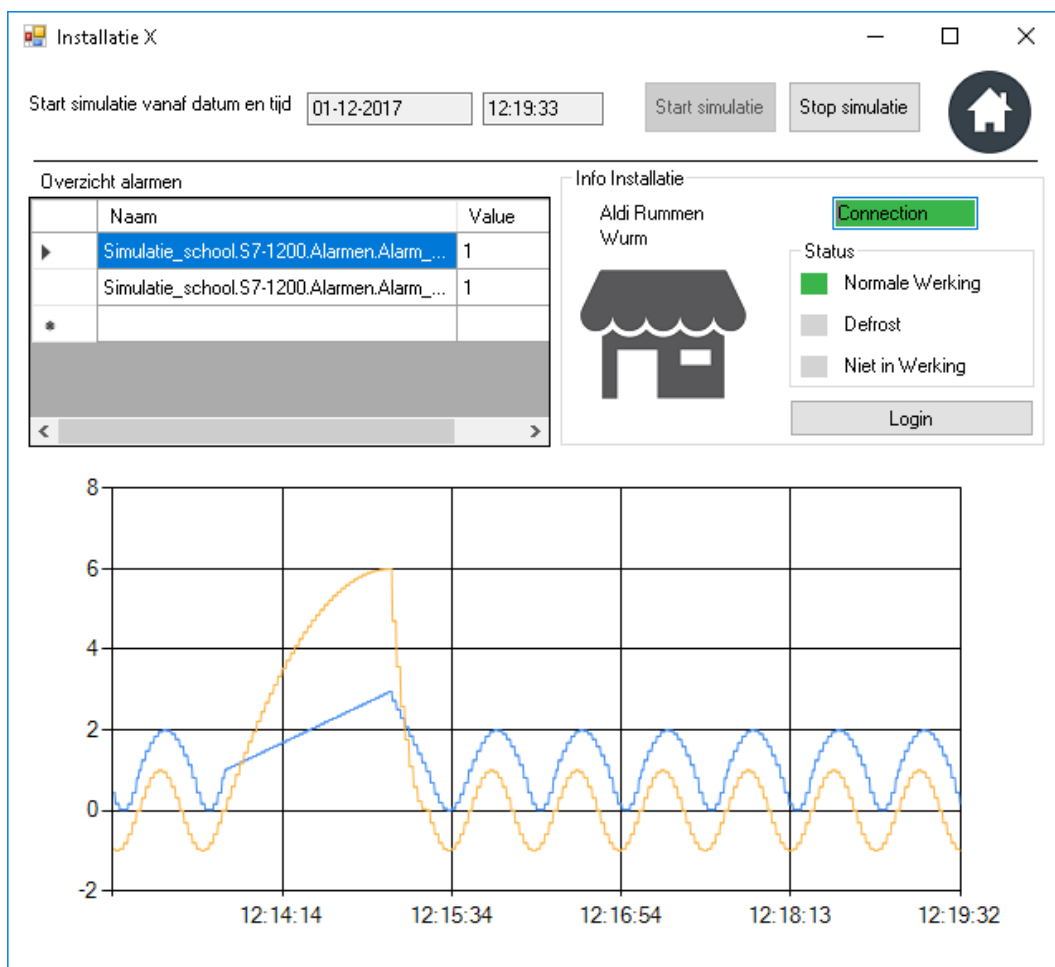


Figuur 31: Dashboard vb.net applicatie

Het tweede scherm dat ontwikkeld is, is uniek voor elke installatie en opent wanneer er op het status of alarm vlak van een installatie gedrukt wordt. Dit scherm laat de huidige alarmen en belangrijke variabelen van een specifieke installatie zien. Hierop is ten eerste een overzicht van alle actieve alarmen en de status van de installatie te zien. Ook bevat dit scherm meer informatie over de installatie en waar deze gelegen is.

De monitoring van data op dit scherm toont standaard de live waardes van de installatie. Als er echter een startdatum en tijd ingevuld worden in de bovenstaande tekstvlakken en er vervolgens op de knop start simulatie gedrukt wordt, zal de applicatie waardes loggen vanaf het opgegeven tijdstip alsof het live waardes zijn. Om het loggen van historische waardes te stoppen dient op de knop stop simulatie gedrukt te worden.

Een voorbeeld van dit tweede scherm is te zien op Figuur 32. Om terug te gaan naar het dashboard te gaan, dient er op de home knop rechtsboven gedrukt te worden. Als dit scherm niet genoeg informatie bevat om een probleem op te lossen, kan er nog altijd op de Login knop gedrukt worden. Dit zal ervoor zorgen dat er ingelogd wordt op de betreffende installatie en het visualisatiescherm van deze installatie zal tonen.



Figuur 32: Monitoring installatie vb.net applicatie

5.6 Automatisch rapport

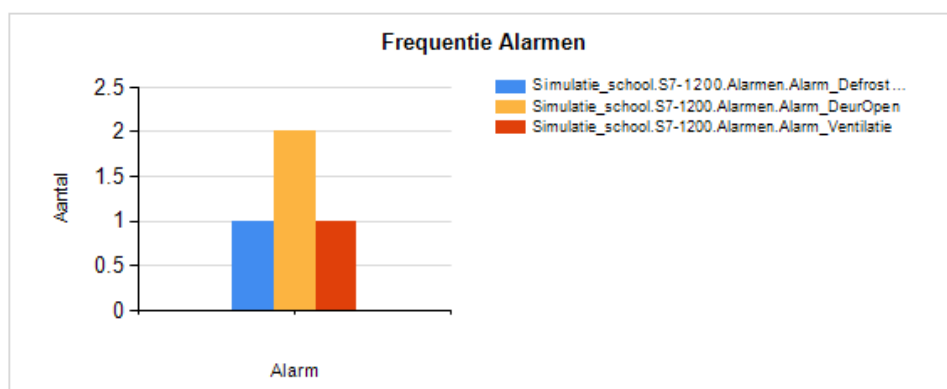
Bijkomend op de vb.net applicatie is er ook met Microsoft Reporting Services een automatisch rapporten ontwikkeld. Dit is ontwikkeld om zoveel mogelijk mogelijkheden te tonen wat er allemaal mogelijk is met de data in de database.

Op Figuur 33 is een voorbeeld te zien van één van de van automatische rapporten. Dit is een rapport gemaakt op basis van waardes die komen van de simulatie. In dit rapport is er een overzicht van alle opgekomen alarmen met hun frequentie te zien. Ook geeft dit een lijst van wanneer elk alarm plaatsvond.

Aan de hand van de waardes die dit rapport toont kunnen een aantal conclusies en eventuele maatregelen genomen worden. Op dit rapport is bijvoorbeeld te zien dat het alarm dat de deur van de koelinstallatie te lang openstaat het meest voorkomt. Vervolgens kan voor deze installatie geopteerd worden om een mechanisme dat de deur automatisch sluit te implementeren, zodat het alarm niet meer voorvalt.

Overzicht alarmen

01/12/2017 00:00:00 - 02/12/2017 00:00:00



ID NAME	VALUE	TIMESTAMP
Simulatie_school.S7-1200.Alarmen.Alarm_DeurOpen	True	01/12/2017 12:19:12
Simulatie_school.S7-1200.Alarmen.Alarm_DeurOpen	True	01/12/2017 12:19:15
Simulatie_school.S7-1200.Alarmen.Alarm_Ventilatie	True	01/12/2017 12:19:24
Simulatie_school.S7-1200.Alarmen.Alarm_DefrostTijd	True	01/12/2017 12:19:29

Figuur 33: Voorbeeld automatisch rapport

Dit voorbeeld toont maar een fractie van wat er mogelijk is met automatische rapporten. Er kan bijvoorbeeld ook niet enkel van 1 installatie, maar van meerdere installaties een overzicht van alarmen gemaakt worden. Ofwel een rapport dat iets volledig toont zoals de gemiddelde ontdooitijd of het aantal ontdooicyclusen. De mogelijkheden zijn eindeloos.

6 Testen werkelijke installaties

Met de controllers van Carel en Green & Cool zijn er testen uitgevoerd op de connectie met de OPC UA server. Dit hoofdstuk zal bespreken hoe er best te werk gegaan wordt om een connectie op te zetten tussen de controller en de OPC UA server voor de geteste controllers. Voor de niet geteste controllers is dit ook beschreven, maar deze manier kan niet met zekerheid bevestigd worden omdat deze niet getest zijn. De manier van aanpak verschilt voor elke controller.

6.1 Carel

Om de Carel controllers te connecteren met de OPC UA server dienen de controllers ingesteld te worden als Modbus slave. Het instellen als Modbus slave is gelijkaardig voor de PlantWatchPro en de PlantVisorPro. Het enige verschil is het aantal variabelen dat ze kunnen beschikbaar stellen zoals beschreven in 3.1.1. De PlantWatchPro kan maximaal 500 read/write variabelen ter beschikking stellen [16]. De PlantVisorPro kan afhankelijk van het model tot maximaal 3000 Read/Write variabelen beschikbaar stellen. De PlantVisorPro 2 Standard kan er 1000 en de PlantVisorPro 2 Hyper kan er 3000 [18].

Om de Modbus slave functionaliteiten mogelijk te maken, moeten de controllers beschikken over de “Modbus Slave plug-in”. Dit is een plug-in dat hoort bij het “Safety Package”. Dit is een pakket dat apart aangekocht dient te worden en heeft voor de PlantWatchPro het nummer: PW3SSFP000 [16]. Voor de PlantVisorPro heeft dit pakket het nummer: PP2STSFP00 [18].

De praktische testen op de Carel controllers zijn enkel uitgevoerd op een PlantWatchPro omdat enkel hiervoor het “Safety Package” aanwezig was. Aangezien de configuratie van beide controllers volledig gelijkaardig is, zal in deze paragraaf enkel de configuratie voor de PlantWatchPro uitgelegd worden.

De configuratie van de OPC UA server is ook volledig gelijkaardig en zal ook enkel uitgelegd worden aan de hand van de testen op de PlantWatchPro.

6.1.1 Configuratie controllers

Allereerst dient de plug-in actief gemaakt te worden. Hiervoor moet er door het menu van de PlantWatchPro genavigeerd worden op de volgende manier: Configuration → Plugins → Modbus Slave → Plugin Configuration. Als dit gebeurd is, is het scherm afgebeeld op Figuur 34 te zien. Om de Modbus slave plug-in te activeren moet het vakje “status”, actief staan. Vervolgens moet de juiste poort ingevuld worden en indien er de mogelijkheid moet zijn om tags te kunnen schrijven, dient het vakje “write enabled” aangevinkt te zijn [16, p. 2].



Figuur 34: PlantWatchPro activeren plug-in [16, p. 2]

Als de Modbus slave plug-in geactiveerd is, moet er een virtueel apparaat aangemaakt worden. Want de Carel controllers kunnen enkel variabelen communiceren over Modbus via virtuele apparaten. Om een virtueel apparaat aan te maken moet er op volgende manier door het menu genavigeerd worden van de PlantWatchPro: Configuration → Plugins → Modbus Slave → Create/remove device. Waarna een scherm zoals op Figuur 35 te zien is. Om een virtueel apparaat aan te maken dient er op de “+” gedruwd te worden, aangeduid met een rode kader op Figuur 35. Vervolgens is er een scherm te zien waar een gekozen naam en Modbus adres aan het apparaat toegekend kan worden. Als dit gebeurd is staat het aangemaakte apparaat tussen de lijst van virtuele apparaten [16, p. 3].



Figuur 35: PlantWatchPro virtueel apparaat [16, p. 3]

Om variabelen toe te kennen aan een virtueel apparaat, dient er eerst genavigeerd te worden doorheen het menu op de volgende manier: Configuration → Plugins → Modbus Slave → manage device. Waarna een lijst van alle virtuele apparaten te zien is. Om dan variabelen aan één van deze apparaten toe te voegen, dient het geselecteerd te worden. Vervolgens is er een scherm zoals op Figuur 36 te zien. Om variabelen toe te voegen dient op de “+” gedruwd te worden, waarna er een lijst van variabelen te zien is. De gewenste variabelen dienen dan één voor één geselecteerd te worden en zijn dan te zien in een lijst met een Modbus adres zoals op Figuur 36 [16, p. 5].

Type	Addr	Device	Variable	R/W
coil	1	Device-1	Defrost	RO
coil	2	Device-1	Defrost command	RW
reg32	1	Device-1	Reg. setpoint	RW
reg32	3	Device-1	Defrost temp.	RO

Figuur 36: PlantWatchPro Modbus variabelenlijst [16, p. 5]

6.1.2 Configuratie OPC UA server

De configuratie van de OPC UA software is gelijkaardig aan de configuratie beschreven in paragraaf 5.3. Hierin staat zowel de configuratie van de OPC UA server als de configuratie van de datalogger in beschreven. De configuratie van de datalogger is volledig hetzelfde zoals eerder beschreven. Echter bij de configuratie van de OPC UA server zijn er enkele verschillen. De grootste verschillen zitten in de adressering van de variabelen aangezien het hier om Modbus TCP/IP en niet Siemens TCP/IP gaat. Alle verschillen met de configuratie beschreven in paragraaf 5.3 zullen in deze paragraaf uitgelegd worden

Het eerste verschil is bij het aanmaken van de connectie. Hier dient het Modbus TCP/IP protocol gekozen te worden als communicatieprotocol. Een tweede verschil is bij het aanmaken van het apparaat in Kepservers. Hier moet in het vakje ID het IP-adres van de controller, gevolgd door een punt en het Modbus adres van het virtuele apparaat ingevuld worden. Dus als bijvoorbeeld het IP-adres van de controller 192.168.1.80 is en het Modbus adres van het virtuele apparaat 3 is, dan dient er bij ID 192.168.1.80.3 ingevuld te worden.

Ten slotte zijn er de verschillen bij de adressering van de “User Defined Tags”. De Modbus adressen van de variabelen in de controller dienen omgevormd te worden naar Modicom adressen. Dit zijn is de standaardadressering voor Modbus variabelen. Hoe de variabelen omgevormd dienen te worden binnen Kepsverex, is afhankelijk van het type variabele (te zien in Tabel 4) en de manier van adressering binnen Kepsverex (te zien in Tabel 5).

Op Tabel 4 is te zien dat er bij de Carel controllers twee verschillende “Modbus read functions” zijn, namelijk “Read coils” en “Read holding Registers”. Dit zijn standaard functies om variabelen via Modbus te lezen.

Om de variabelen van de virtuele apparaten van de controllers te kunnen lezen, moet er gebruik gemaakt worden van Tabel 4 en Tabel 5. De “Modbus read functions” van de Carel controllers uit Tabel 4 moeten overeen komen met de “Function codes” van Kepsverex uit Tabel 5 om de variabelen te kunnen lezen. Vervolgens moet er naar de bijhorende Range van de “Function codes” uit Tabel 5 gekeken worden om het Modicom adres te kennen.

Dit principe om het juiste Modicom adres te gebruiken wordt uitgelegd aan de hand volgend voorbeeld. Stel het type van de variabele dat van de controller dient gelezen te worden is een “digital variabele” en heeft het Modbus adres 1. Dan moet er eerst naar Tabel 4 gekeken worden, wat de “Modbus read function” van een digital is. In dit geval is dat 01. Dit nummer moet er vervolgens gezocht worden tussen de “Function codes” van Tabel 5. Als de functiecode gevonden is, dan geeft de bijhorende range het bereik van de Modicom adressen weer. In dit geval is de range 000001-065536. Merk ook op dat de adressering hier begint bij 1 en niet bij 0. Dit zorgt ervoor dat het Modbus adres 1 voor een digitale variabele, het Modicom adres 000002 heeft. Een ander voorbeeld is een analoge temperatuur met Modbus adres 9. Het Modicom adres voor dit voorbeeld is dan 400010.

Merk ook op dat in Tabel 5 de Modicom adressen in Kepsverex altijd 6 cijfers lang zijn. De standaard adressering van Modbus gebruikt echter een 5-cijferige adressering, maar Kepsverex gebruikt één cijfer meer, zodat er per variabeltype meer variabelen mogelijk zijn. Bij Modicom variabelen is het zo dat het eerst karakter aangeeft om welk type variabele het gaat en de rest geeft het adres [39].

Tabel 4: Omvorming Modbus adressen Carel [18, p. 9]

Physical variable type	Modbus variable type	Length variable	Modbus read functions	Modbus write functions	Acquisition coding
Digital	Coil	1 bit	01 Read Coils	05 Write Single Coil 15 Write Multiple Coils	Binary
Alarm	Coil	1 bit	01 Read Coils	-	Binary
Analogue	Register	2 word	03 Read Holding Registers	06 Write Single Register 16 Write Multiple Registers	Standard Floating-point (32 bit)
Integer	Register	1 word*	03 Read Holding Registers	06 Write Single Register 16 Write Multiple Registers	Signed integer

Tabel 5: Kepservex Modbus adressering [39, p. 46]

Address Type	Range	Data Type	Access*	Function Codes
Output Coils	000001-065536	Boolean	Read/Write	01, 05, 15
Input Coils	100001-165536	Boolean	Read Only	02
Internal Registers	300001-365536	Word , Short, BCD Float, DWord, Long, LBCD Double	Read Only	04
	300001-365535		Read Only	04
	300001-365533		Read Only	04
	xxxxx=1-65536 bb=0/1-15/16**	Boolean	Read Only	04
	300001.2H- 365536.240H***	String	Read Only	04
300001.2L- 365536.240L***	String	Read Only	04	
Holding Registers	400001-465536	Word , Short, BCD Float, DWord, Long, LBCD Double	Read/Write	03, 06, 16
	400001-465535		Read/Write	03, 06, 16
	400001-465533		Read/Write	03, 06, 16
	xxxxx=1-65536 bb=0/1-15/16*	Boolean	Read/Write	03, 06, 16, 22
	400001.2H- 465536.240H***	String	Read/Write	03, 16
400001.2L- 465536.240L***	String	Read/Write	03, 16	

6.2 Green & Cool

De connectie tussen de Green & Cool controllers en de OPC UA server werd ook getest. Om deze connectie tot stand te brengen is er geen extra configuratie nodig op de controllers. Er dient enkel in de Schneider PLC een parameterlijst te bestaan met variabelen, dewelke toegankelijk is via Modbus. Zelf kon er geen parameterlijst uit de PLC gehaald worden met de programmatiesoftware “SoMachine” omdat de PLC volledig afgeschermd was.

Om toch een parameterlijst te bemachtigen is er contact opgenomen met Green & Cool. Zij hebben een volledige lijst bezorgd met alle Read Only variabelen, toegankelijk via Modbus. Een deel van deze lijst is te zien in Tabel 6. De adressen en namen van de variabelen, te zien in deze tabel komen niet overeen met de werkelijke adressen en namen van de variabelen voor veiligheids- en geheimhoudingsredenen. Deze tabel is indicatief en voldoende om de configuratie van de adressering binnen Kepsserverex uit te leggen.

Tabel 6: Parameterlijst Green & Cool [40]

GREEN & COOL Green Refrigeration Systems		Modbus - M241 Parameter list									
Variable	Alarm	Object description	Signal description	Modicon Address	1-based Address	%Mx0 Address	Eng. Unit	Datatype	Scale	Read/Write	Register Type
LIQ_L_ALM	B	Liquid Out Temp	Low Alarm	101	101	100		BOOL		Read Only	Coil
CO2_H_ALM	B	CO2 Out Temp	High Alarm	102	102	101		BOOL		Read Only	Coil
CO2_L_ALM	B	CO2 Out Temp	Low Alarm	103	103	102		BOOL		Read Only	Coil
T1		Discharge Temp	Process value	40201	201	200	°C	INT	x0.1	Read Only	Holding Register
T2		Temp After Gas Cooler	Process value	40202	202	201	°C	INT	x0.1	Read Only	Holding Register
T3		Liquid Temp	Process value	40203	203	202	°C	INT	x0.1	Read Only	Holding Register

6.2.1 Configuratie OPC UA server

De configuratie van de OPC UA server voor de controllers van Green & Cool is gelijkaardig aan de configuratie van de Carel controllers (beschreven in 6.1.2) omdat het communicatieprotocol hier ook Modbus TCP/IP is. Hier zijn ook enkele verschillen met de configuratie van de OPC UA server zoals beschreven in 5.3, maar de configuratie van de datalogger is ook hier hetzelfde.

Net zoals bij de Carel controllers dient hier bij het aanmaken van de connectie het Modbus TCP/IP protocol gekozen te worden. Bij het aanmaken van het apparaat, dient het IP-adres van de Schneider PLC ingevuld te worden in het vakje van ID.

Ten slotte is de adressering van de tags bij deze controller eenvoudiger omdat in de gekregen tabel de omgevormde Modicom adressen er al mee in vermeldt staan. Dus de omvorming dient hier niet meer te gebeuren. Enkel bij de adressen van de Coils dienen nog enkele nullen bijgeplaatst te worden, zodat het adres beschikt over de zes karakters dat Kepsserverex nodig heeft. Zo zal het te gebruiken adres in Kepsserverex voor de variabele “LIQ_L_QLM” met het Modicom adres “101” in Tabel 6, “000101” worden.

6.3 Danfoss

Met de Danfoss controllers zijn er geen testen uitgevoerd op één van de werkelijke installaties. Echter wordt er wel advies gegeven worden over hoe een connectie met deze controller het beste wordt aangepakt.

Na contact op te nemen met Danfoss is het duidelijk geworden het communiceren via Modbus wordt afgeraden voor Danfoss controllers. Er wordt aangeraden om de “eWON” routers te gebruiken en te communiceren via het HTTP protocol. De aanvragen en antwoorden van de waardes van de variabelen zijn gecodeerd in XML [22].

“eWON” heeft een volledig uitgewerkt voorbeeld, om variabelen te kunnen communiceren met Danfoss controllers. Dit voorbeeld is te downloaden via volgende link: <https://developer.ewon.biz/content/danfoss-io-server>. De “eWON” routers hebben zoals eerder beschreven ook een mogelijkheid om ze te gebruiken als een OPC UA server. Hierdoor de totaaloplossing hetzelfde blijven enkel dat voor de Danfoss controllers er een “eWON” router tussen komt.

De werking van het voorbeeld van “eWON” om te communiceren met de Danfoss controllers is niet mee opgenomen in de scope van deze masterproef.

6.4 Wurm

Met de Wurm controllers zijn ook geen praktische testen uitgevoerd. Er zal echter voor deze controller ook advies gegeven worden over hoe de connectie met deze controller het beste wordt aangepakt.

Zoals eerder besproken in paragraaf 3.1.4 is een C2C module noodzakelijk om de gebruikte CAN-bus van de controllers om te zetten naar Modbus RTU. Deze module kan als modbus slave gebruikt worden en zorgt dus voor de mogelijkheid om enkele variabelen via modbus RTU uit te lezen.

Modbus RTU is een serieel protocol waarbij een aantal extra instellingen zoals: Baud rate, parity bits en stop bits zeer belangrijk zijn. Deze moeten juist ingesteld worden vooraleer er gecommuniceerd kan worden. Bij de oudere C2C module gebeuren deze instellingen via DIP-switches [24] en bij het nieuwere model gebeuren deze instellingen via de software “FRIGODATA XP” [41].

De Modbus adressering van de gekoppelde apparaten is hetzelfde als dat van CAN-bus. Wat zorgt dat de CAN-bus adressen 1-120 ook de adressen 1-120 zijn via Modbus [24].

Om actuele waardes uit de controller te kunnen lezen, dienen de adressen van de variabelen ook gekend te zijn. Deze adressen dienen ook hier een conversie te ondergaan, maar de conversie van de adressen is bij de C2C modules apparaat afhankelijk. Dus elk gekoppeld apparaat op het CAN-bus heeft een andere manier voor de conversie van de adressen [24]. De Tabellen om de juiste conversie te maken zijn terug te vinden in de pdf van de C2C module (bron 24 en 39).

De C2C module ondersteund enkel de “Modbus Funtion code” 03 [24]. Wat betekend dat als er naar Tabel 5 gekeken wordt, dit “Holding registers” zijn. Dus moeten de adressen van de variabelen in Kepsserverex de range van de “Holding registers” zoals weergegeven in Tabel 5 respecteren.

Er wordt hier echter niet dieper op ingegaan omdat enkel na het uitvoeren van praktische testen de manier van werken correct beschreven kan worden. Aan de hand van bovenstaande informatie en met de pdf's van de C2C modules (bron 24 en 39) zou men in staat moeten zijn om een connectie tussen de wurm controllers en de OPC UA server te maken.

6.5 Dixell

Ten slotte is er zijn er de Dixell controllers. Op deze controllers zijn ook geen testen op uitgevoerd kunnen worden, maar hiervoor wordt ook advies gegeven hoe dit in de toekomst het best gebeurt.

Bij de Dixell controllers kan er zoals eerder besproken enkel gecommuniceerd worden met de XWEB EVO. Met de XWEB, het oudere model, kan er standaard niet via Modbus RTU gecommuniceerd worden. Echter zou voor de XWEB zou wel gebruik gemaakt kunnen worden van het PLC-programma dat Dixell, zoals eerder besproken, gemaakt heeft om een extra PLC op het Modbus netwerk te connecteren. Op deze manier zouden er vervolgens wel enkele variabelen gecommuniceerd kunnen worden via de PLC, maar hiervoor moet er een extra PLC bijgeplaatst worden op de installatie.

Om variabelen van de XWEB EVO te kunnen uitlezen dient er gebruik gemaakt te worden van het HTTP protocol. Hiermee is het mogelijk om variabelen uit te controller te lezen via API's zoals te zien op onderstaand voorbeeld (Figuur 37).

```
https://192.168.0.1/api?action=devices_list&username=Admin&password=Admin
```

```
{
  "status": "ok",
  "message": "Devices list successfully fetched",
  "devices": [
    {
      "id": 3,
      "addr": "RS1-003",
      "name": "New_XR75CX",
      "model": "002C-0063-0058-XR75CX",
      "enabled": true,
      "protocol": "dixelldev",
      "serial": "1,9600,n,8,1,30,5,2"
    },
    {
      "id": 4,
      "addr": "RS1-004",
      "name": "New_XR75CX",
      "model": "002C-0063-0058-XR75CX",
      "enabled": true,
      "protocol": "dixelldev",
      "serial": "1,9600,n,8,1,30,5,2"
    }
  ]
}
```

Figuur 37: Voorbeeld API Dixell [42]

De werking en het gebruik van API's, is niet opgenomen in de scope van deze masterproef en staat dus niet beschreven in deze scriptie

7 Besluit

Met de huidige technologieën zijn IIoT toepassingen voornamelijk interessant voor monitoring en analyse van systemen en processen. Want IIoT toepassingen zijn gebonden aan het internet en proces en kwaliteit kritische activiteiten en veiligheidssystemen mogen nooit afhankelijk zijn van een internet connectie. Daarom raad ik aan om IIoT toepassing enkel te gebruiken voor de monitoring van installaties.

IIoT monitoring toepassingen kunnen echter wel een aanzienlijke meerwaarde bieden aan een bedrijf. Zo is het mogelijk om informatie van meerdere sites te bundelen in één grote database te analyseren. Dit maakt een aantal functies mogelijk. Een aantal voorbeelden zijn:

- Predictive maintenance,
- Het uitbesteden van grondstof voorraadbeheer;
- Mogelijkheid dat R&D gebruiksgegevens ontvangt;
- Het bundelen van procesgegevens van meerdere sites.

Algemeen kan er besloten worden dat IIoT nog zeer fel aan het groeien is en nog lang niet op zijn maximale capaciteit zit. De toekomst zal moeten uitwijzen wat IIoT nog allemaal mogelijk zal maken. Echter één ding is zeker IIoT zal de industrie in positieve zin veranderen.

Voor de praktische case van Cooltech is er gekozen om de OPC UA server software van Kepware te gebruiken voor het connecteren van alle controllers. De controllers kunnen deze server bereiken via het Modbus TCP/IP of RTU protocol. Bijkomend dient er een VPN-connectie opgezet te worden tussen het hoofdkantoor van Cooltech en elke koelinstallatie. Zo is er maar één enkele OPC UA server noodzakelijk op het hoofdkantoor van Cooltech, wat de licentiekost doet dalen.

De VB.NET applicatie en de rapporten van de Microsoft Reporting services geven een goed beeld van wat mogelijk is met de data, eens het in een database staat. De applicatie geeft in eerste instantie een overzicht over de status en alarmen van meerdere koelinstallaties. Alsook een detailoverzicht en monitoring van elke individuele koelinstallatie.

De simulatie van de koelinstallatie heeft aangetoond dat de gekozen totaaloplossing werkt indien er een connectie tussen de OPC UA server en de controller gemaakt kan worden. Dit is ook getest en bewezen voor de controllers van Carel en Green & Cool. Verdere testen moeten deze werking ook nog aantonen voor de andere controllers.

De langdurige en grootschalige werking van de voorgestelde totaaloplossing dient nog bevestigd te worden door de implementatie en veelvoudige testen op de werkelijke koelinstallaties.

Literatuurlijst

- [1] Cooltech, [Online]. Available: <http://www.cooltech.be/>. [Geopend 11 September 2017].
- [2] Reditech Engineering, [Online]. Available: <https://www.reditech.be/nl>. [Geopend 11 September 2017].
- [3] "Industry 4.0: the fourth industrial revolution," [Online]. Available: <https://www.i-scoop.eu/industry-4-0/>.
- [4] P. McLaughlin and R. McAdam, "The Undiscovered Country: The Future of Industrial Automation," *White paper Honeywell*, p. 14, 2016.
- [5] Manufacturers Automation, "Moxa – Make Your OT, IT, IIoT Protocol Interoperability Easier," 2017.
- [6] IEB Media GbR, "IIoT: Combining the Best of OT and IT," *Industrial Ethernet Book*, 2017.
- [7] Reditech engineering, „MES,” in *GastCollege KUL*, 2017.
- [8] GE, "What is Edge Computing?," 2017.
- [9] S. Aron, "IIOT Protocols to Watch," *Whitepaper Kepware*, no. 2, p. 6, 2016.
- [10] B. Farnham and R. Barillère, "MIGRATION FROM OPC-DA TO OPC-UA," Geneva, Switzerland, 2011.
- [11] X. Chen and R. Jain, "Constrained Application Protocol for Internet of Things," wustl.edu, 2014.
- [12] T. Salman and R. Jain, "Networking Protocols and Standards for Internet of Things," 2015.
- [13] A. Tozon, "Internet of things track (IoT) - Smart Home," in *NT Konferenca*, Slovenië, 2015.

- [14] Trunomi, "EUGDPR," [Online]. Available: <https://www.eugdpr.org/>. [Accessed 14 September 2017].
- [15] Carel, PlantWatchPRO - User manual, Padova (Italy): CAREL INDUSTRIES Hqs, 2014.
- [16] Carel, Modbus Slave Plugin_rev1.0, Padova (Italy): CAREL INDUSTRIES HQs.
- [17] Carel, "Carel - PlantVisorPro," [Online]. Available: http://www.carel.com/local-monitoring-systems/-/journal_content/56_INSTANCE_i4q5KIMLInKK/10191/63184. [Accessed 4 October 2017].
- [18] Carel, Modbus Slave_rev1.1, Brugine, 2015.
- [19] Controls, drives & automation, "AUTOMATION OFFERING EXPANDED," 2014.
- [20] Schneider Electric, Industrial Ethernet, 2017.
- [21] Danfoss, AK-SM 800 Series - User Guide v2.x, 2016.
- [22] Danfoss , XML Interface 1.0 (Deprecated), Danfoss inc., 2015.
- [23] Wurm, HKS-G3 - Productinformatie, 2017.
- [24] Wurm, C2C-MOD-S-MAN: Coupling module as Modbus slave, 2017.
- [25] Frigro, [Online]. Available: <http://www.frigro.be/>. [Geopend 9 Oktober 2017].
- [26] Dixell, XWEB EVO: OPERATING MANUAL (V.4.3), Italy: Emerson, 2017.
- [27] LANCOM Systems GmbH, LANCOM 1781-4G: Datasheet, Germany, 2017.
- [28] Cooltech, *Handleiding - Monitoring Online*, 2017.

- [29] IO-Link Company Community, IO-Link System Description, Germany, 2013.
- [30] Comtrol, "Comtrol Introduces Microsoft Azure IoT Gateway SDK Integration for its IO-Link Master Family," 24 April 2017. [Online]. Available: <http://www.comtrol.com/news/press/comtrol-introduces-microsoft-azure-iot-gateway-sdk-integration-io-link-master-family>. [Accessed 28 September 2017].
- [31] Comtrol, "Microsoft Azure IoT Gateway integration on IO-Link Master's," 2017. [Online]. Available: <http://www.comtrol.com/io-link-master-gateways/io-link-competency-center/microsoft-azure-iot-gateway-integration-io-link-masters>. [Accessed 28 September 2017].
- [32] eWon, "eWON Flexy," [Online]. Available: <https://ewon.biz/products/ewon-flexy>. [Accessed 24 October 2017].
- [33] Kepware, "Kepserverex," [Online]. Available: <https://www.kepware.com/en-us/products/kepserverex/>. [Accessed 31 October 2017].
- [34] Microsoft, "Microsoft Open Database Connectivity (ODBC)," 19 Januari 2017. [Online]. Available: <https://docs.microsoft.com/en-us/sql/connect/odbc/windows/microsoft-odbc-driver-for-sql-server-on-windows>. [Accessed 29 12 2017].
- [35] Microsoft, „Prijsen voor SQL Database,” [Online]. Available: <https://azure.microsoft.com/nl-nl/pricing/details/sql-database/>. [Geopend 26 December 2017].
- [36] Microsoft, „Database Transaction Units (dtu's) en elastische Database Transaction Units (edtu's)," 14 April 2017. [Online]. Available: <https://docs.microsoft.com/nl-nl/azure/sql-database/sql-database-what-is-a-dtu>. [Geopend 26 December 2017].
- [37] Carel, "ir33+ platform," in *User manual*, Padova (Italy), CAREL INDUSTRIES, 2013, p. 29.
- [38] Kepware, "Kepserverex V6," in *Manual*, PTC Inc., 2017, pp. 134-146.
- [39] Kepware, Modbus Ethernet Driver, ptc inc., 2017.
- [40] Green & Cool, *Modbus - Parameter list M241.Rev35*, 2017.

[41] Wurm, C2C-MOD-SGT: Coupling module as Modbus slave, 2017.

[42] Dixell, XWEB EVO API.

Auteursrechtelijke overeenkomst

Ik/wij verlenen het wereldwijde auteursrecht voor de ingediende eindverhandeling:
Praktische IIoT case

Richting: **master in de industriële wetenschappen: energie-automatisering**
Jaar: **2018**

in alle mogelijke mediaformaten, - bestaande en in de toekomst te ontwikkelen - , aan de Universiteit Hasselt.

Niet tegenstaand deze toekenning van het auteursrecht aan de Universiteit Hasselt behoud ik als auteur het recht om de eindverhandeling, - in zijn geheel of gedeeltelijk -, vrij te reproduceren, (her)publiceren of distribueren zonder de toelating te moeten verkrijgen van de Universiteit Hasselt.

Ik bevestig dat de eindverhandeling mijn origineel werk is, en dat ik het recht heb om de rechten te verlenen die in deze overeenkomst worden beschreven. Ik verklaar tevens dat de eindverhandeling, naar mijn weten, het auteursrecht van anderen niet overtreedt.

Ik verklaar tevens dat ik voor het materiaal in de eindverhandeling dat beschermd wordt door het auteursrecht, de nodige toelatingen heb verkregen zodat ik deze ook aan de Universiteit Hasselt kan overdragen en dat dit duidelijk in de tekst en inhoud van de eindverhandeling werd genotificeerd.

Universiteit Hasselt zal mij als auteur(s) van de eindverhandeling identificeren en zal geen wijzigingen aanbrengen aan de eindverhandeling, uitgezonderd deze toegelaten door deze overeenkomst.

Voor akkoord,

Thoelen, Glen

Datum: **15/01/2018**