Alexander Svezhenets

BI and Automation Projects

TABLE OF CONTENTS

PREREQUISITES

Actual numbers, metrics, and dates in this document are confidential and have been changed for the purpose of demonstration. Reports and reports design and structure, eventual company names, task descriptions etc. are actual.

DATA WAREHOUSE

Objective

Unified data storage and source of information for all kinds of analytical reports across the holding. One solution for both ferry and bus subsidiaries (dimension types, metrics and reporting requirements coincide almost completely)

Challenges

- Dimensions (ships, vehicles, departments, ferry, and bus lines) have different codification in different OLTP source systems.
- Complex dimension dependencies across the holding (for example 30% of costs that belong to dimension 200, must be also accounted for in dimension 300 etc.)

Original reporting process

- Isolated reporting routines where subsidiaries used their own tools, established dataflow routines and reporting standards.
- Lage volumes of manual work involved.
- Reports are updated monthly, and updates are available not earlier than on the 10th of the next month (the process can take up to 10 days to complete)
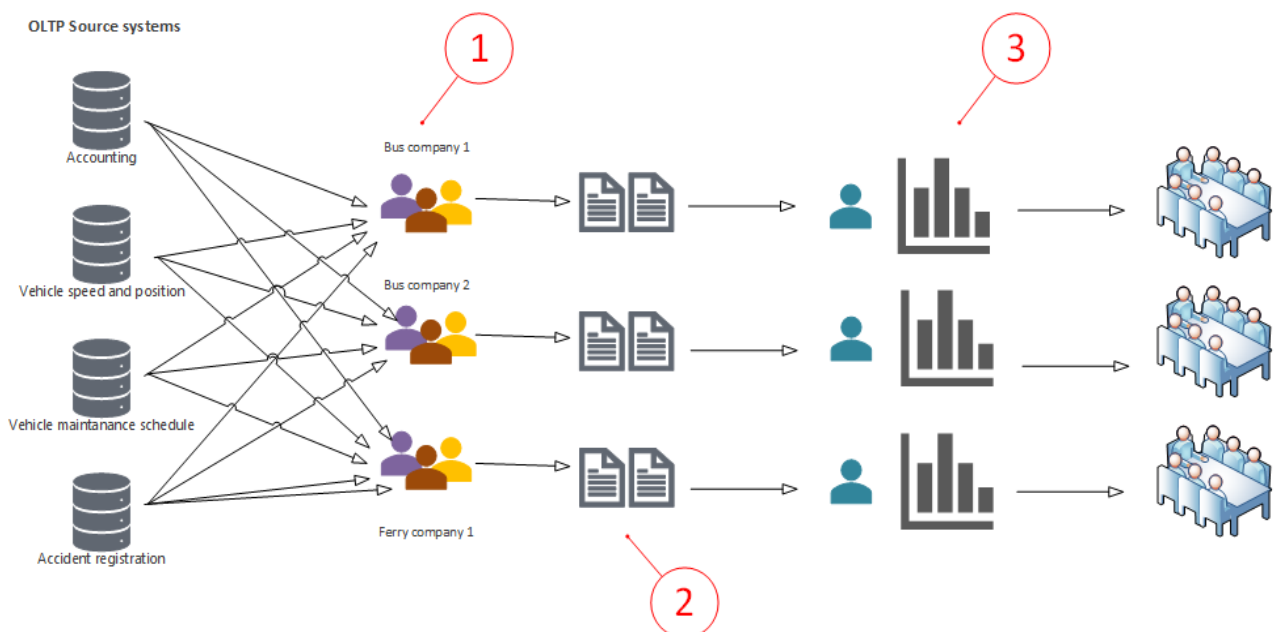
1. Raw data had to be extracted from isolated OLTP source systems (manually, copy / paste)
2. The data was then grouped and transformed manually and saved as Excel / Word files
3. Persons responsible for reporting, used the files to prepare final reports (Excel, Power Point). The final reports were then presented to the Board

Optimized reporting process

- Unified reporting process with common rules and metrics.
- Fully automated dataflows and report updates (zero manual work)
- Reports are updated daily / hourly, the process is fully automated
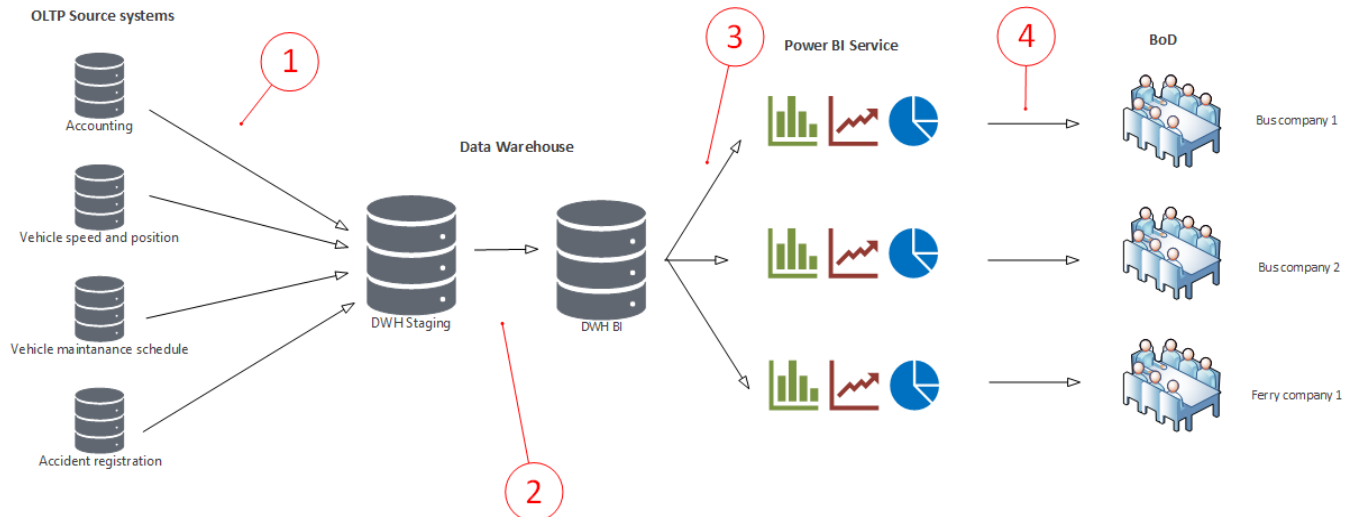
1. Raw data is partially transformed and transferred to Staging DWH (on-premises MS SQL Server), using automated ETL / ELT tools (SSIS packages scheduled as SQL Server Agent jobs)
2. The same ETL tools transform the data further and update both fact and dimensions tables in DWH BI
3. Power BI Service reports are updated automatically through Power BI Gateway (daily, hourly)
4. Members of the Board have instant, direct access to the updated Power BI reports through their desktops, laptops, and mobile devices

DWH: screenshots
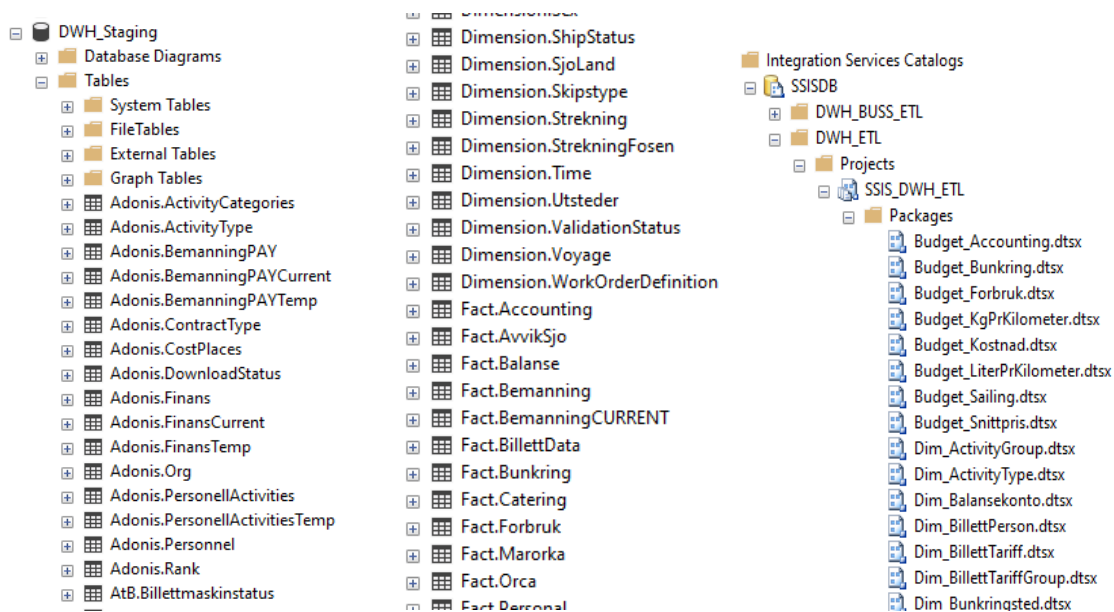
BI: HSE REPORT, FERRY COMPANIES

Objective

HSE (Health, Safety and Environment) report is of crucial importance to a ferry company in Norway. A company that fails to report right or fails to report on time, can get its license revoked. Objective was to unify and automate the reporting process and to make it fast, predictable, and error-free across the holding.

Challenges

- OLTP source systems operate using different technical backgrounds with different update frequencies and data formats.
- Controlling indicators must be calculated using non-linear or non-straightforward calculation algorithms.
- Each subsidiary follows its own reporting routines and sometimes accident classification could vary from company to company.

Original reporting process

- Raw data had to be extracted from isolated OLTP source systems (manually, copy / paste)
- The data was then grouped and transformed manually and saved as Excel / Word files
- Persons responsible for reporting, used the files to prepare the final report (Word file).
- The final report was then sent in separate e-mails to both controlling authorities and ferryboat's crews.
- Once-a-month update.

Optimized reporting process

- Raw data is gathered and transferred from OLTP systems to DWH automatically
- DWH supplies the transformed data and precalculated metrics to Power BI Service reports through automated updates (Power BI Gateway).
- Direct access to Power BI Service HSE report is granted to both crews and authorities
- Regular, automated daily updates.
- Approximately 50 hours of manual work per month have been eliminated

HSE report: screenshots

HMS indikatorer og frekvensverdier

Måned og siste 12 måneder

0,00	10,65
K-verdi periode	K-verdi sis
0,00	0,00
G-verdi periode	G-verdi sis
0,00	0,83
P-verdi periode	P-verdi sis
31,81	18,50
H-verdi periode	H-verdi sis
127,24	84,57
F-verdi periode	F-verdi sis
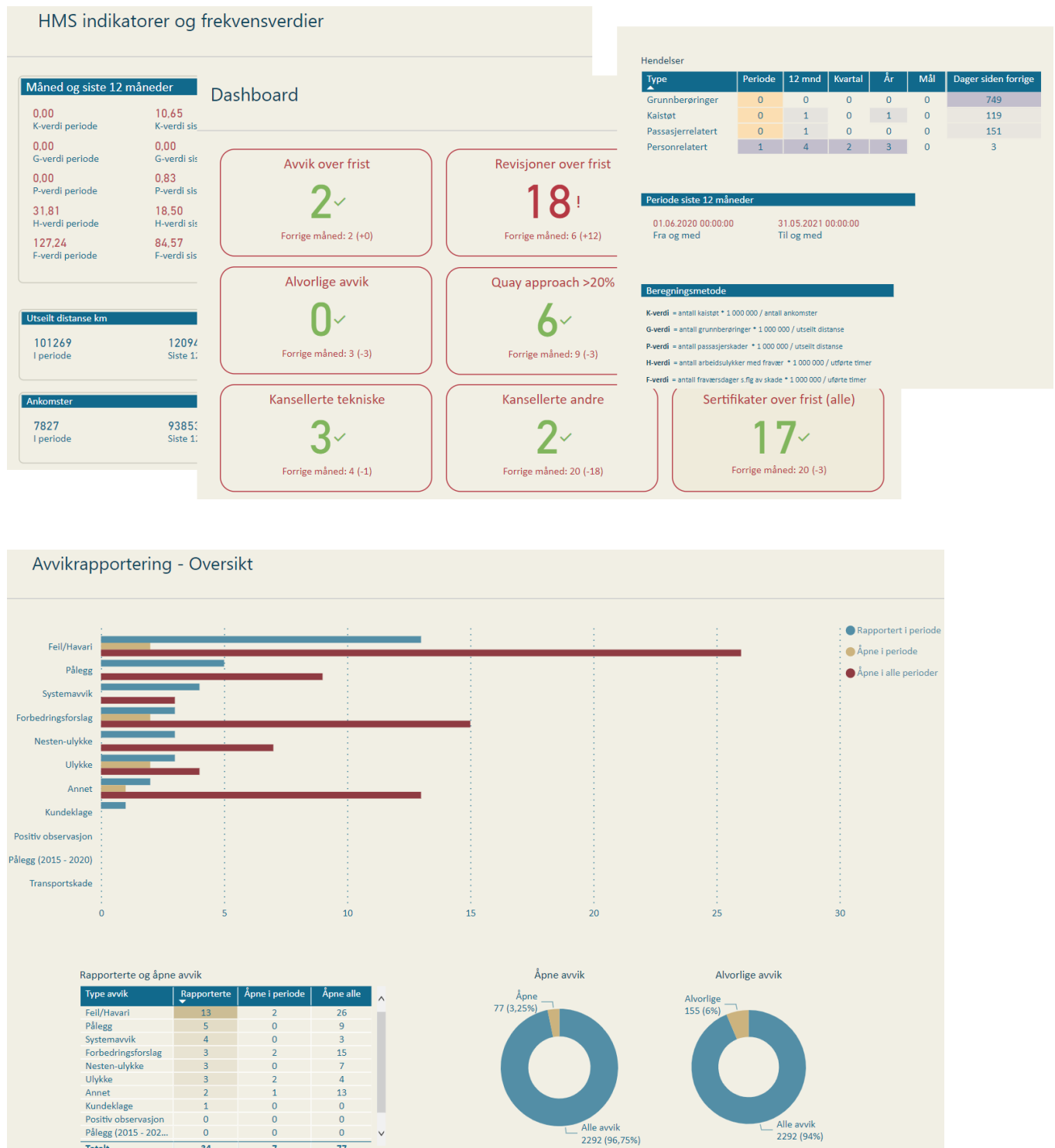
Utseilt distanse km

101269	12094
I periode	Siste 1:

Ankomster

7827	93853
I periode	Siste 1:

Dashboard

Avvik over frist
2 ✓
Forrige måned: 2 (+0)

Revisjoner over frist
18 !
Forrige måned: 6 (+12)

Alvorlige avvik
0 ✓
Forrige måned: 3 (-3)

Quay approach >20%
6 ✓
Forrige måned: 9 (-3)

Kansellerte tekniske
3 ✓
Forrige måned: 4 (-1)

Kansellerte andre
2 ✓
Forrige måned: 20 (-18)

Sertifikater over frist (alle)
17 ✓
Forrige måned: 20 (-3)

Hendelser

Type	Periode	12 mnd	Kvartal	År	Mål	Dager siden forrige
Grunnberøringer	0	0	0	0	0	749
Kaistøt	0	1	0	1	0	119
Passasjerrelatert	0	1	0	0	0	151
Personrelatert	1	4	2	3	0	3

Periode siste 12 måneder

01.06.2020 00:00:00	31.05.2021 00:00:00
Fra og med	Til og med

Beregningsmetode

K-verdi = antall kaistøt * 1 000 000 / antall ankomster

G-verdi = antall grunnberøringer * 1 000 000 / utseilt distanse

P-verdi = antall passasjerskader * 1 000 000 / utseilt distanse

H-verdi = antall arbeidsulykker med fravær * 1 000 000 / utførte timer

F-verdi = antall fraværsdager s.flg av skade * 1 000 000 / utførte timer

Avvikrapportering - Oversikt

Legend: Rapportert i periode / Åpne i periode / Åpne i alle perioder

Rapporterte og åpne avvik

Type avvik	Rapporterte	Åpne i periode	Åpne alle
Feil/Havari	13	2	26
Pålegg	5	0	9
Systemavvik	4	0	3
Forbedringsforslag	3	2	15
Nesten-ulykke	3	0	7
Ulykke	3	2	4
Annet	2	1	13
Kundeklage	1	0	0
Positiv observasjon	0	0	0
Pålegg (2015 - 202...	0	0	0
Totalt	**34**	**7**	**77**

Åpne avvik

Åpne 77 (3,25%)
Alle avvik 2292 (96,75%)

Alvorlige avvik

Alvorlige 155 (6%)
Alle avvik 2292 (94%)

HSE report: screenshots

Premaster vedlikehold

Vedlikehold over frist (periodebasert)*

Fartøy	Beskrivelse	Regelmessighet	Kritikalitet	Forfall	Sist utført	Dager over frist
MS Frøyfart	Årlig kontroll av magnetkompass	Y	Ikke Kritisk	31.03.2021	20.06.2019	61
MS Frøyfart	Sjekk spjeld mororer ved funktions test	M	Ikke Kritisk	14.04.2021	14.03.2021	47
MS Frøyfart	Sjekk ekspansjons tank til AC annleg	W	Ikke Kritisk	15.04.2021	08.04.2021	46
MS Frøyfart	Vernemøte	M	Ikke Kritisk	18.04.2021	18.10.2020	43
MS Frøyfart	Kontroll av avfallshåndtering	M	Ikke Kritisk	22.04.2021	22.03.2021	39

Vedlikehold over frist (timebasert)

Fartøy	Beskrivelse	Regelmessighet	Kritikalitet	Forfall timer	Faktiske timer	Forfall
MS Frøyfart	Skift brennolje fin filter	*	M	Ikke Kritisk	2793	2985	NA

Antall forfalte

27 ✓

Forrige måned: 203 (-176)

Snitt dager over frist

27 ✓

Forrige måned: 224 (-197)
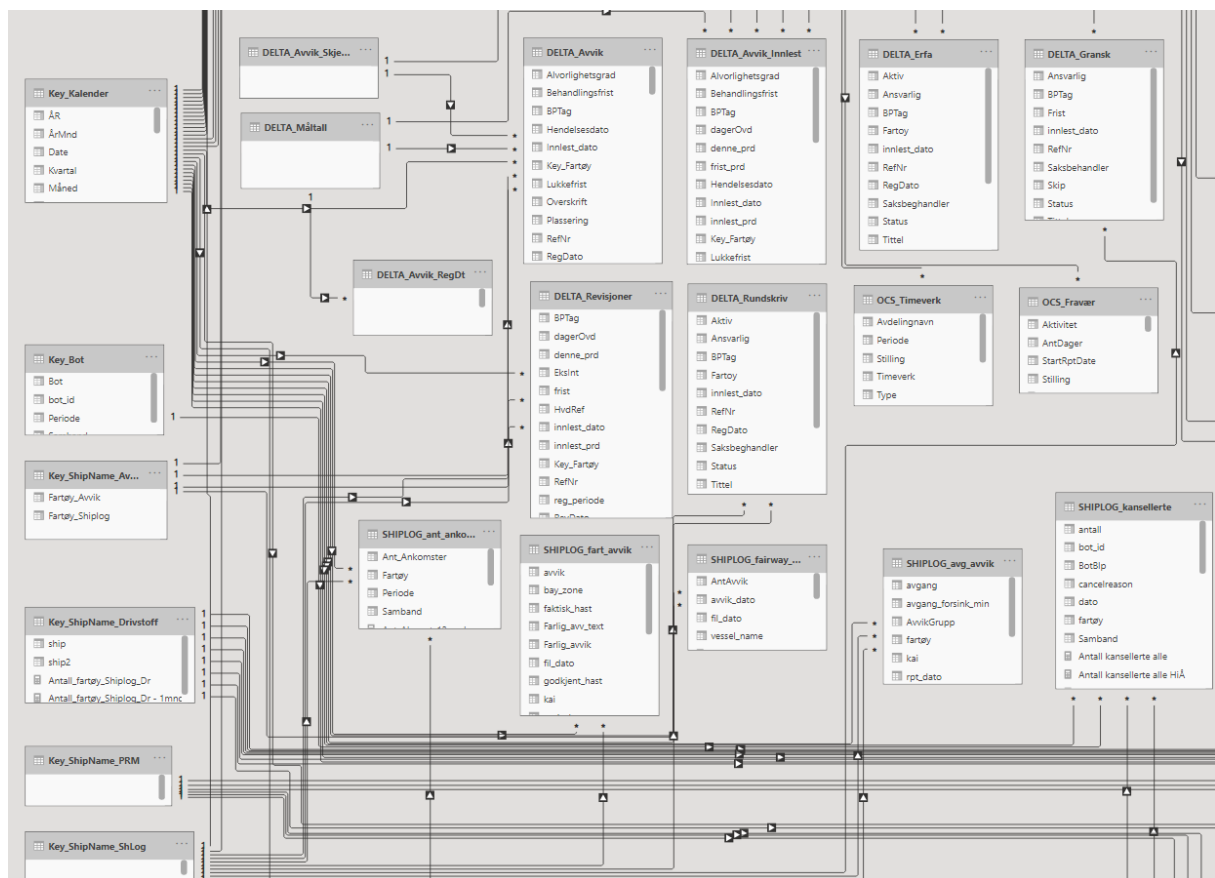
# BI: FUEL CONSUMPTION REPORT, FERRY

Objective

Unified reporting format across the holding. Faster updates and elimination of manual inputs into the accounting system.

Challenges

- Not all the ferries have been equipped with automated fuel counters.
- Crews used different ways to register fueling information (some of them used Speed and Positioning system, while other used Excel-based reports etc.).
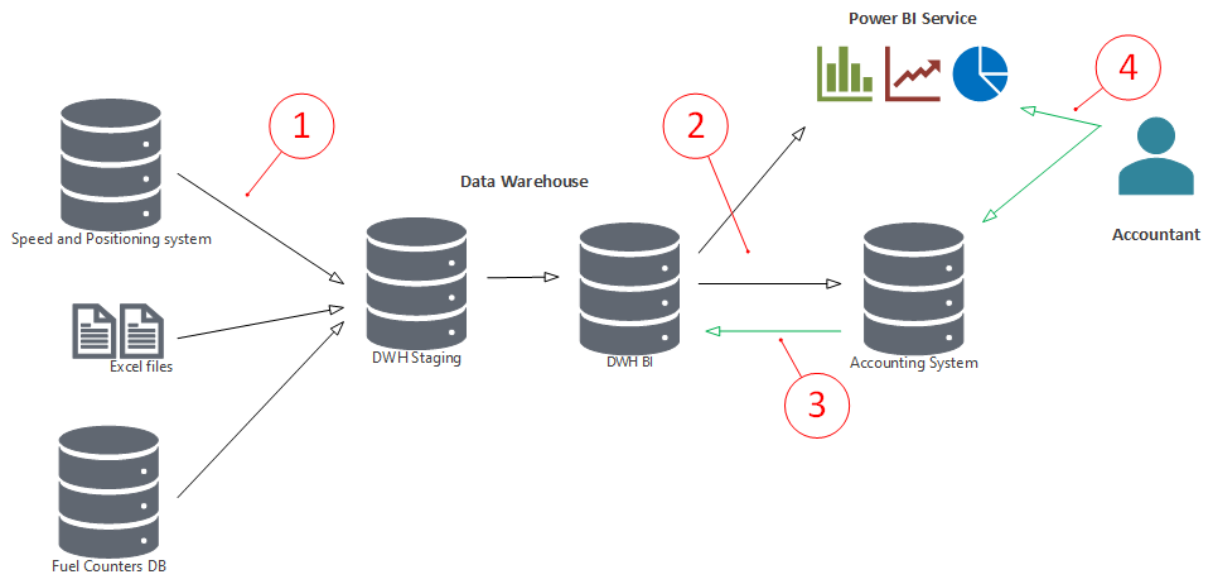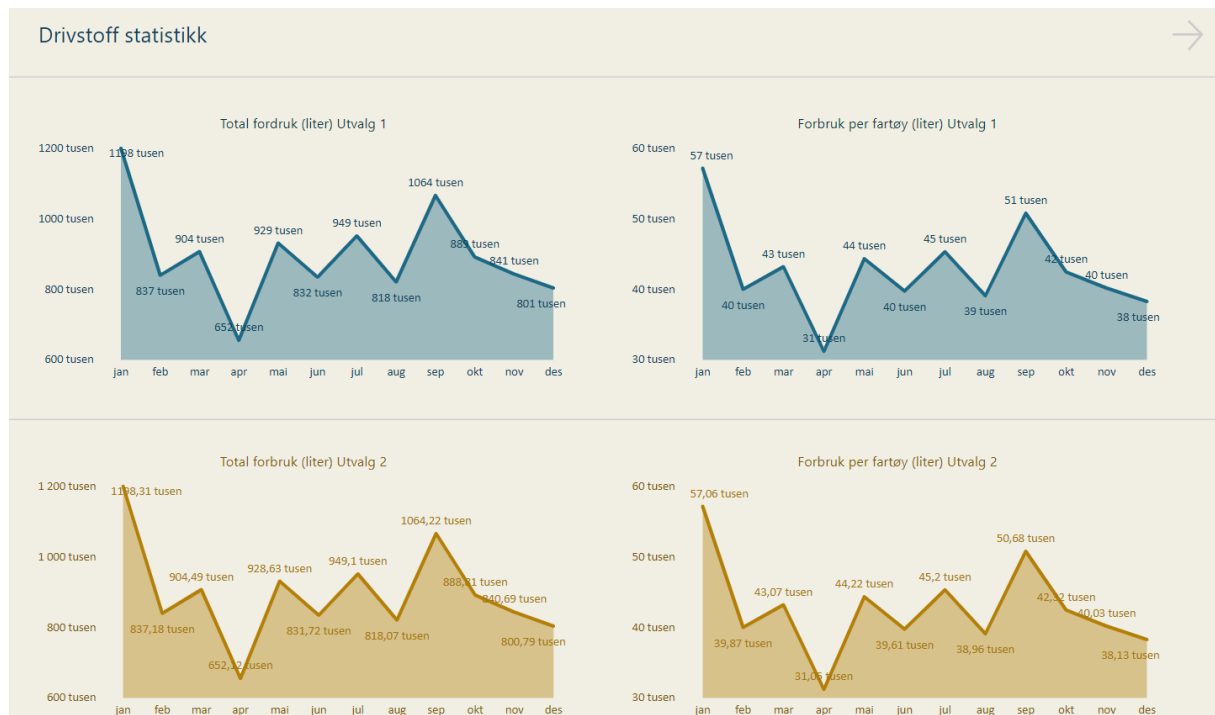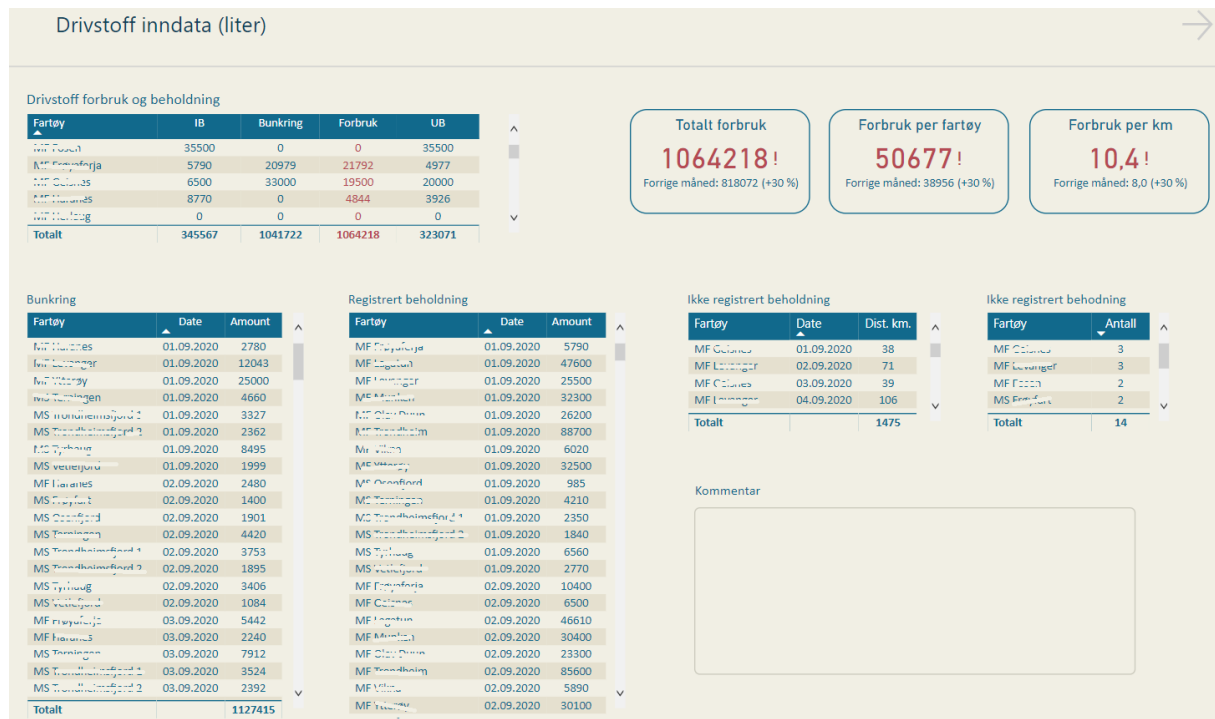
Original reporting process
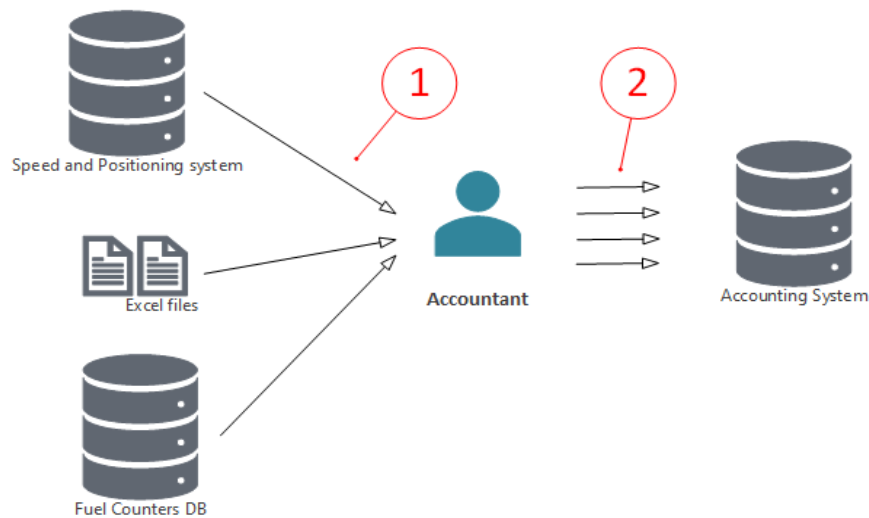
- Fuel repots are copied from a corresponding source system manually
- An accountant registers the reports in the accounting system manually (one report per subsidiary)

Optimized reporting process

1. Fueling data is read into DWH, using API and in-house ETL process (SSIS)
2. An RPA process reads fueling reports from DWH and registers these reports in the accounting system automatically
3. The same RPA process marks fueling reports as registered in DWH to avoid double registrations in the accounting system. The fueling reports remain available in DWH and can be used further for analytical purposes.
4. The accountant's role is just to control if reports has been read and registered correctly. Accountants use Power BI Service to control the registrations.
5. The updated reporting routine eliminated almost 100% of manual input into the accounting system.

Fuel consumption report: screenshots

Drivstoff inndata (liter)

Drivstoff forbruk og beholdning

Fartøy	IB	Bunkring	Forbruk	UB
MF Fosen	35500	0	0	35500
MF Frøynferja	5790	20979	21792	4977
MF Geisnes	6500	33000	19500	20000
MF Harانes	8770	0	4844	3926
MF Hertzog	0	0	0	0
Totalt	345567	1041722	1064218	323071

Totalt forbruk
1064218 !
Forrige måned: 818072 (+30 %)

Forbruk per fartøy
50677 !
Forrige måned: 38956 (+30 %)

Forbruk per km
10,4 !
Forrige måned: 8,0 (+30 %)

Bunkring

Fartøy	Date	Amount
MF Harانes	01.09.2020	2780
MF Levanger	01.09.2020	12043
MF Vikna	01.09.2020	25000
MS Tarningen	01.09.2020	4660
MS Trondheimsfjord 1	01.09.2020	3327
MS Trondheimsfjord 2	01.09.2020	2362
MS Tyrhaug	01.09.2020	8495
MS Vetlefjord	01.09.2020	1999
MF Harانes	02.09.2020	2480
MS Frøyfart	02.09.2020	1400
MS Osenfjord	02.09.2020	1901
MS Tarningen	02.09.2020	4420
MS Trondheimsfjord 1	02.09.2020	3753
MS Trondheimsfjord 2	02.09.2020	1895
MS Tyrhaug	02.09.2020	3406
MS Vetlefjord	02.09.2020	1084
MF Frøyfart	03.09.2020	5442
MF Harانes	03.09.2020	2240
MS Tarningen	03.09.2020	7912
MS Trondheimsfjord 1	03.09.2020	3524
MS Trondheimsfjord 2	03.09.2020	2392
Totalt		1127415

Registrert beholdning

Fartøy	Date	Amount
MF Frøynferja	01.09.2020	5790
MF Logstun	01.09.2020	47600
MF Levanger	01.09.2020	25500
MF Munken	01.09.2020	32300
MF Olav Duun	01.09.2020	26200
MF Trondheim	01.09.2020	88700
MF Vikna	01.09.2020	6020
MF Vikran	01.09.2020	32500
MS Osenfjord	01.09.2020	985
MS Tarningen	01.09.2020	4210
MS Trondheimsfjord 1	01.09.2020	2350
MS Trondheimsfjord 2	01.09.2020	1840
MS Tyrhaug	01.09.2020	6560
MS Vetlefjord	01.09.2020	2770
MF Frøynferja	02.09.2020	10400
MF Geisnes	02.09.2020	6500
MF Logstun	02.09.2020	46610
MF Munken	02.09.2020	30400
MF Olav Duun	02.09.2020	23300
MF Trondheim	02.09.2020	85600
MF Vikna	02.09.2020	5890
MF Harانes	02.09.2020	30100

Ikke registrert beholdning

Fartøy	Date	Dist. km.
MF Geisnes	01.09.2020	38
MF Levanger	02.09.2020	71
MF Geisnes	03.09.2020	39
MF Levanger	04.09.2020	106
Totalt		1475

Ikke registrert behodning

Fartøy	Antall
MF Geisnes	3
MF Levanger	3
MF Fosen	2
MS Frøyfart	2
Totalt	14

Kommentar

Drivstoff statistikk

Total fordruk (liter) Utvalg 1

Forbruk per fartøy (liter) Utvalg 1

Total forbruk (liter) Utvalg 2

Forbruk per fartøy (liter) Utvalg 2

BI: TRAFFIC REPORTS AND PASSENGER STATISTICS

Objective

To automate reporting process to AtB. AtB is one of the biggest transportation companies in Norway. All the companies that use AtB own ferry and bus lines as subcontractors, must report in accordance with predefined rules and procedures.

Challenges

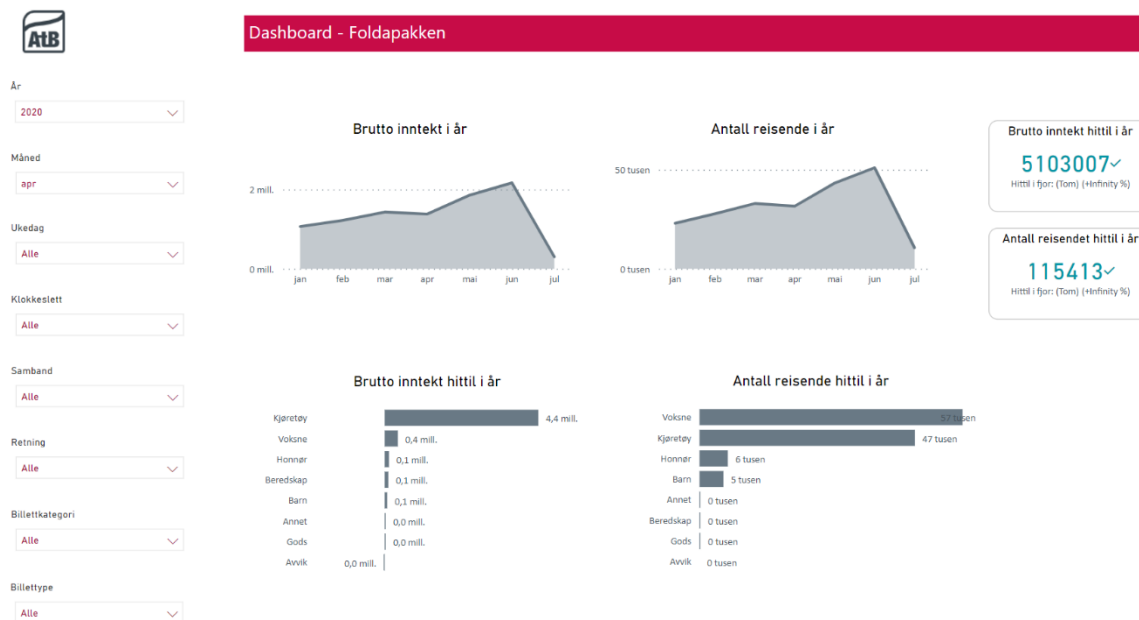Source data from different ticketing systems had to be read into a unified fact and dimension tables in DWH.

Original reporting process

- Raw data had to be extracted from different ticketing systems manually.
- Some of the ticketing system owners did not wish to grant direct access to the back-system / raw data depositories
- Once-a-month update

Optimized reporting process

- Export of source data (sales transactions) to DWH have been automated using both SSIS packages and RPA processes (where direct access to the ticketing systems database was not possible)
- Reporting is now made through Power BI Service, where reports are shared with AtB officials.
- The ETL and update processes are fully automated.
- Reports are updated daily.

Traffic reports and passenger statistics: screenshots

Traffic reports and passenger statistics: screenshots

AtB

Trafikkstatistikk - Foldapakken

År
2020

Måned
apr

Ukedag
Alle

Klokkeslett
Alle

Samband
Alle

Retning
Alle

Billettkategori
Alle

Billettype
Alle

Trafikkstatistikk

Nr	Billettype	Antall	Brutto NOK	MVA NOK	Netto NOK
1	Barn	971	17 815,97	540,71	16 807,55
1	Barn under 4 år	265	0,00	0,00	0,00
2	Bilfører	12951	0,00	0,00	0,00
2	Honnør	1710	33 939,00	925,52	32 016,21
2	Voksen	2900	106 409,92	2 842,06	100 387,21
3	Bil 10m	310	76 078,80	947,63	71 772,59
3	Bil 12m	261	81 064,60	504,25	76 475,31
3	Bil 14m	11	3 565,60		3 363,75
3	Bil 17m	244	125 944,40	1 797,78	118 814,91
3	Bil 19m	148	70 351,20	505,70	66 368,92
3	Bil 22m	2	1 274,40		1 202,26
3	Bil 6m	10341	693 704,00	19 836,16	654 438,07
3	Bil 7m	170	31 440,70	688,88	29 661,42
3	Bil 8m	140	25 565,70	444,15	24 118,65
3	Bil m/ henger <10m	589	75 375,20	1 843,89	71 108,45
3	Bil m/ henger >10m	43	9 032,10	212,76	8 520,90
3	El. bil 6m	514	14 218,20	308,61	13 414,05
4	MC	49	2 423,20	0,00	2 379,26
6	Beredskap X-tur	1	5 480,00	310,19	5 169,81
6	X-Beredskapstur VCH	5	27 400,00	1 550,95	25 849,05
7	Avvik bankkort	0	-20 107,00	0,00	-20 107,00
8	Gjenstående 10m	1	0,00	0,00	0,00
8	Gjenstående 12m	2	0,00	0,00	0,00
Totalt		**31644**	**1 380 975,99**	**33 259,24**	**1 301 761,37**

Reisende / klokkeslett

Ukedag
Alle
- man.
- tir.
- ons.
- tor.
- fre.
- lør.
- søn.

Reisende / ukedag

Klokkeslett
Alle

Billettype "Bilfører" = antall billetter solgt under kategori "Kjøretøy". Registreres ikke i billettsystemet

---

AtB

Sammendrag Flakk-Rørvik

År
2020

Måned
august

Kategori
Alle

Håndholdte lesere, brikke

Kategori	Brutto	MVA sats	MVA	Netto
Håndholdt, Billett Pris STL	264 974,60	12%	28 391,48	236 583,12
Håndholdt, Bompenger STL	81 667,20	0%	0,00	81 667,20
Håndholdt, Gebyr STL	0,00	0%	0,00	0,00
Håndholdt, Billett Pris EASYGO	597,60	12%	64,04	533,56
Håndholdt, Bompenger EASYGO	345,60	0%	0,00	345,60
Håndholdt, Gebyr EASYGO	-16,52	0%	0,00	-16,52
Totalt	**347 568,48**		**28 455,52**	**319 112,96**

Veikant, brikke

Kategori	Brutto	MVA sats	MVA	Netto
Veikant, Billett Pris STL	4 174 603,30	12%	447 231,19	3 727 372,11
Veikant, Bompenger STL	1 939 814,40	0%	0,00	1 939 814,40
Veikant, Gebyr STL	-13,39	0%	0,00	-13,39
Veikant, Billett Pris EASYGO	4 819 424,40	12%	516 430,33	4 302 994,07
Veikant, Bompenger EASYGO	2 445 379,20	0%	0,00	2 445 379,20
Veikant, Gebyr EASYGO	-126 548,43	0%	0,00	-126 548,43
Veikant, FULLPRIS	970 995,00	12%	104 037,08	866 957,92
Veikant, FULLPRIS bompenger	537 312,00	0%	0,00	537 312,00
Totalt	**14 760 966,48**		**1 067 698...**	**13 693 267,88**

Kontant

Kategori	Brutto	MVA sats	MVA	Netto
Billettsalg	45 356,00	12%	4 860,33	40 495,67
Bompenger	19 872,00	0%	0,00	19 872,00
Til innbetaling	**65 228,00**		**4 860,33**	**60 367,67**

Bankkort

Kategori	Brutto	MVA sats	MVA	Netto
Billettsalg	258 858,00	12%	27 737,28	231 120,72
Bompenger	71 232,00	0%	0,00	71 232,00
Avvik*	98,00	0%	0,00	98,00
Til innbetaling	**330 188,00**		**27 737,28**	**302 450,72**

* - avvik mellom billettmaskin og betalingsterminal

BI: ELECTRICAL BUSES. BATTERY CAPACITY AND CALCULATION OF OPTIMAL SPEED

Objective

Finding of actual battery capacity for each vehicle and its dependence on the outer temperature. Calculation of optimal driving patterns under given weather conditions. Comparison of the actual metrics with those declared by the vendor.

Reporting process
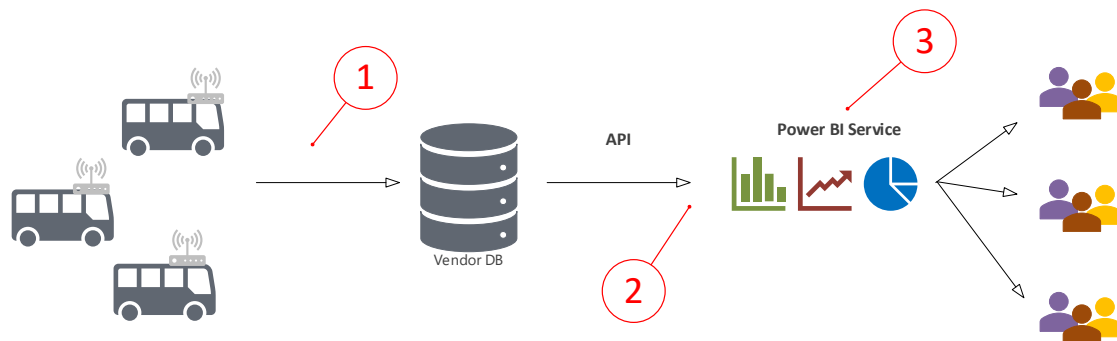
1. Vehicles communicate regularly with dedicated database to transfer the data (speed, temperature, battery charge etc.)
2. Power BI Service report is updated automatically using vendor's API.
3. Reports are available to end users (management, bus drivers etc.)

Battery capacity and calculation of optimal speed: screenshots
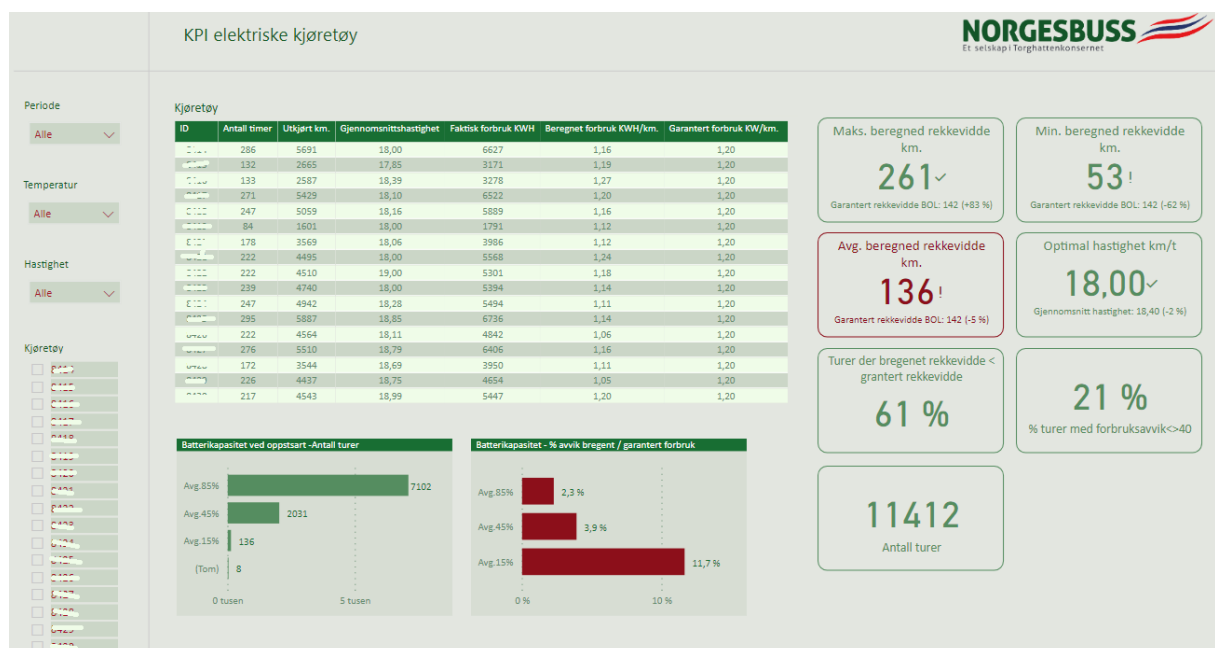
Battery capacity and calculation of optimal speed: screenshots
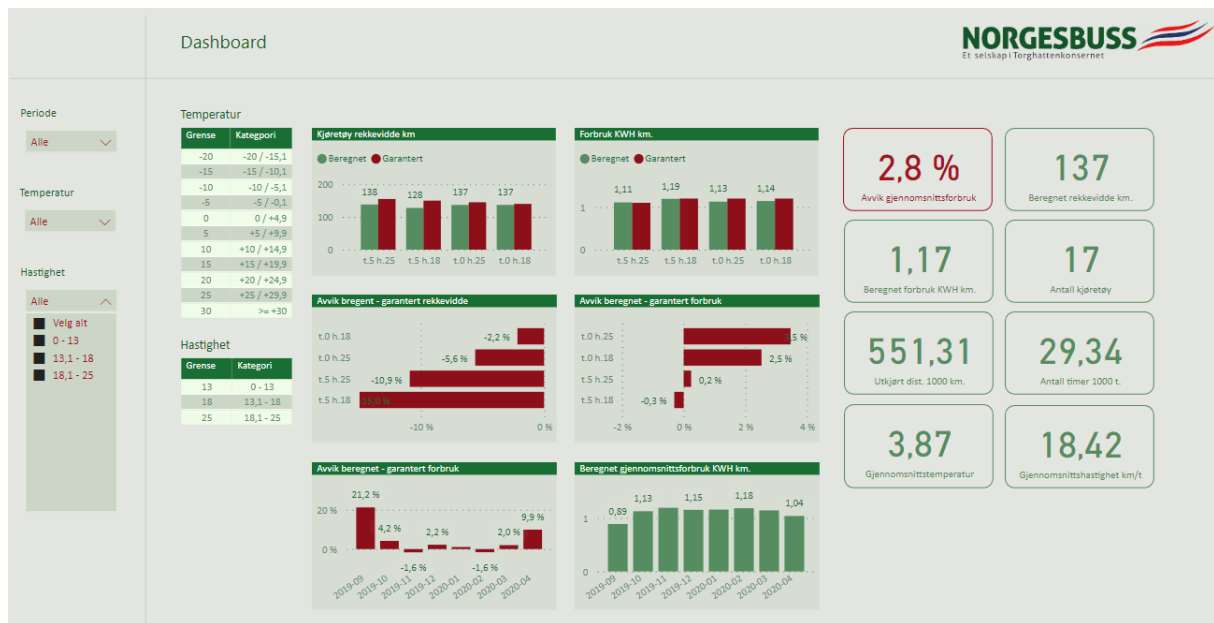
BI: ACCOUNTING REPORTS

Objective

To extend the capacity of the accounting systems reporting module and to grant access to the accounting reports to extern users without granting full access to the accounting system. To provide a better and more clear graphical presentation of accounting metrics.

Original reporting process

Subsidiaries used accounting system's original reporting module that was quite professional and technically advanced. However, it was difficult for a user without any technical background to develop a customized report. The original module was also lacking those reach graphical facilities that are available in Power BI.

Optimized reporting process

Power BI Accounting reports were developed as an operative tool, that apart from other operative information, contains annual and quarterly reports. This reporting tool was devised rather as an extension to an OLTP system than a stand-alone analytical tool. Reports read raw data directly from accounting system using direct SQL queries and can be updated on demand by users themselves through Power BI API and Gateway.
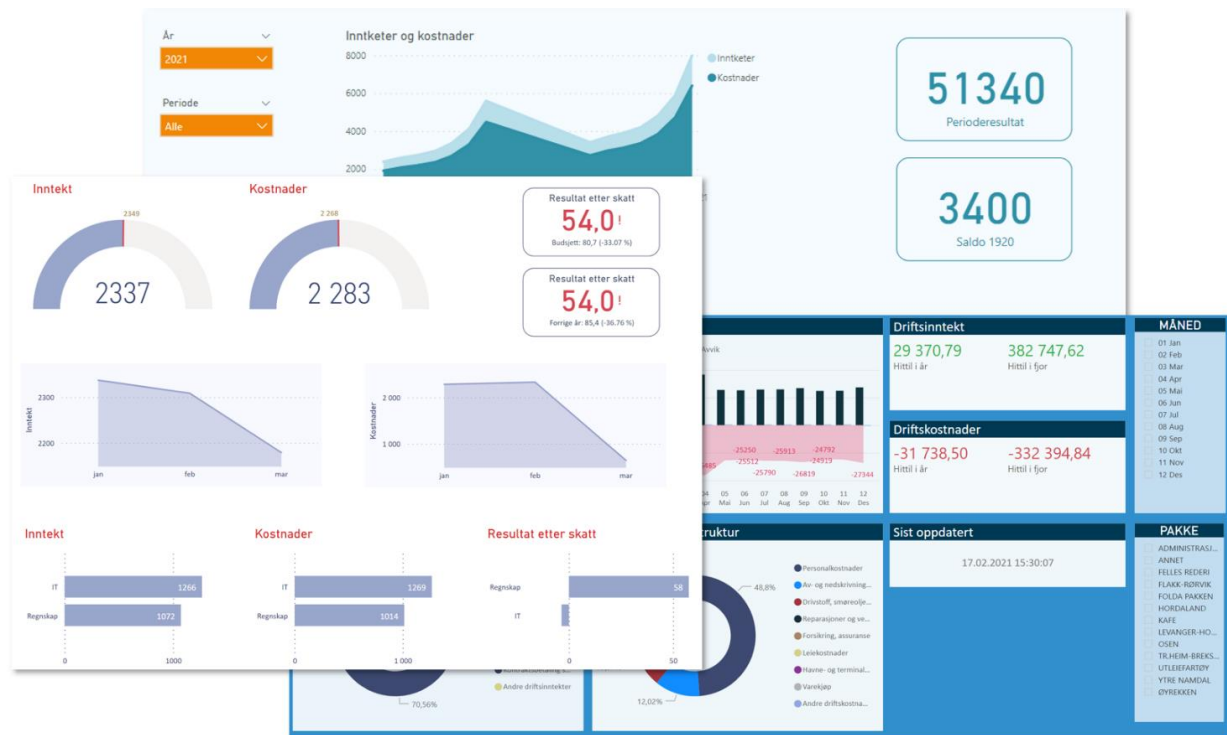
PROCESS AUTOMATION: E-INVOICING (EHF)

Objective
To automate invoicing process across the holding and to improve control and traceability of both outgoing and incoming invoices.

E-invoicing
E-invoices are transferred automatically from issuing to receiving company in form of specially formatted XML-files. These files are automatically authorized in specialized clearing centers and clearing centers issue transfer reports that show the fact of approval or denial of e-invoices. These reports are sent to the issuer and this, among other things, ensures full traceability of the invoicing process.

E-invoicing eliminates many of manual processes and much of paperwork. In addition, state-run organizations in Norway send and receive invoices only electronically.

E-invoicing process:

Issuing company → E-invoice → Clearing Center → E-invoice → Recieving company

Transfer report

E-invoicing, control report: screenshots

Information about outgoing and incoming e-invoices across the holding (Power BI Service):
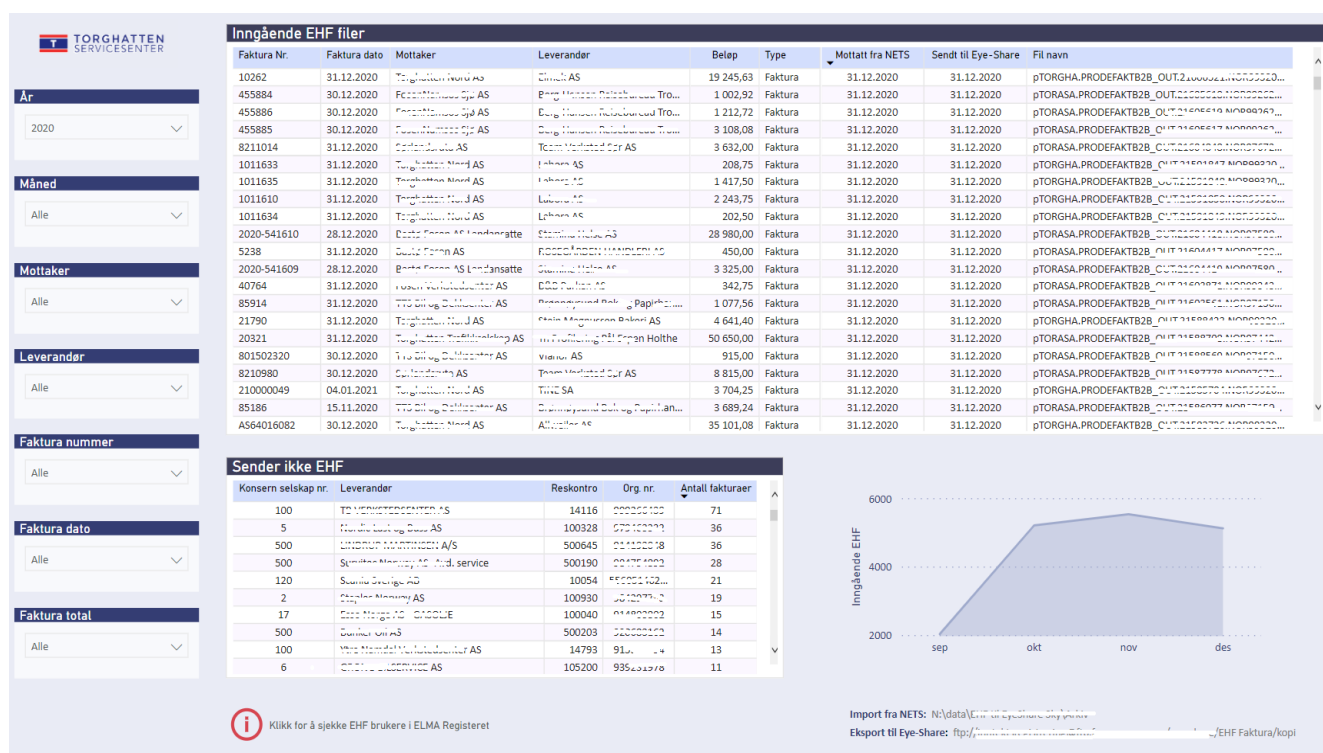
Transfer reports received from the clearing center (Power BI service):

Report users can check directly from the report, if a Norwegian vendor or supplier can issue or receive e-invoices:

The report shows the number of e-invoices that are being sent and received across the holding, in relation to the number of paper invoices and invoices sent through email:
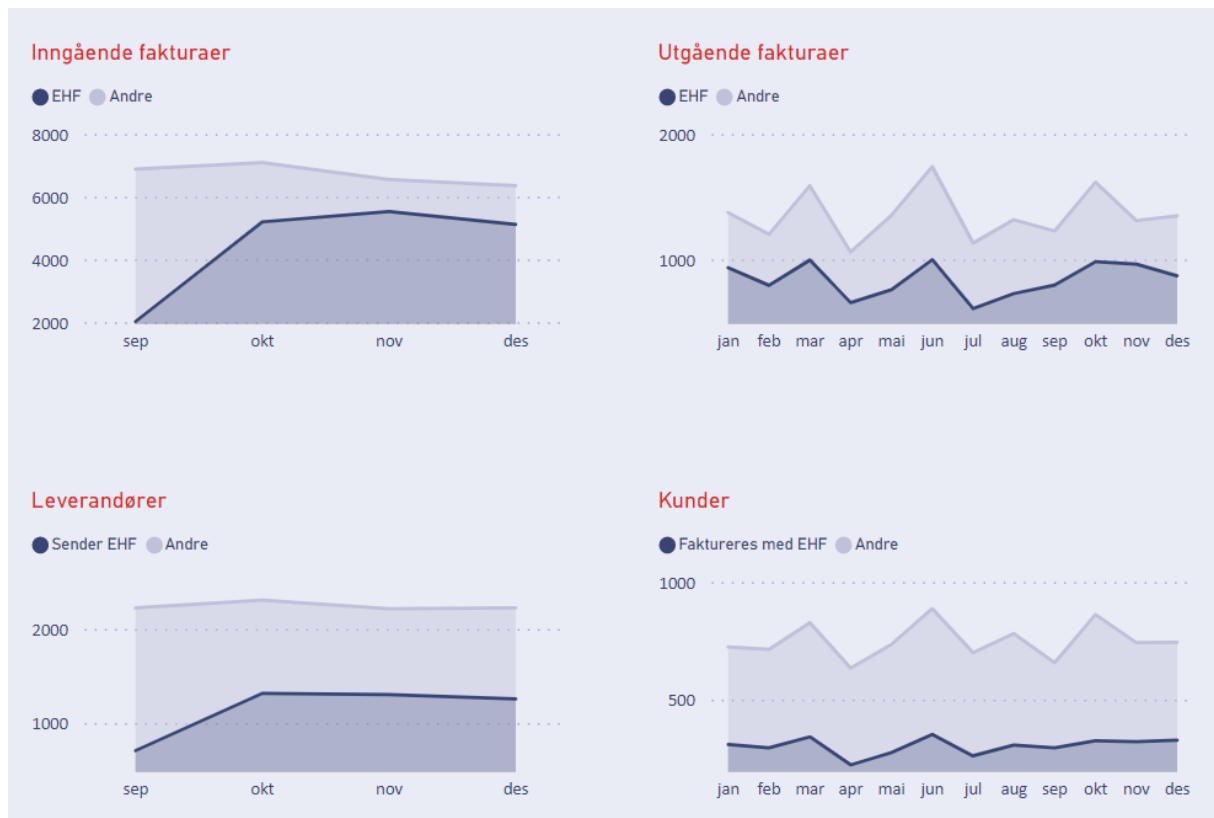
PROCESS AUTOMATION: TICKETING REPORTS (AZURE APP SERVICE + AZURE SQL)

Objective

Ferry crews must report ticket sales daily. The reporting process involved a great deal of paperwork and much of manual input. Crews had to fill out paper reporting schemes and send them to the central office.

Accountants in the office had to register reports in the accounting system manually. An average ferry company could generate as many as 30 000 sales transactions per month. The holding owns 5 ferry companies which use 3 different types of ticketing systems.

The goal was to eliminate 100% of paperwork and 90% of manual work.

Optimized reporting process

Tools: Azure App Service (ASP.NET Core), Azure SQL, RPA (Blue Prism). Manual reporting and accounting process has been fully eliminated. Sales reports are being exported automatically from the ticketing system to a dedicated OLTP database (Azure SQL). Sales transactions are then mapped with corresponding account numbers and exported automatically to the accounting system. The process can be controlled by both crews and accountants through a dedicated web application.

It is still possible to register ticket sales manually for example in case of communication failure in the ticketing system (a sale transaction has been registered in the ticket-machine but has not been transferred to the dedicated back system)

Ticketing reports: screenshots

Dato	Skift Nr.	Total FARA NOK	Voucher NOK	Total Bank NOK	Korrigering NOK	Utbetale til FNS NOK	Innkastnr. kontant	Dato innbet / safe	Kommentar	Fartøy	
04.07.2021	1188	7328.00	0.00			7328.00					Åpent
03.07.2021	1187	8757.00	0.00			8757.00					Åpent
02.07.2021	1186	9522.00	0.00			9522.00					Åpent
01.07.2021	1185	2199.00	0.00			2199.00					Åpent
30.06.2021	1184	3594.00		3594.00		0.00				MS Frøyfart	Eksportert
29.06.2021	1183	1184.00		1184.00		0.00				MS Frøyfart	Eksportert
28.06.2021	1182	3633.00		3633.00		0.00				MS Frøyfart	Eksportert
	Total:	36217.00	0.00	8411.00	0	27806.00					

Ticketing reports: screenshots

BILLETTOPPGJØR FNS

Alle operatorer
Uke rapport

Uke: 26   År: 2021   Søk   Last ned Excel   Ukerapport per operator

Operator	Dato	Skift Nr.	Total FARA NOK	Voucher NOK	Total Bank NOK	Korrigering NOK	Utbetale til FNS NOK	Innkastnr. kontant	Dato innbet / safe	Kommentar	
40703	04.07.2021	1188	7328.00	0.00			7328.00				Åpent
40743	04.07.2021	688	1240.00	0.00			1240.00				Åpent
40715	04.07.2021	11	17527.00	0.00			17527.00				Åpent
177006	04.07.2021	255	1180.00		1180.00		0.00				Eksportert
177004	04.07.2021	178	310.00	0.00			310.00				Åpent
40703	03.07.2021	1187	8757.00	0.00			8757.00				Åpent
40715	03.07.2021	10	8515.00	0.00			8515.00				Åpent
177004	03.07.2021	177	424.00	0.00			424.00				Åpent
40743	03.07.2021	687	1091.00	0.00			1091.00				Åpent


```csharp
namespace FNS_SalesRapport
{
    2 references
    public class Startup
    {
        private IConfiguration config = null;

        0 references
        public Startup(IConfiguration config)
        {
            this.config = config;
        }

        0 references
        public void ConfigureServices(IServiceCollection services)
        {
            services.AddControllersWithViews();
            services.AddDbContext<SalgRptDbContext>(options => options.UseSqlServer(this.config.GetConnectionString("FNS_SalgConnection")));
            services.AddSingleton<DataProtectionPurposeStrings>();
        }

        0 references
        public void Configure(IApplicationBuilder app, IWebHostEnvironment env)
        {

            if (env.IsDevelopment())

            {

                app.UseDeveloperExceptionPage();

            }

            //app.UseExceptionHandler("/Error");
            //app.UseDeveloperExceptionPage();
            app.UseStatusCodePages();
            app.UseStaticFiles();


            app.UseRouting();
```

FNS_SalesRapport
- Connected Services
- ▷ Dependencies
- ▷ Properties
- ▷ wwwroot
- ▲ Controllers
  - ▷ HomeController.cs
- ▲ Models
  - ▷ BankRapport.cs
  - ▷ Operator.cs
  - ▷ SaleRapport.cs
  - ▷ SaleRapport_Table.cs
  - ▷ SalgRptDbContext.cs
  - ▷ SambName.cs
  - ▷ SkipName.cs
  - ▷ TestID.cs
- ▷ Security
- ▲ Views
  - ▲ Home
    - Admin.cshtml
    - Edit.cshtml
    - Index.cshtml
    - ▷ Login.cshtml
    - ManueltSkift.cshtml
    - OperatorEndring.csh
    - OperatorListe.cshtml
    - UkeRapporter.cshtml
  - ▷ Shared
    - _ViewImports.cshtml
    - _ViewStart.cshtml
- ▷ appsettings.json

Ticketing reports: screenshots

Azure services

Create a resource	Resource groups	App Services	Subscriptions	Azure Active Directory	SQL servers	Azure SQL	Data Lake Storage Gen1	SQL databases
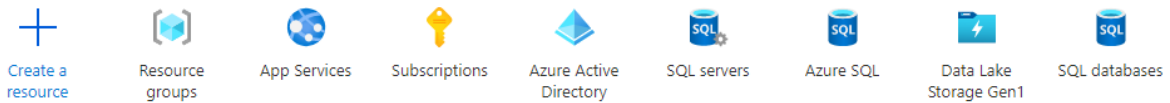
Recent resources

Name	Type	Last Viewed
TSS_Dev	Resource group	a month ago
Torghatten Azure Enterprise	Subscription	a month ago
FNSSalesRapport20201120101716PL...	App Service plan	a month ago
FNSSalesRapport	App Service	a month ago
FNS_SalesRapport_db	SQL database	4 months ago
KESalesRapport20210120150227	App Service	5 months ago

PROCESS AUTOMATION: TICKETING REPORTS (DESKTOP + MS SQL)

Objective

Bus drivers must report ticket sales daily. The reporting process involved a great deal of paperwork and a lot of manual input. The drivers had to fill out paper reporting schemes and send them to the central office. Accountants in the office had to register reports in the accounting system manually.

The goal was to eliminate 100% of paperwork and 90% of manual work.

Optimized reporting process

Tools: Desktop App (C#), MS SQL, RPA (Blue Prism). Manual reporting and accounting process has been fully eliminated. Sales reports are being exported automatically from the ticketing system to a dedicated OLTP database (MS SQL). Sales transactions are then mapped with corresponding account numbers and exported automatically to the accounting system. The process can be controlled by accountants through a dedicated desktop application.

It is still possible to register ticket sales manually for example in case of communication failure in the ticketing system (a sale transaction has been registered in the ticket-machine but has not been transferred to the dedicated back system)

Ticketing reports: screenshots

Rutiner og innstillinger

Sjåføroppgjør

Eksport og oppdatering

Innstillinger

Versjon: 1.0.0.47

Filtre

Selskap: 964 ▾   Operator: Med skift > 0 ▾   Periode 202105 ▾   Avdeling: Tromsø   **Søk**

Status 964-202105

Avsluttet: JA   Eksportert: JA   Fakturagrunnlag gods: JA   Periodeavslutning

Sjåføroppgjør | Rekvisisjoner | Gods | Åpne poster

Dobbeltklikk rad for å redigere skiftet   Selskap: 964   Periode: 202105   Operator: —   Null skift   Database oppdatert: 06.07.2021   Last ned Excel

ID Skift nr og Dato	Kontant FR	Voucher FR	Bank Krt FR	Modif. FR	TOTAL FARA	Kontant OP	Kontant Dato	Voucher OP	Bank Krt OP	TOTAL OPPG	Avvik	Kommentar
9641006 451 02.05.2021	0	0	135	0	135	0		0	135	135	0	
9641006 453 06.05.2021	0	0	1530	0	1530	0		0	1530	1530	0	
9641006 454 06.05.2021	0	0	30	0	30	0		0	30	30	0	
9641006 455 07.05.2021	0	0	855	0	855	0		0	855	855	0	
9641006 456 10.05.2021	0	0	585	0	585	0		0	585	585	0	
9641006 461 11.05.2021	0	0	45	0	45	0		0	45	45	0	
9641006 463 12.05.2021	0	0	1040	0	1040	0		0	1040	1040	0	
9641006 466 21.05.2021	0	0	195	0	195	0		0	195	195	0	
9641006 468 22.05.2021	0	0	135	0	135	0		0	135	135	0	
9641006 469 24.05.2021	0	0	495	0	495	0		0	495	495	0	
9641006 472 25.05.2021	0	0	855	0	855	0		0	855	855	0	
9641006 473 26.05.2021	0	6050	0	0	6050	0		6050	0	6050	0	
Total ID 9641006	0	6050	5900	0	11950	0		6050	5900	11950	0	
9641007 337 03.05.2021	0	0	210	0	210	0		0	210	210	0	
9641007 338 04.05.2021	0	0	255	0	255	0		0	255	255	0	
9641007 339 05.05.2021	0	0	315	0	315	0		0	315	315	0	
9641007 340 05.05.2021	0	0	720	0	720	0		0	720	720	0	
9641007 341 10.05.2021	0	0	630	0	630	0		0	630	630	0	
9641007 342 11.05.2021	0	0	300	0	300	0		0	300	300	0	
9641007 343 12.05.2021	0	0	1494	0	1494	0		0	1494	1494	0	
9641007 344 18.05.2021	0	0	180	0	180	0		0	180	180	0	
9641007 345 19.05.2021	0	0	225	0	225	0		0	225	225	0	
9641007 346 19.05.2021	0	0	90	0	90	0		0	90	90	0	
9641007 347 19.05.2021	0	0	540	0	540	0		0	540	540	0	
9641007 348 25.05.2021	0	0	1080	0	1080	0		0	1080	1080	0	
9641007 349 26.05.2021	0	0	405	0	405	0		0	405	405	0	
9641007 350 26.05.2021	0	0	1435	0	1435	0		0	1435	1435	0	

	Kontant FR	Voucher FR	Bank Krt FR	Modif. FR	TOTAL FR	Kontant OP		Voucher OP	Bank Krt OP	TOTAL OPPG	Avvik	
TOTAL PERIODE:	135	19860	194434	-135	214294	0		19860	194434	214294	0	

Ticketing reports: screenshots

Filtre
Selskap: 964   Operator: Med skift > 0   Periode: 202105   Avdeling: Tromsø   [Søk]

Status 964-202105
Avsluttet: JA   Eksportert: JA   Fakturagrunnlag gods: JA   [Periodeavslutning]

Tabs: Sjåføroppgjør | Rekvisisjoner | Gods | Åpne poster

Registrerte rekvisisjoner 964-202105   [Last ned Excel]

OPERATOR	SKIFT	ANTALL	BLP FR	BLP OP	AVVIK
9641006	473	3	6050	6050	0
9641020	753	1	3000	3000	0
9641053	317	2	1740	1740	0
9641060	469	1	570	570	0
9641060	471	2	1080	1080	0
9641060	472	1	450	450	0
9641060	473	1	450	450	0
9641061	676	1	810	810	0
9641079	395	1	1080	1080	0
9641109	206	2	4000	4000	0
9641138	43	1	630	630	0

Innleverte rekvisisjoner 964-202105   [Last ned Excel]

OPERATOR	DATO	ANTALL	BLP	KATEGORI
9641006	26.05.2021	1	6050	Billett
9641020	26.05.2021	1	3000	Billett
9641053	06.05.2021	2	1740	Billett
9641060	03.05.2021	1	570	Billett
9641060	05.05.2021	2	1080	Billett
9641060	06.05.2021	1	450	Billett
9641060	06.05.2021	1	450	Billett
9641061	20.05.2021	1	810	Billett
9641079	04.05.2021	1	1080	Billett
9641109	11.05.2021	1	4000	Billett
9641138	03.05.2021	1	630	Billett

Kontroll OK

KILDE	BLP
Registrert	19860
Innlevert	19860
Avvik	0

THB_driftsrapporter

- Database Diagrams
- Tables
  - System Tables
  - FileTables
  - External Tables
  - dbo.DlmTBL
  - dbo.FactTBL
  - dbo.tblAvdeling
  - dbo.tblAvregning
  - dbo.tblBetalTerminal
  - dbo.tblBlockKntrl
  - dbo.tblBrukerStedID
  - dbo.tblEMV
  - dbo.tblFara
  - dbo.tblGodsFara
  - dbo.tblGodsModif
  - dbo.tblHovedSelskap
  - dbo.tblNrDKT
  - dbo.tblOperator
  - dbo.tblOperatorRPA
  - dbo.tblOppgPapir
  - dbo.tblPayEx
  - dbo.tblPeriodeAvslut
  - dbo.tblRekvisisjonManuell
  - dbo.tblRute
  - dbo.tblTeller
  - dbo.tblTellerBrukerSted
  - dbo.tblAapnePost
- Views
- External Resources
- Synonyms
- Programmability
- Service Broker

Ticketing reports: screenshots

```csharp
1 reference
private void comboOperIdOgPeriodePopulate(ComboBox cmbID, ComboBox cmbPrd, int ruteId, Label lblAvd)
{
    if (ruteId == 0)
    {
        return;
    }

    string sqlStrRuteID = "select operid from tblfara where ruteid = " + ruteId + " group by operid order by 1";
    string sqlStrPrd = "select prd from vFara_PapirOppg where prd is not null and rutenr = " + ruteId + " group by prd order by 1 desc";
    string sqlStrAvd = "select rtrim(avdeling) as avdeling from tblrute where ruteID = " + ruteId;

    DataTable dt = new DataTable();

    //operator
    dt = ConstParam.SQLtoDt(sqlStrRuteID, ConstParam.ConnStringMDb, "t1");
    cmbID.Items.Clear();

    if (dt.Rows.Count > 0)
    {
        cmbID.Items.Add("Med skift > 0");
        cmbID.Items.Add("Med alle skift");
        cmbID.Items.Add("Kun med avvik");
        cmbID.Items.Add("Kun med voucher > 0");
        cmbID.Items.Add("Kun med kontant > 0");

        foreach (DataRow r in dt.Rows)
        {
            cmbID.Items.Add(r[0].ToString());
        }

        cmbID.SelectedIndex = 0;
        dt.Rows.Clear();
        dt.Columns.Clear();
    }

    //periode
    dt = ConstParam.SQLtoDt(sqlStrPrd, ConstParam.ConnStringMDb, "t1");
    cmbPrd.Items.Clear();

    if (dt.Rows.Count > 0)
    {
        foreach (DataRow r in dt.Rows)
        {
            cmbPrd.Items.Add(r[0].ToString());
        }

        cmbPrd.SelectedIndex = 0;
        dt.Rows.Clear();
        dt.Columns.Clear();
    }
}
```
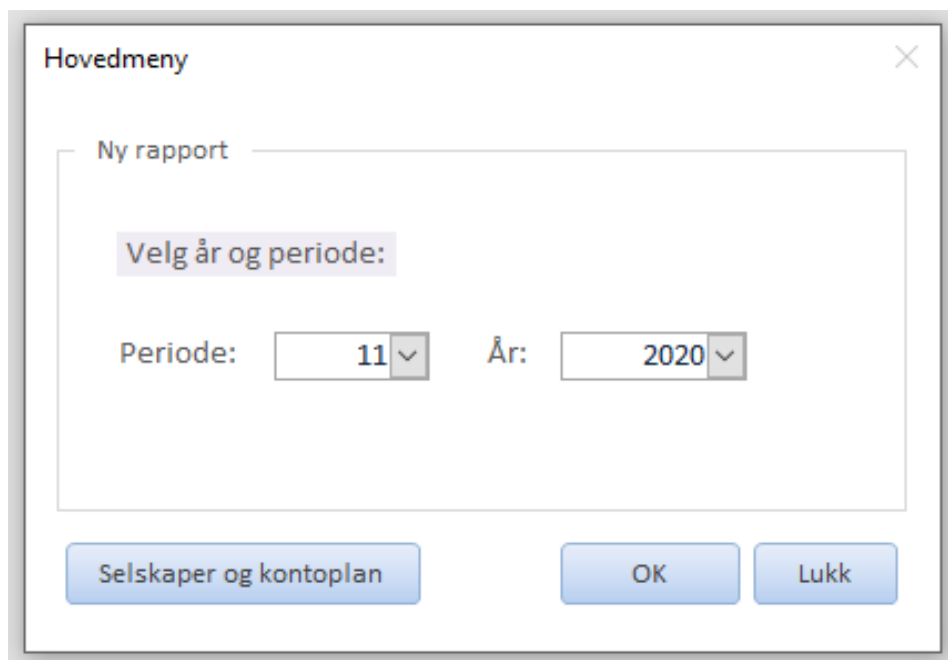
PROCESS AUTOMATION: RECONCILIATION OF GROUP BALANCES, ACCESS + VBA

Description

Reconciliation of group accounts has been made manually and involved reconciliation of internal revenues and costs inside the holding and elimination of those posts from the consolidated report.
The main challenge in the process was to extract internal posts from the accounts that initially was not set up as group internal accounts in the charts of accounts.

The application that has been developed, had its own version of group account numbers, and could automatically query all the necessary posts from the accounting system. The reconciliation process has been considerably simplified and took now several hours to complete (the process took approximately 1 week per month before optimization)

Reconciliation of group accounts: screenshots

Reconciliation of group accounts: screenshots

Konsernmellomværende

Periode: 11 2020

Dobbeltklikk tilsvarende selskapsnavn:

Nr ▾	Selskapsnavn ▾	Eget regnskap ▾	Motst. regnskap ▾	Differanse ▾
1	Torghatten ASA		94̶ ̶ ̶5	-11̶ ̶ ̶ ̶2
2	Torghatten Trafikkselskap AS	-1̶ ̶ ̶6	-1̶ ̶ ̶0	-̶ ̶ ̶4
5	Torghatten Buss AS	̶ ̶ ̶1	̶ ̶ ̶0	̶ ̶ ̶
6	TTS Bil og Dekksenter AS	2̶0̶6 ̶0̶7̶8̶	̶ ̶ ̶4	1̶5 ̶ ̶
8	Sør-Helgeland Vaktselskap As	̶ ̶ ̶9	̶ ̶ ̶1	̶ ̶ ̶
11	Sørlandsruta AS	̶ ̶ ̶5	̶ ̶ ̶	̶ ̶ ̶
13	T-FINANS AS	2̶ ̶ ̶	̶ ̶ ̶2	̶ ̶ ̶9
15	Partsrederiet Kysteksspressen AI	-1̶ ̶ ̶	̶ ̶ ̶5	1̶ ̶ ̶6
17	Bastø Fosen AS	-8̶5 ̶ ̶6	̶ ̶ ̶0	-̶ ̶ ̶7
20	Torghatten Servicesenter AS	̶ ̶ ̶4	-̶ ̶ ̶3	̶ ̶ ̶
25	Fosen Verkstedservice AS	̶ ̶ ̶	-3̶ ̶ ̶	̶ ̶ ̶3
55	Tromsø Busscharter AS	̶ ̶ ̶	̶ ̶ ̶9	̶ ̶ ̶
75	FosenNamsos Sjø AS	-̶ ̶ ̶	̶ ̶ ̶2	6̶ ̶ ̶
100	TrønderBilene as	-2̶5 ̶0̶7̶1 ̶6̶	̶ ̶ ̶6	̶ ̶ ̶0̶0̶
500	Torghatten Nord AS	̶ ̶ ̶7̶2	̶ ̶ ̶0̶6	̶ ̶ ̶6

[Hoved meny] [Oppfrisk] [Lage XLS rapport]

| Sammendrag | Kunde - Leverandør | Kortsiktige | Langsiktige |

Periode: 11 2020

Hoved selkap:	Motstående selskap:	
Torghatten Trafikkselskap AS	**Torghatten ASA**	

Samlet balanse:		Samlet balanse:		Differanse:	
Kunde kontoer	̶5̶0̶0̶0̶	Leverandør kontoer	̶0̶1̶0̶0̶	Kunde - Leverandør	1̶2̶0̶0̶
Leverandør kontoer	-̶2̶0̶7̶0̶0̶0̶	Kunde kontoer	2̶0̶7̶0̶0̶0̶	Leverandør - Kunde	0
Kortsiktig gjeld	0	Kortsiktig fordring	0	Krt. gjeld - Krt. fordr.	0
Kortsiktig fordring	0	Kortsiktig gjeld	0	Krt. fordr. - Krt. gjeld	0
Langsiktig gjeld	0	Langsiktig fordring	0	Lng. gjeld - Lng.fordr.	0
Langsiktig fordring	2̶0̶0̶0̶0̶0̶0̶0̶	Langsiktig gjeld	-2̶0̶0̶0̶0̶0̶0̶0̶	Lng.forfr. - Lng. gjeld	0
				TOTAL DIFF:	1̶2̶0̶0̶

[<< Alle motst.] [Oppfrisk]

Reconciliation of group accounts: screenshots

Tabeller

- Knt_plan
- Knt_plan_og_beløp
- Knt_type
- Notater
- Periode
- Selskaps_nvn
- Trans_year

Første skjerm

- Hoved_periode_først
- Hoved_sls_total
- MAIN_QUERY
- Most_periode_først
- Motst_sls_total
- MAIN_QUERY-delskjema
- Total_MVM
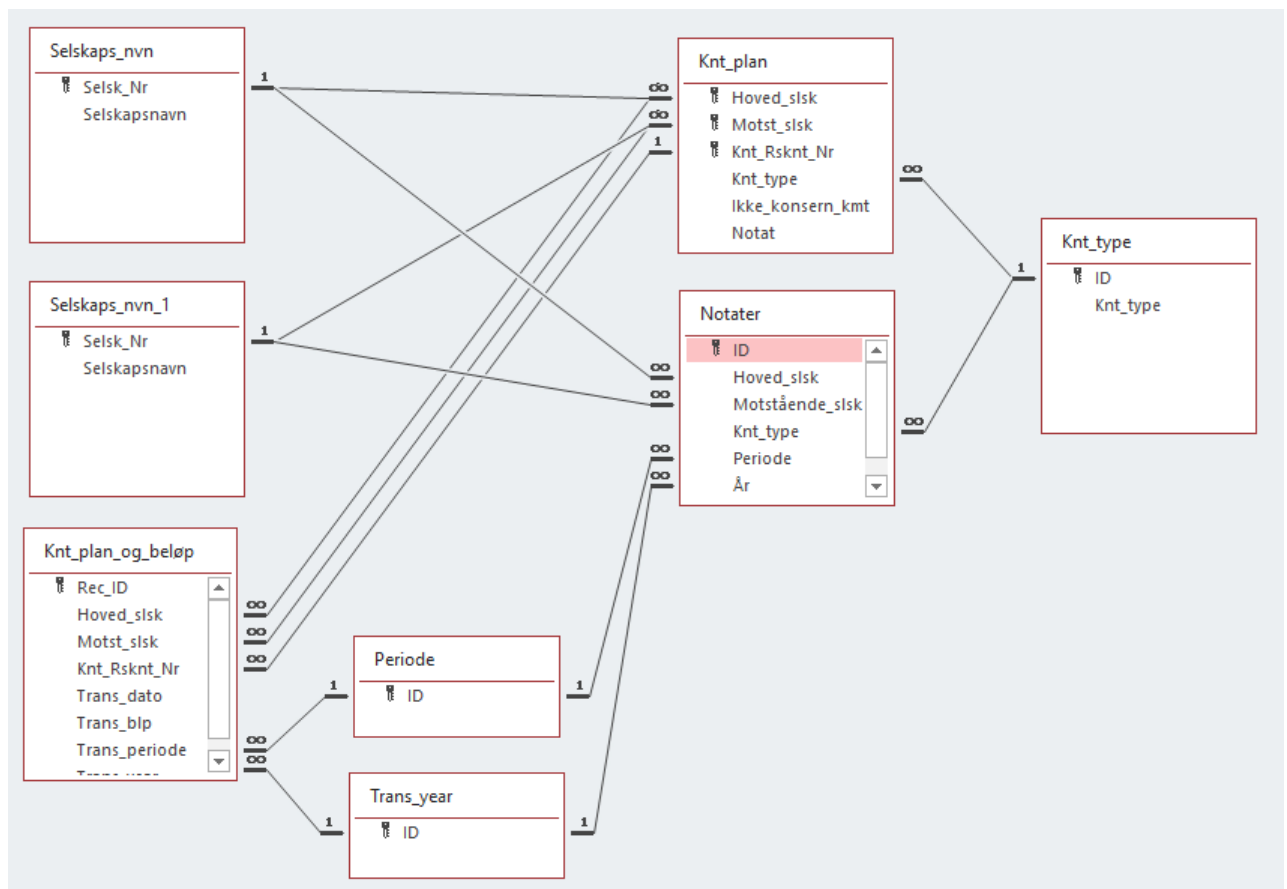
Andre skjerm

- Hoved_and_Motst
- Hoved_vs_Motst
- Motst_vs_Hoved
- Sum_ Hoved_and_Motst
- Hoved_and_Motst-delskjema
- Selskap_MVM

Tredje skjerm KUNDE-LEVER

- Hvd_vs_Mts_kunde1
- Hvd_vs_Mts_lev2
- Mts_vs_Hvd_kunde2
- Mts_vs_Hvd_lev1
- Hoved_Kunde_vs_Mot_lev
- Hvd_vs_Mts_kunde1-delskjema
- Hvd_vs_Mts_lev2-delskjema
- Mts_vs_Hvd_kunde2-delskjema1
- Mts_vs_Hvd_lev1-delskjema

HOVEDMENY

- Alle_selskaper
- Felles_kontoplan
- Alle konsernselskaper
- Alle_selskaper-delskjema
- Felles_kontoplan-delskjema1
- Kontoplan
- Ny rapport
- Selskaper og kontoer
- SlsMVMl_Velg_Oppfrisk
- Total_Velg_Oppfrisk
- Velg_Oppfrisk

Reconciliation of group accounts: screenshots

```vba
Sub InfoEasy()

Dim Prd As Integer ' rapportens perode
Dim Year As Long ' rapportens år

Dim Snvn_H_Knt(500, 10000) ' Konserninterne hovedkonti
Dim Snvn_R_Knt(500, 10000) ' Konserninterne reskontroer
Dim Svnv_H_Mnvn(500, 10000) ' konsernselskapsnavn og motstående sleskapsnavn / hovedkonti
Dim Svnv_R_Mnvn(500, 10000) ' konsernselskapsnavn og motstående sleskapsnavn / reskontroer
Dim i As Integer ' første matriseelement (500)
Dim a As Integer ' andre matrise element (10000)hovedkonto
'Dim b As Integer '' andre matrise element (10000)reskontro
Dim AntSls As Integer ' antall unike selskaper

Dim db As DAO.Database
Dim Rst As Recordset
Dim SqlStr As String ' sql string i Access
Dim y As Integer ' løkketeller

Prd = Forms![Total_Velg_Oppfrisk].Etikett19.Caption
Year = Forms![Total_Velg_Oppfrisk].Etikett20.Caption

' ta med alle selskapsnavn
Set db = CurrentDb()

Set Rst = CurrentDb.OpenRecordset("SELECT Selskaps_nvn.Selsk_Nr " & _
"FROM Selskaps_nvn " & _
"ORDER BY Selskaps_nvn.Selsk_Nr;")

With Rst
Do While Not .EOF
Snvn_H_Knt(i, a) = ![Selsk_Nr] ' selskapsnummer / hovedkonti
Snvn_R_Knt(i, a) = ![Selsk_Nr] ' selskapsnummer / reskontroer
i = i + 1
.MoveNext
Loop
End With
AntSls = i ' tilordner antall selskaper
i = 0

'***********H O V E D*****************************************************
For y = 0 To AntSls - 1 ' begynner å ta med alle tilsvarende HOVED kontonummre

a = 0
```

27

PROCESS AUTOMATION: ACCOUNTING INPUT, EXCEL + VBA

Description

This VBA-driven Excel model takes a csv file with accounting posts as an input, reworks the posts, maps them with corresponding accounting dimensions and produces an output in a form of a specifically structured text file (DKT-formatted file) that can be exported automatically to the accounting system (InfoEasy). The model helps to avoid hours of manual work in case when accountants must regularly register incoming external documents with hundreds or even thousands of typical accounting posts.

Accounting input: screenshots

Kildefil:	...TSS\Kundedokument...r\5 Torghatten Buss AS\LeaseWeb bilagregistrering\Klldefil tekst\
Kildefil arkiv:	...TSS\Kundedokument... \5 Torghatten Buss AS\LeaseWeb bilagregistrering\Klldefil tekst\Arkiv\
DKT fil:	...Data\dx\

Neste bilagsnummer:	96824
Neste DKT filnummer:	86
Periode:	
År:	
Faktura nummer:	
KID:	

Fil Import

DKT Eksport

Filimport ✕

Inndata

Bilagsnummer | 96824

Periode |

| 05-2021 |
| 06-2021 |
| 07-2021 |
| 08-2021 |
| 09-2021 |
| 10-2021 |

Fakturanummer |

KID |

Ok

Leasing DNB Torghatten Buss AS

Dato	Kode	TK	MVA kode	Tekst	Dim A	Dim B	Beløp	Konto

Konto mapping *

Kode	Navn	Konto
102	Utkjøp	0
104	Forsk. leie	6012
105	Utkjøp	0
109	Kilometeravregning	0
125	Avgiftskrav	0
215	Miljøgebyr	0
235	Park.bot	0
240	Overlast	0
245	Årsavgift	7508
246	Bomp.bøter	0
250	Km.avg.	0
252	Vektavgift	7508
254	Overkjørt km	0
255	Refusjon bot	0
256	Bot piggdekk	0
270	Månedlig leie	6012
271	Månedlig leie	6012
272	Merverdiavgift	2710
293	Øresavrunding	2999
298	Rentenota	8155

Accounting input: screenshots

```vb
Public Sub filExport_Excel() '!! textfil eksport til Excel model

Dim filSti As String 'filsti og filnavn
Dim filStiArkiv As String 'filsti arkiv
Dim filNavn As String 'filnavn
Dim filNum As Integer
Dim txtLinje As String
Dim fso As New FileSystemObject
Dim bilagPost() As String
Dim bilagTotal As Double
Dim radNr As Integer 'radnummer excel ark


filSti = Range("M3")
filStiArkiv = Range("M4")
filNavn = Dir(filSti)
radNr = 4

If filNavn <> "" Then ' hvis det finnes noe som skal eksporteres

  Range("A4:I617").Select
  Selection.ClearContents
  Range("A4").Select

End If


While filNavn <> "" ' hver fil i filsti

   ' hvis filen er txt
   If InStr(1, filNavn, ".txt", vbTextCompare) > 1 Then

      'Åpne filen
      FileNum = FreeFile()
      Open filSti & filNavn For Input As #FileNum
      bilagTotal = 0
```

RPA: PROJECT EXAMPLES

Objective

The holding uses Blue Prism as RPA tool (public information). The tool is used rather intensively to perform the following tasks:

- Read table-formed reports from internal and external web-based applications and transfer the data to a dedicated database
- Write input into the accounting system (new subaccount numbers, new posts, invoices, and accounting dimensions)
- Read voluminous and structured Excel reports, rework the data into accounting posts and register the posts in the accounting system
- To initiate a script or a pre-programmed event in a user application when it is not possible to automate this in some other way

There are more than 10 major RPA processes in use with two RPA licenses across the holding.

RPA: screenshots
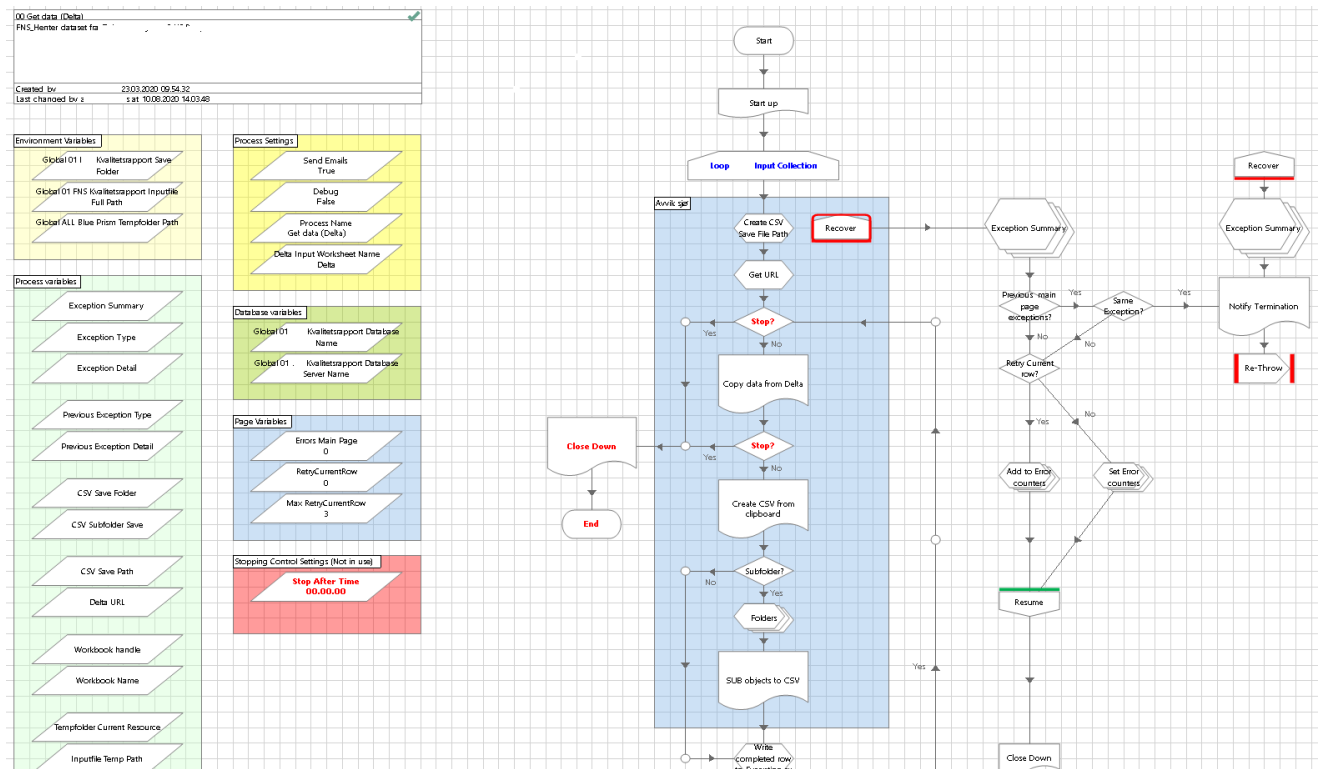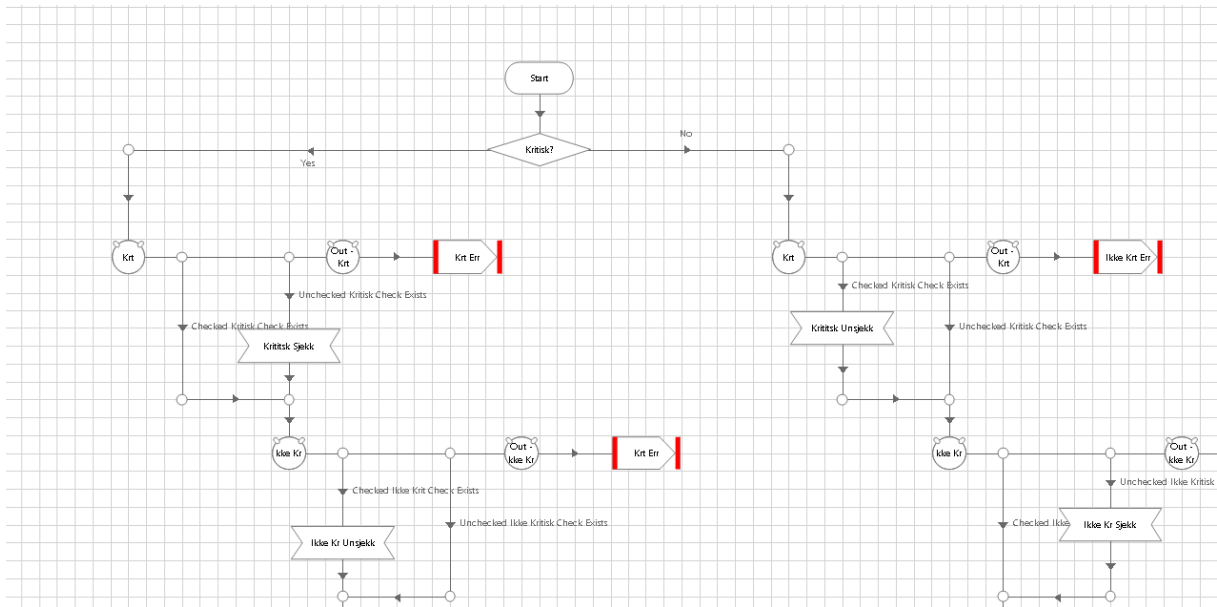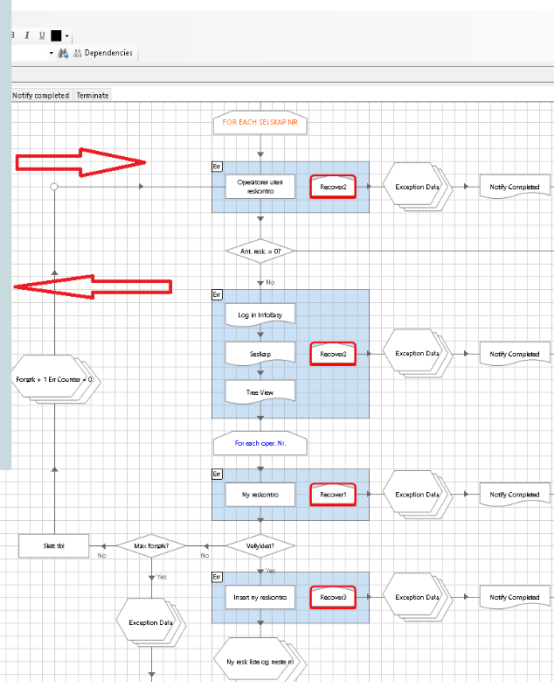
RPA: screenshots
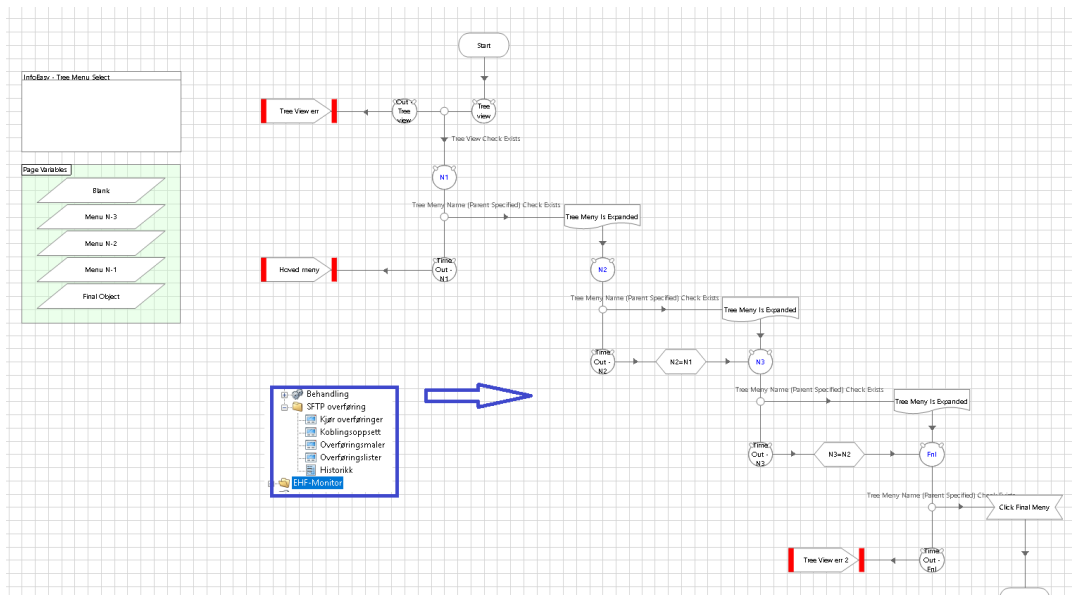
RPA: SOME OBSERVATIONS

- RPA is a brilliant technology which sometimes can be "the one and only" way to extract or export data automatically. It is quite reliable (if used properly), relatively cheap and as a rule has a short payback period (9-12 month).

- RPA tool, on the other hand, is not a "silver bullet" or "one-size-fits-all" solution. It seems that for example Blue Prism works perfectly well with applications where user interface was made using Microsoft products (C#) and can sometimes function unpredictably in an application where UI programming language is Java. Not all UI elements can be identified and not always. Even when all of them can be "found" by an RPA tool, it is not a 100% guaranty that the application itself can "tolerate" this (however this situation can be "cured" quite effectively by use of an integrated C# or VB.Net scripts inside of the RPA process)

- Best if an automation project does not rely completely and solely on the use of an RPA tool alone. A combination of different tools can be a more effective solution (for example a combination of SSIS packages and RPA tools)

- Despite some instructions that recommend not to use "pure" scripts (VB.NET, C#) inside of an RPA processes, are such scripts extremely helpful in some complex situation. In general, it can be said that an RPA developer should have at least basic understanding of programing concepts and patterns (such as for example reusability of code). Otherwise, it is highly likely that both RPA processes and the development process itself can at least become unmanageable after the number of developed objects and processes reaches not more then 10-15 pieces (which is not many, especially in a large company or holding)

LEVEL OF COMPETENCE: SQL

What I know and what I can

- All basic and some advanced CRUD operations
- Nested queries, unions, virtual tables
- Conditional operators (CASE WHEN etc.) and looping through a record set (virtual table)
- Ordering and grouping inclusive ROLLUP etc.
- Windows functions, all data types, use of variables etc.

```sql
-- ERFA ERFA ERFA ERFA
with alv_grad as
(
select  fv.fkObject
,dl.DisplayText
,lf.fkSelectionList
from delta_Lang dl
left join delta_SelectionListLeaf lf on dl.pkLang = lf.fkLang_Display
left join delta_FieldValue fv on lf.pkSelectionListLeaf = fv.fkSelectionListLeaf
where lf.fkSelectionList = 17
),
hend_type as
(
select  fv.fkObject
,dl.DisplayText
,lf.fkSelectionList
from delta_Lang dl
left join delta_SelectionListLeaf lf on dl.pkLang = lf.fkLang_Display
left join delta_FieldValue fv on lf.pkSelectionListLeaf = fv.fkSelectionListLeaf
where lf.fkSelectionList = 10
),
skjema_type as
(
select df.pkForm
,ot.SystemName
from delta_Form df
left join delta_ObjectType ot on df.fkObjectType = ot.pkObjectType
)
select convert(varchar,do.pkObject) + '_' + convert(varchar,GETDATE(),112) as RecID
```

```sql
declare @periode as int
set @periode = 201902

--avviste brikker
select Registered, PANID as [Brikke nr.],'Billett pris' as ChargeType,
case when Debit_Credit = 'D' then ChargedGrossAmount else ChargedGrossAmount*-1 end as [Brutto pris],
case when Debit_Credit = 'D' then ChargedAmountVATRate else ChargedAmountVATRate*-1 end as [MAV sats],
case when Debit_Credit = 'D' then ChargedAmountVAT else ChargedAmountVAT*-1 end as [MVA blp],
case when Debit_Credit = 'D' then ChargedNetAmount else ChargedNetAmount*-1 end as [Netto pris],
case when Debit_Credit = 'D' then OBUIssuerFee else OBUIssuerFee*-1 end as Gebyr,
Debit_Credit, InformationCode, STLPostingDate,
case when DeviceType = 1 then 'Veikant' else 'Håndholdt' end as [Billett. utstyr],
STLTicketVersionNo, TicketCodeCharged, ChargedType, DeviceID, (RecID + 'Billettpris') as RecID
from FR_brikke_transer
where datepart(year, Registered) * 100 + datepart(month, Registered) = @periode
and rtrim(ChargedType) = 'NOTACCEPTED'
and InformationCode not in (0,100)
union all
```

LEVEL OF COMPETENCE: C#, ASP.NET CORE

What I know and what I can

- All the basic programming concepts
- Classes, interfaces, delegates, and events
- MVC model
- LINQ
- Entity Framework and database programming

```csharp
//[ResponseCache(Location = ResponseCacheLocation.None, NoStore = true)]
0 references
public async Task<IActionResult> Index(string searchStringW, string searchStringY, string operID)//liste med alle skiftnumre
{

    ////this is regular operator
    //ViewBag.IsAdmin = 0;
    DateTime dtNow = DateTime.Now;
    int yrNow = dtNow.Year;

    //dropdown ukenumre + år
    List<SelectListItem> weeks = new List<SelectListItem>();
    List<SelectListItem> years = new List<SelectListItem>();
    string d_operID = protector.Unprotect(operID);

    foreach (var week in (db.vSaleRapports.AsEnumerable().Where(i => i.OperatorID == int.Parse(d_operID)).OrderByDescending(n => n.Aar*100 + n.UkeNr).Select(n => n.UkeNr).Distinct()))
    {
        if (!string.IsNullOrEmpty(week.ToString()))
        {
            weeks.Add(new SelectListItem { Value = week.ToString(), Text = week.ToString() });
        }
    }

    foreach (var yr in (db.vSaleRapports.AsEnumerable().Where(i => i.OperatorID == int.Parse(d_operID)).OrderByDescending(n => n.Aar).Select(n => n.Aar).Distinct()))
    {
        if (!string.IsNullOrEmpty(yr.ToString()))
        {
            years.Add(new SelectListItem { Value = yr.ToString(), Text = yr.ToString() });
        }
    }


    ViewBag.EncrOperId = operID;
    ViewBag.UkeNumre = weeks;
    ViewBag.AarNr = years;
    ViewBag.OperID = d_operID;
    ViewBag.ActUkeNr = searchStringW;
    ViewBag.EOperID = operID;
```

```csharp
5 references
public class SalgRptDbContext :DbContext
{
    0 references
    public SalgRptDbContext (DbContextOptions<SalgRptDbContext> options) : base(options) { }

    18 references
    public DbSet<SaleRapport> vSaleRapports { get; set; }

    6 references
    public DbSet<BankRapport> VERIFONE_bankkort { get; set; }

    3 references
    public DbSet<SaleRapport_Table> SaleRapports { get; set; }

    2 references
    public DbSet<SkipName> SkipNvn { get; set; }

    2 references
    public DbSet<SambName> SambNvn { get; set; }

    0 references
    public DbSet<TestID> TestID { get; set; }

    15 references
    public DbSet<Operator> Primitve_LogIn { get; set; }

    //kyes
    0 references
    protected override void OnModelCreating(ModelBuilder modelBuilder)
    {
        modelBuilder.Entity<SaleRapport>().HasNoKey();
        modelBuilder.Entity<BankRapport>().HasKey(k => new { k.BAX, k.Avstem_nr, k.Avstem_dato});
        modelBuilder.Entity<SaleRapport_Table>().HasKey(k => new { k.OperatorID, k.SkiftNr, k.Skift_Dato });
    }
```
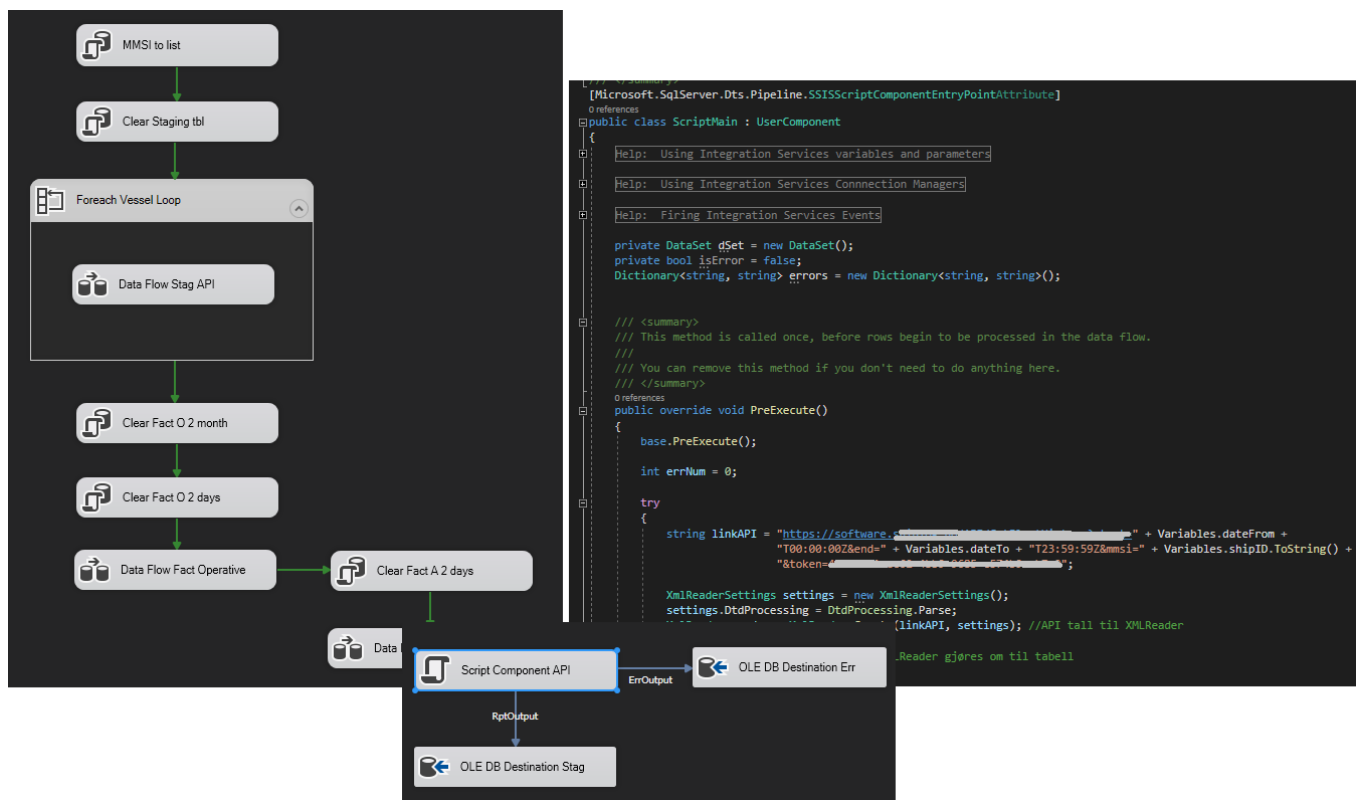
LEVEL OF COMPETENCE: DWH AND ETL TOOLS (SSIS, APACHE AIRFLOW)

What I know and what I can

- SSIS: advanced level inclusive scripting modules (C#) and migrating of packages to Azure Data Factory
- Apache Airflow – basic experience with DAG scripting
- Sound understanding of DWH theory and dimensional modeling techniques (Kimball)

LEVEL OF COMPETENCE: PYTHON

What I know and what I can

- DAG scripting
- All the basics related to variables, lists, dictionaries etc.
- Functions and classes
- Files and exceptions
- I/O, loops, conditions, and basic programming concepts

```python
import requests
import json
import os
from file_directories import file_path
import sys
import yaml


#download config. fil / загрузка конфигурационного файла
def load_config(config_path):
    with open(config_path, mode="r") as yaml_file:
        config = yaml.safe_load(yaml_file)
        return config



def app(config):

    dates = ["2021-01-03", "2021-01-04", "2021-01-05", "2021-01-06"]
    create_dir = file_path()
    con = config["app"]

    #download API token / загрузка API-токена
    try:
        url = con["url_auth"]
        headers = {"content-type": con["content_type"]}
        data = {"username": con["username"], "password": con["password"]}
        r = requests.post(url, data=json.dumps(data), headers=headers)
        token = f"JWT {r.json()['access_token']}"
    except Exception:
```

LEVEL OF COMPETENCE: STATISTICS

What I know and what I can

- Basic concepts
- Averages, means and medians
- Central limit theorem, distributions, deviations, confidence intervals etc.
- Some advanced topics

LEVEL OF COMPETENCE: POWER BI AND DAX

What I know and what I can

- Power BI desktop, Power BI Service inclusive Gateway and Power BI API
- Integration of Power Apps into Power BI reports
- All the basic DAX operators and concepts, DAX is used mainly to make measurements in Power BI

LEVEL OF COMPETENCE: RPA

What I know and what I can

Blue Prism, UiPath, certified developer and solution architect

LEVEL OF COMPETENCE: VBA

What I know and what I can

- Excel + VBA, advanced level
- Access + VBA, advanced level

LEVEL OF COMPETENCE: HTML, CSS

What I know and what I can

All the basics that allow:
- to develop a web page with all standard UI elements
- to format web pages as necessary

```html
@{
    ViewData["Title"] = "Edit";
}

<br />
<form class="form-inline">
    <br />
    <div class="form-group">
        <h3 style="margin-left: 7px;"> @ViewBag.OperNvn </h3>
        <h3>
            <label style="margin-left: 13px" ; class="badge badge-secondary">@Model.OperatorID</label>
        </h3>
    </div>

</form>
<br />

<form class="form-inline">
    <div class="form-group">
        <h6 class="text-muted" style="margin-left: 8px;">Skift nummer:</h6>
        <h6 style="margin-left: 5px; color:#B03E51">@Model.SkiftNr</h6>
        <h6 class="text-muted" style="margin-left: 25px;">Skift dato:</h6>
        <h6 style="margin-left: 7px; color:#B03E51">@Model.Skift_Dato.ToString("dd.MM.yyyy")</h6>
        <h6 class="text-muted" style="margin-left: 25px;">Skift total NOK:</h6>
        <h6 style="margin-left: 7px; color:#B03E51">@Model.Skift_Total</h6>
        <h6 class="text-muted" style="margin-left: 25px;">Voucher:</h6>
        <h6 style="margin-left: 7px; color: #B03E51">@Model.Voucher</h6>
        <h6 class="text-muted" style="margin-left: 25px;">Bilagsnummer:</h6>
        <h6 style="margin-left: 7px; color: #B03E51">@Model.BilagNr</h6>
        <h6 class="text-muted" style="margin-left: 25px;">Registrert av:</h6>
        <h6 style="margin-left: 7px; color: #B03E51">@Model.registrertav</h6>
    </div>
</form>

<br />

@if (Model.Skift_Eksportert == 1)
{
    <div class="form-row">
        <div class="form-group">
            <smal class="form-text text-muted" style="margin-left: 13px;">Fartøy</smal>
            <select class="custom-select my-1 mr-sm-2" id="ShipName" disabled style="width: 300px; height: 40px; margin-left: 14px"><option selected>@Model.SkipNvn</option></select>
        </div>
        <div class="form-group">
            <smal class="form-text text-muted" style="margin-left: 7px;">Samband</smal>
            <select class="custom-select my-1 mr-sm-2" id="SambName" disabled style="width: 412px; height: 40px; margin-left: 7px"><option selected>@Model.Samband</option></select>
        </div>
    </div>
    <br />
```

LEVEL OF COMPETENCE: JAVASCRIPT

What I know and what I can

- Work with and iterate through UI elements on a web page
- Real-time content control and value extraction
- All the general basics

```javascript
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>

<script>

    //Dropdown avstem. total
    $(function () {
        $('#AvstDt').change(function () {
            var dt = $('#AvstDt').val();
            //alert(dt);
            $.getJSON('/Home/BAXAmountList/?dt=' + dt,
                function (data) {

                    var $states = $("#AvstTot");
                    var $statesB = $("#AvstBAX");
                    var $statesA = $("#AvstNr");

                    $states.empty();
                    $statesB.empty();
                    $statesA.empty();

                    var skiftTot = $('#SkiftTot').val()
                    var korTot = $('#inputKor').val()


                    var selValg = $("#AvstDt option:selected").text()

                    if (selValg != "Velg dato--") {
                        $states.append($("<option>").text("Velg beløp--").val("Velg beløp--"));
                    }

                    document.getElementById("betFNS").innerHTML = (Number(skiftTot.replace(",", ".")) + Number(korTot.replace(",", "."))).toFixed(2);


                    $.each(data, function (i, BAXAmounts) {
                        //alert("Hello: " + BAXNumbers.Text + "   " + BAXNumbers.Value);
                        $states.append($("<option>").text(BAXAmounts.Text).val(BAXAmounts.Value));
                    });
                });
        });
    });
```