



COMMON EUROPE CONGRESS 2026

14 - 17 June
Lyon, France

The largest conference in Europe
for solutions around IBM Power (IBM i, AIX, Linux) & IBM Storage



www.comeur.org



LYON EVENTS | CENTRE DE CONGRÈS DE LYON

Serverless architecture with RPG

Niels Liisberg

Chief architect

Sitemule - System & Method A/S

Copenhagen

Denmark

CEC2026 Lyon





Who is Niels Liisberg?

- Chief architect - System & Method A/S – Denmark
- IBM Power evangelist
- “The doctor” – modernization consultant
- Father to **IceBreak**™ – Application server for ILE / RPG
- Architect of **Sitemule**™ – Web and microservice platform for IBM i
- CEAC – Common Europe Advisory Council member
- Devoted to open source:

- My stuff <https://github.com/NielsLiisberg>
- noxDB <https://github.com/sitemule/noxdb>
- ILEastic <https://github.com/sitemule/ILEastic>
- ILEvator <https://github.com/sitemule/ILEvator>
- Gist <https://gist.github.com/NielsLiisberg>
- VSCode [RPG/SQL/Debugger](https://github.com/NielsLiisberg/RPG/SQL/Debugger)

- Founder of **IBM Power CPH**
- IBM Champion 2019 - 2025

Follow me at X, tweeter, github, linkedin , gluf, clap, clup or what eve





System & Method A/S

- System & Method was established in 1989
- IBM Advanced Business Partner
- Sales directly to customers
- Sales via partners
- System development and implementation
- More than 1000 customers in 28 countries
- Subsidiaries in Denmark and Brazil
- Partners in more than 20 countries





Agenda

- Serverless architecture – What is it?
- What is “normal”
- Serverless architecture
 - In general
 - In RPG
- Open source
- Example time:
 - Build ILElastic
 - Hello world
 - Static files - service
 - Microservices





A blue thought bubble graphic containing the text "Serverless Architecture???" and a yellow thinking face emoji (🤔). The bubble has several smaller blue circles leading to it from the right side.

Serverless Architecture???





What is this?

JS hello.js ×

Users > nli > JS hello.js > ...

```
1  const express = require('express')
2  const app = express()
3  const port = 3000
4
5  app.get('/', (req, res) => {
6    res.send('Hello World!')
7  })
8
9  app.listen(port, () => {
10   console.log(`Example app listening on port ${port}`)
11 })
12
```

hello.py ×

Users > nli > hello.py

```
1  from fastapi import FastAPI
2
3  app = FastAPI()
4
5  @app.get("/")
6  async def root():
7      return {"message": "Hello World"}
8
```



Serverless architecture

- *An approach to software design that allows developers to build and run services without having to manage the underlying infrastructure*
- Aka – as we know them:
 - No Apache
 - No Tomcat, WebSphere
 - No Liberty, IWS
- So the http stack need to be in your code:
 - Embedded server
 - Using a framework (probably)





Why is this important?

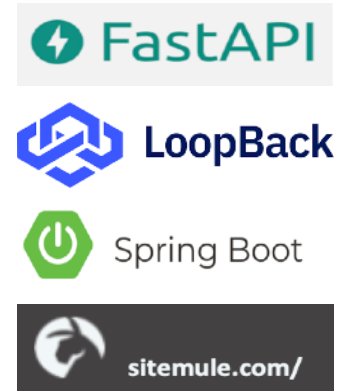
- No external dependencies in Develop, Test and Deploy





Serverless architecture - examples

Language	Framework	Embedded server
Python	FastAPI	Uvicorn
Node.js	LoopBack	Express
Java	Spring Boot	Tomcat/Undertow
RPG	noxDb	ILElastic



Also: Seneca, Flask, micronaut - to namedrop a few



“Normal” Serverless architecture

- Develop anywhere:
 - Win/Mac/Linux
 - VSCode (or what you like)
 - Using git presumably
- Deploy everywhere:
 - Cloud
 - On prem
 - Hybrid
 - Using container presumably





I am not strange!
I am just NOT normal

Salvador Dalí



What is “normal”

- IBM i can do everything .. But different (Often better !!)
- PASE
 - Brings AIX to the platform
 - yum, Node.js, Python, bash , git ... “Normal stuff”
- ILE
 - Native to the OS
 - EBCDIC
 - RPG, COBOL, CL (C, C++)
 - Different toolset: Compilers, editors ...
 - ... “Not so normal .. Today”





IBM i can do everything .. But different

- Springboot, LoopBack, FastAPI:
 - Is “cloud technology” but runs perfectly on IBM i
 - Also runs inside containers
 - It’s open source
- ILEastic
 - Runs only on IBM i
 - Wraps legacy code and databases
 - Implements Microservices in RPG and ILE
 - No entanglement
 - It’s open source





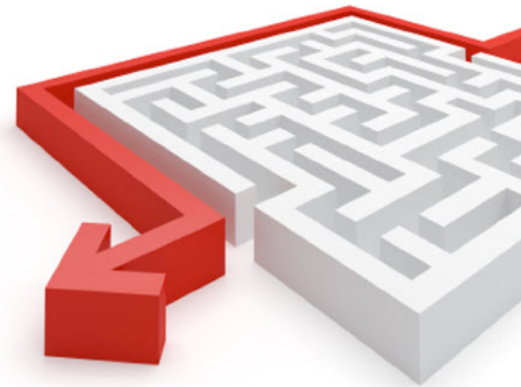
Pick the right tool for the task

- Springboot, LoopBack, FastAPI:
 - New projects
 - Onboarding staff
- ILElastic
 - Transformation
 - Performance / mission critical
 - Simplicity
 - The “Rosetta stone”
 - Legacy code modernization





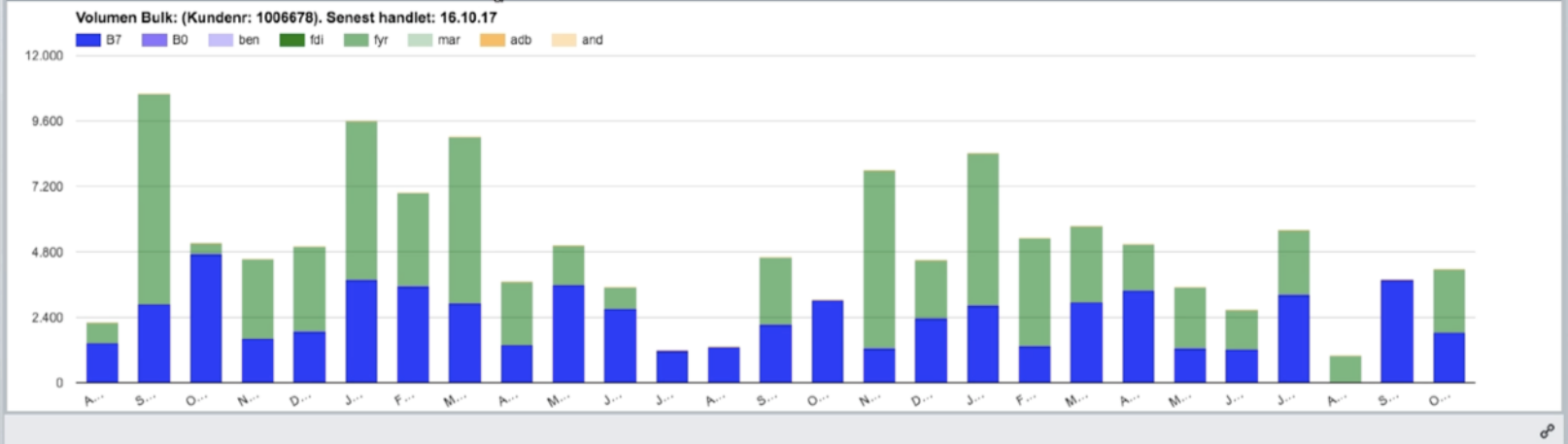
Why?



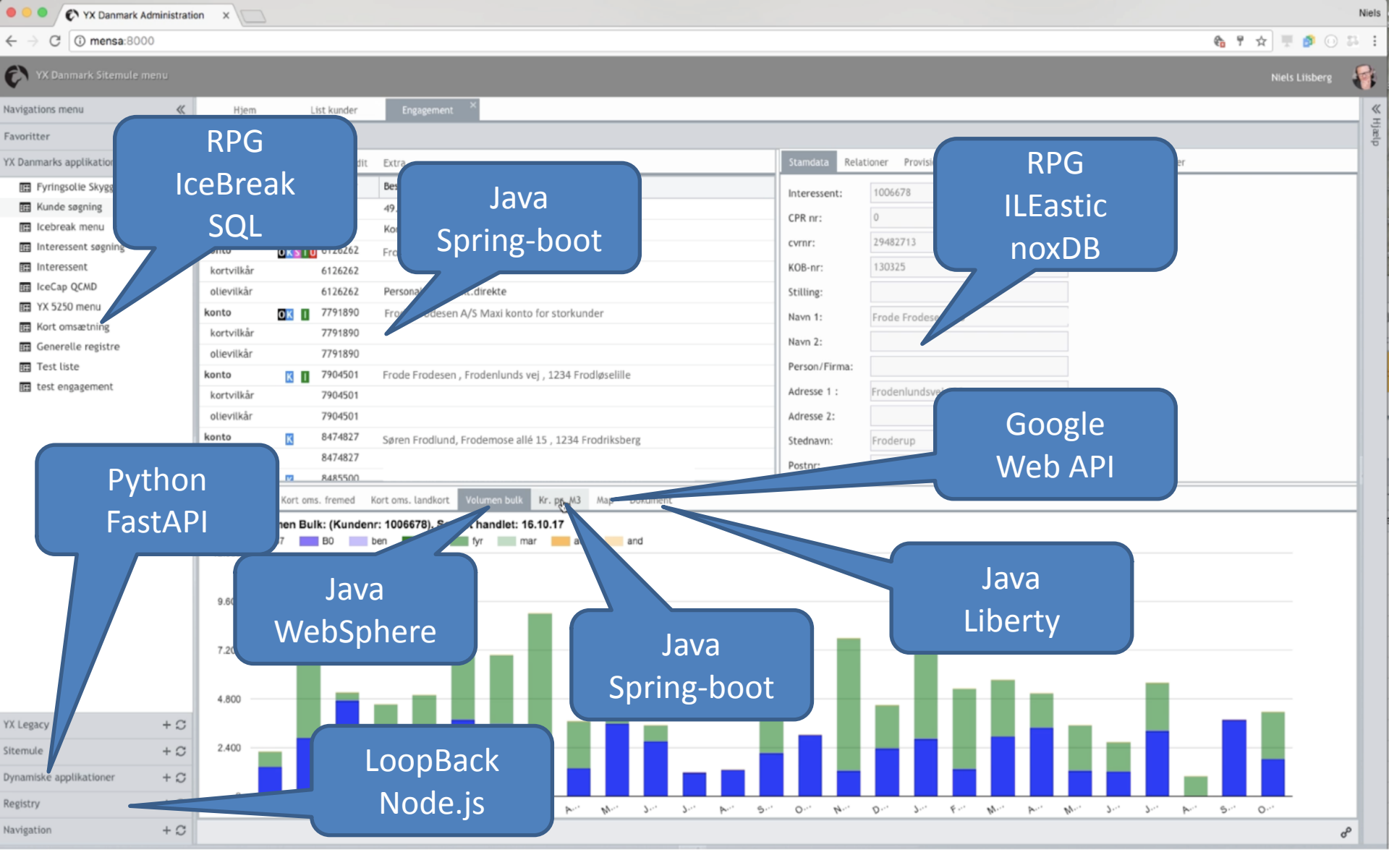
- Fyringsolie Skygge
- Kunde søgning
- Icebreak menu
- Interessent søgning
- Interessent
- IceCap QCMD
- YX 5250 menu
- Kort omsætning
- Generelle registre
- Test liste
- test engagement

Engagement	Type	Konto nr	Beskrivelse
firma		0	49.41.00 Vejgodstransport
kunde		0	Kort Stor / Kort DS / Store >250000
konto	OK S I U	6126262	Frode Frodesen A/S
kortvilkår		6126262	
otievilkår		6126262	Personale skal fakt.direkte
konto	OK I	7791890	Frode Frodesen A/S Maxi konto for storkunder
kortvilkår		7791890	
otievilkår		7791890	
konto	K I	7904501	Frode Frodesen, Frodenlunds vej, 1234 Frodløselille
kortvilkår		7904501	
otievilkår		7904501	
konto	K	8474827	Søren Frodlund, Frodemose allé 15, 1234 Frodrisberg
kortvilkår		8474827	
konto	K	8485500	

Stamdata	Relationer	Provision	Rabat ordninger	Bidrag	Grupper	Dokumenter
Interessent:	1006678					
CPR nr:	0					
cvnrn:	29482713					
KOB-nr:	130325					
Stilling:						
Navn 1:	Frode Frodesen A/S					
Navn 2:						
Person/Firma:						
Adresse 1 :	Frodenlundsvej 123					
Adresse 2:						
Stednavn:	Froderup					
Postnr:	1234					



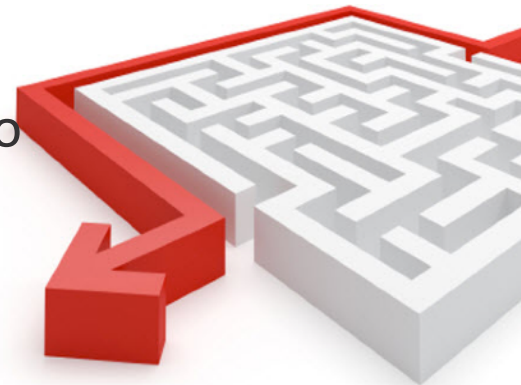
- YX Legacy +
- Sitemule +
- Dynamiske applikationer +
- Registry +
- Navigation +





Why serverless architecture?

- A cornerstone in microservices
- Integration on the glass
- Heterogenous development
- Use the language and tool that fits the solution best
 - ... but a similar approach
 - The “Rosetta stone”:
 - Young and mature staff work with same lingo
 - JSON / HTTP
 - The coding language becomes secondary



Navigation

Mine Favoritter

Indtast for at søge

Brugere

303-2 Smøreliefakturering

Legacy Applikationer / AS400

Søg

- 110 Forespørgsel/Kartoteksvedligehold
- 120 Kundecentret - Daglige
- 121 Kundecentret - Specielle
- 126 Fyringsolie
- 130 Kreditafdelingen
- 131 Kreditafdelingen - diverse kredittal
- 132 Kreditvurdering
- 140 Nettofakturering
- 150 RKI AKS open
- 160 Flytteoplysninger fra Post Danmark
- 170 DCC afregning
- 190 Debitor specialmenu
- 191 Lister debitor/kreditor
- 260 Fastpriser
- 265 Dagens fastpris - Nettopriser
- 266 Dagens fastpris
- 301 Ordremodtagelse & Forespørgsel
- 302 Kartoteksvedligeholdelse
- 303 Fakturering
- 304 Udskrifter
- 310 Volumenregnskabssystem
- 311 Pøjlinger
- 315 Kundecenter

1. Kundesøgning 2. Test konto

Afvig	Engage...	Type	Konto nr	Beskrivelse	Division	Bet. ...	Sen...	Bet. Betingel...	R.	Saldo	Forfal...	Ej fak...	Retail sald...
		person	0	Personoplysninger						0	0,00	0,00	0,00
		kunde	0		Fyringsolie					0	0,00	0,00	0,00
		konto	9431654	Test konto	Fyringsolie					0	0,00	0,00	0,00
		olievilk...	9431654			DK		LEV.+10 DG.		0	0,00	0,00	0,00

Stamdata Orlentanke Kortoversigt Abonnementer Ordreoversigt Relationer Kontaktpersoner Provision

GDPR: Er GDPR Kunde

Division: Fyringsolie

Interesent-type: Person

AKS/RKI: Ikke tilmeldt

Oprettelsesdato: 18.12.2025

Info:

Landekode: DK

Telefon nr.:

Mobil nr.:

E-mail:

Arbejd med dosmersedler

KUN850RP NLI **Arbejd med dosmersedler** 12.06.26 15:32:28

Bruger NLI F4 Niels Liisberg

Inkluder med

Kundenr. . . . 943165-4 Test Firma

Marked (E=Erhverv, F=Fyringsolie, L=Landbrug, R=Retail, Blank=Alle)

Indtast valg, tryk på Enter.

55=Tank Adr. 60=Tilmeld PBS 61=Afmeld PBS 62=PBS oplysninger

87=Overfør til salgssted 0 96=Kundelog 97=Sagsbehandleroplysninger

Valg	Dato	Kundenr.	Huskeseddel
<input type="checkbox"/>	12.06.2026	943165-4	NLI : Test

Bottom

F1-Hjælp F3-Afslut F4-Liste F5-Genopfrisk F6-Opret F8-Vis mine/konto F23-Flere valg F24-Flere taster

Dosmersedlen er oprettet.

11/3 da



Serverless architecture in RPG and Node.js



hello.rpgle

Users > nli > hello.rpgle

```
1  **free
2  ctl-opt main(main) debug(*yes) bndDir('ILEASTIC') thread(*concurrent);
3  /include include/ileastic.rpgle
4  dcl-proc main;
5      dcl-ds config likeds(IL_CONFIG);
6      config.port = 3000;
7      il_listen (config : %paddr(sayHello));
8  end-proc;
9  dcl-proc sayHello;
10     dcl-pi *n;
11         request likeds(IL_REQUEST);
12         response likeds(IL_RESPONSE);
13     end-pi;
14     il_responseWrite(response:'Hello world');
15 end-proc;
```

JS hello.js

Users > nli > JS hello.js > ...

```
1  const express = require('express')
2  const app = express()
3  const port = 3000
4
5  app.get('/', (req, res) => {
6      res.send('Hello World!')
7  })
8
9  app.listen(port, () => {
10     console.log(`Example app listening on port ${port}`)
11 })
12
```



Serverless architecture in RPG and Python



hello.rpgle ×

Users > nli > hello.rpgle

```
1  **free
2  ctl-opt main(main) debug(*yes) bndDir('ILEASTIC') thread(*concurrent);
3  /include include/ileastic.rpgle
4  dcl-proc main;
5      dcl-ds config likeds(IL_CONFIG);
6      config.port = 3000;
7      il_listen (config : %paddr(sayHello));
8  end-proc;
9  dcl-proc sayHello;
10     dcl-pi *n;
11         request likeds(IL_REQUEST);
12         response likeds(IL_RESPONSE);
13     end-pi;
14     il_responseWrite(response:'Hello world');
15 end-proc;
```

hello.py ×

Users > nli > hello.py

```
1  from fastapi import FastAPI
2
3  app = FastAPI()
4
5  @app.get("/")
6  async def root():
7      return {"message": "Hello World"}
8
```



ILElastic is

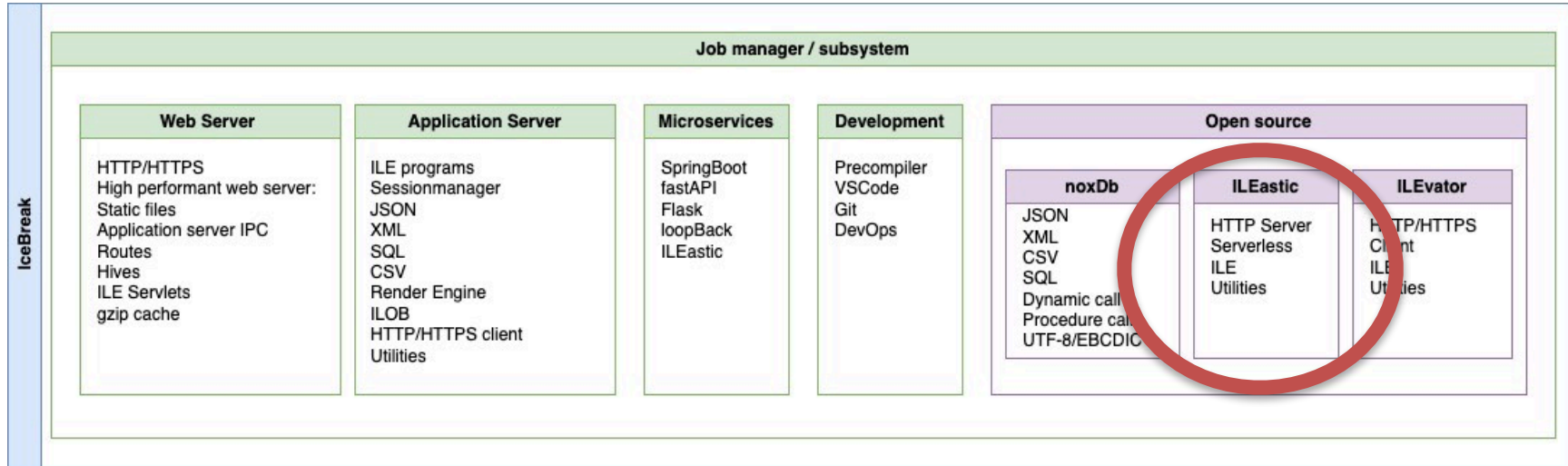
- Native serverless framework for ILE *
- noxDB – json/xml/csv and more - object graph for ILE *
- Open source
- Same development pattern as:
 - Spring-boot, FastAPI, LoopBack
- Build system (by gmake)
- Git native (clone from git, build using IFS)
- Super fast
- Small footprint
- Multithreaded
- Easy integration with CICD / DevOps
- Works with service-discoveries/ Gateways / load balancing
 - (Eureka, Kong)

*RPG, COBOL, C, C++, CLLE





The background



- ILElastic is carved out of IceBreak as an open source project
- ILElastic is community driven
- ILElastic is build, tested and delivered in the IceBreak product
- System & Method A/S provides support



It is showtime !!



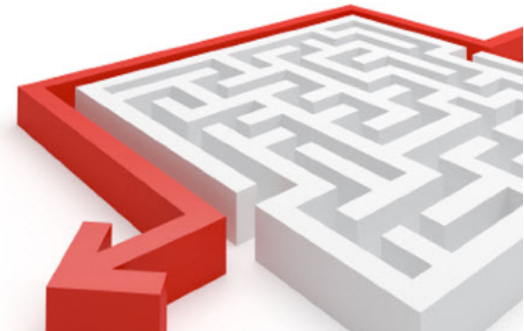
Github:

<https://github.com/sitemule/ILElastic>

- RPG:
 - ILElastic : RPG framework for embedded HTTP (opensource)
- noxDB
 - SQL, JSON, XML, CSV and dynamic procedure/program framework
- VSCode
 - Microsoft open source development tool
 - Sitemule RPG extension (16.100 + users !!)
- OSS for IBM i
 - yum/DNF , gmake



Demo time





Serverless architecture in RPG and Python



```

ctl-opt debug(*yes) bndDir('ILEASTIC':'NOXDB');
ctl-opt thread(*CONCURRENT);
ctl-opt main(main);
#include ./headers/ILElastic.rpgle
#include ./noxdb/headers/jsonParser.rpgle
//
// Program Entry Point
//
dcl-proc main;
  dcl-ds config likeds(IL_CONFIG);

  config.port = 44001;
  config.host = '*ANY';

  il_addRoute(config : %paddr(listServices) : IL_ANY : 'listServices');
  il_listen(config);
end-proc;
//
// Servlet callback implementation
//
dcl-proc listServices;
  dcl-pi *n;
    request likeds(IL_REQUEST);
    response likeds(IL_RESPONSE);
  end-pi;
  dcl-s pResult pointer;
  dcl-s search varchar(64);

  // Assume everything is OK
  response.status = 200;
  response.contentType = 'application/json';
  search = il_getParmStr(request : 'search');

  // Use noxDB to produce a JSON resultset to return
  pResult = json_sqlResultSet ('-
select * -
from qsys2.services_info -
where service_name like upper(' + strQuot('%' + search + '%') + ')
');

  // Use the stream to input data from noxDB and output it to ILElastic
  il_responseWriteStream(response : json_stream( pResult));
end-proc;

```

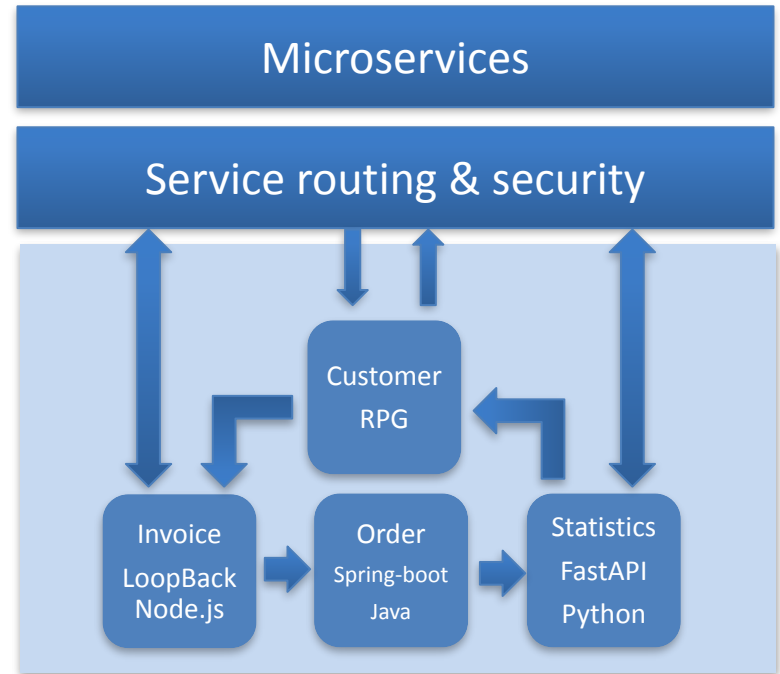
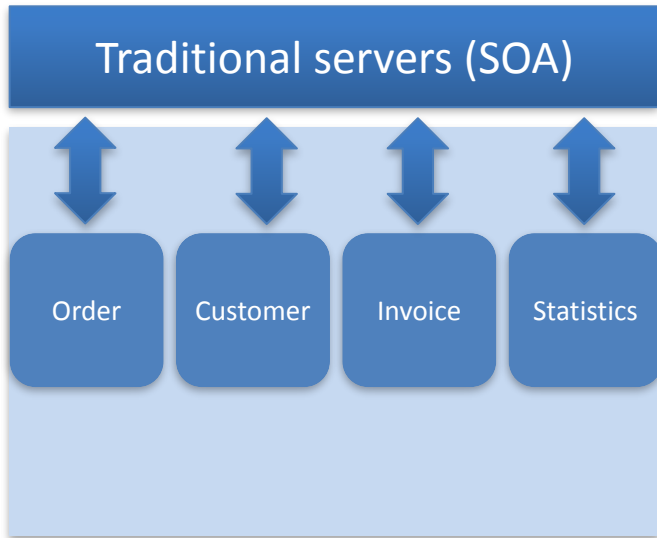
```

1 import pyodbc
2 from fastapi import FastAPI, Query
3 from typing import Union
4
5 connection = pyodbc.connect("DSN=*LOCAL")
6 app = FastAPI()
7
8 # Our primary endpoint:
9 # Usecase:
10 # http://my_ibm_i:60300/listServices?search=ptf
11 # http://my_ibm_i:60300/docs
12 @app.get("/listServices")
13 def listServices(search: Union[str, None] = Query(default=None, max_length=50)):
14     sql = f'select * from qsys2.services_info where service_name like {string_quote("}'
15     return fetch_all (connection , sql)
16
17 # Utility:
18 # Returns a result-set produced by a SQL statement
19 def fetch_all(connection , sql):
20     c1 = connection.cursor()
21     c1.execute(sql)
22     ret = result_to_dict(c1)
23     c1.close()
24     return ret
25

```

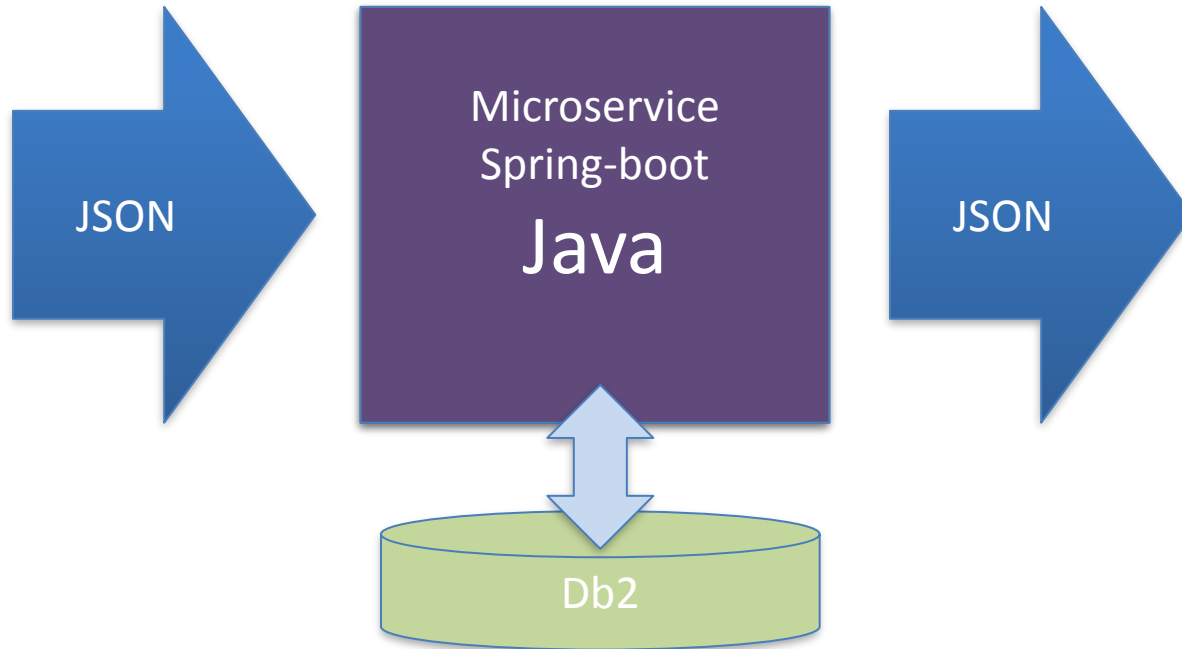


How does it work in the real world?





Transformation example:



EXPLORER

OPEN EDITORS

noxdbcust.rpgle examples M

ILEASTIC

.vscode

c_cpp_properties.json

settings.json

tasks.json M

examples

base64.rpgle

datachunks.sqlrpgle

errorlist.txt

Fastcgi01.rpgle

header.rpgle

helloworld.rpgle

index.html

invalidreq.rpgle

jsondata.sqlrpgle

jsonp.rpgle

jwtsecrout.rpgle

makefile M

multroutes.rpgle

noxdbcust.rpgle M

plugin.rpgle

querystr.rpgle

scheduler.rpgle

staticfile.rpgle

headers

ILEfastCGI S

noxDB S

plugins

src

unittests

.gitignore

.gitmodules

barryci.json

LICENSE

OUTLINE

SSH FILE SYSTEMS

MAVEN PROJECTS

noxdbcust.rpgle ×

examples > noxdbcust.rpgle

```

20  ctl-opt debug(*yes) bndDir('ILEASTIC':'NOXDB');
21  ctl-opt thread(*CONCURRENT);
22  ctl-opt main(main);
23
24  /include ./headers/ILElastic.rpgle
25  /include ./noxdb/headers/jsonParser.rpgle
26
27  // -----
28  // Program Entry Point
29  // -----
30  dcl-proc main;
31
32      dcl-ds config likeds(IL_CONFIG);
33      config.port = 44001;
34      config.host = '*ANY';
35
36      il_listen (config : %paddr(customerList));
37
38  end-proc;
39
40  // -----
41  // Servlet callback implementation
42  // -----
43  dcl-proc customerList;
44
45      dcl-pi *n;
46          request likeds(IL_REQUEST);
47          response likeds(IL_RESPONSE);
48      end-pi;
49
50      dcl-s pResult pointer;
51
52      // Assume everything is OK
53      response.status = 200;
54      response.contentType = 'application/json';
55
56      // Use noxDB to produce a JSON resultset to return
57      pResult = json_sqlResultSet ('-
58          select *           -
59          FROM QIWS/QCUSTCDT -
60      ');
61
62      // Use the stream to input data from noxDB and output it to ILElastic
63      il_responseWriteStream(response : json_stream( pResult));
64
65
66  end-proc;

```




Case Stories



DFD

HT BENDIX^{A/S}





The screenshot displays the HT Bendix software interface. On the left, a navigation sidebar shows a tree view of projects, with 'INQ-1457' selected. The main area is divided into two panes. The top pane shows a summary for 'Forespørgsel - INQ-1457', including a 'TILBAGE' button and a progress bar indicating 1 forecast, 2 products, 3 goods, and 4 suppliers. The bottom pane shows a detailed view of the 'Fildisem-anlæg' project, including a list of items with columns for 'Vare', 'Lst', 'Val', 'Søkv', 'start', 'K', 'Pris', and 'varetekst'. A context menu is open over the 'INQ-1418' item, showing options like 'Upload', 'Leverandørbilag', 'Opret vare', 'Rediger vare', 'Rediger prislst', 'Opdater Astra', 'Valgmuligheder', and 'Send Email'. The bottom right corner of the interface shows a keyboard shortcut menu with options like 'F3=Slut', 'F4=Liste', 'F6=Opret', etc.

HT Bendix

- RPG Microservices with ExtJS and Angular II
- Use of open source projects for services and tooling
- Website build with webservices written in RPG
- Transformation from 5250 to ExtJS

<https://www.ht-bendix.dk/>

<https://www.ibm.com/it-infrastructure/us-en/resources/power/ibm-i-customer-stories/>



HT BENDIX A/S

Navigation: Projekter

Favoritter: Brugere, Menu

Projekter: Create inquiry, Projekter, Mine projekter, Projekter pr. kunde, Projekter pr. ansvarlig, Projekter pr. status, Projects by salesperson, Projekt E-mails, Ubehandlede E-mails, Jobs, Jobs pr. delegeret til, Jobs pr. stadi, Jobs pr. ansvarlig, Specifikationer, Opsætning, Statistik, Logs

Projekter: 00. Ikke startet, 20. Opret tilbuds materiale, 40. Afventer leverandertilbud, 60. Opret kundertilbud

Projekter: INQ-1467, INQ-1457, INQ-1450, INQ-1449, INQ-1448, INQ-1445, INQ-1444, INQ-1441, INQ-1440, INQ-1433, INQ-1432, INQ-1427, INQ-1424, INQ-1419, INQ-1418

Forespørgsel - INQ-1457

TILBAGE: INQ-1457, 1 Forespørgsel, 2 Produkter, 3 Godkendelse, 4 Leverander

013-LBA-LBQ__Fit-glidesem-anlæg - Filtr - Fitdup rund

Varegruppe: 013 | Filtr-glidesem-anlæg

UPLOAD: images, images

bestovelse: Fitdup rund Ø20 3,5 4 08, filtr indfarvet sort, Selvkøbende

Pakkeinstruktion: 4 stk ark bulk i æsker

Kunde reference: Valgt en kunde reference

Mere info: Varenummer, Varebeskrivelse, Specifikationer

Tilbudsliste - INQ-1457

Linje, Leve..., Kund..., Varebeskrivelse, Pris, Pris pr. e...

0.0924 EUR 1.120...

Søren Caben Møller

Rediger vare, Rediger prislist, Opdater Astra, Valgmuligheder, Send Email

Arbejd med Listepris

HT BENDIX A/S 15/05/20 13:16:47

Kunde: 16265 SANVEI SCANDINAVIA ApS

Vare: Lst Val Sekv Ikraft dato, POS DKK

Tast ønske, tryk Enter.

1=Valg 2=Bevilder 4=Stet SvVis 8=Arbejd med Priser på enheder 9=Provision 1=Vareinf. kvantumrabat Røket pris med N S=Stet Prisk Ud P=Prisk Ud T=Int.Txt

V Vare Lst Val sekv start K Pris varetækt Enh t dato

5250 Session

80,41 hy7da 8E400x20 1,0 stå STK
70,30 hy7da 8E315x20 1,0 stå STK
82,59 hy7da 8E420x20 1,0 stå STK
100,88 hy7da 8E4500x20 1,0 stå STK
100,00 hy7da 8E4400x20 1,0 stå STK
100,29 hy7da 10E835E20 1,0 stå STK
116,08 hy7da 10E8420x20 1,0 stå STK

F3=Slut F4=Liste F6=Opret F10=Skift valuta F13=Alle/Akt F15=Ret F16=Renumerer
F17=Kopier F20=Prisgrupper F21=Udskriv Tilbud F22=Kmd F23=Udskriv Tilbud

IQ COPYRIGHT Steenbath & Hertum AS 1995

Valuta Sags... Tilby... ACCE... Vare... Astra Vare Skole... 4/7 WEB

Log & beskeder



Any language?

Of course!! You can make microservices with almost any languages:

- RPG [ILElastic and noxDB](#)
- RPG [IceBreak](#)
- JAVA [spring boot framework](#)
- Python [fastAPI](#)
- Node.js [LoopBack](#)

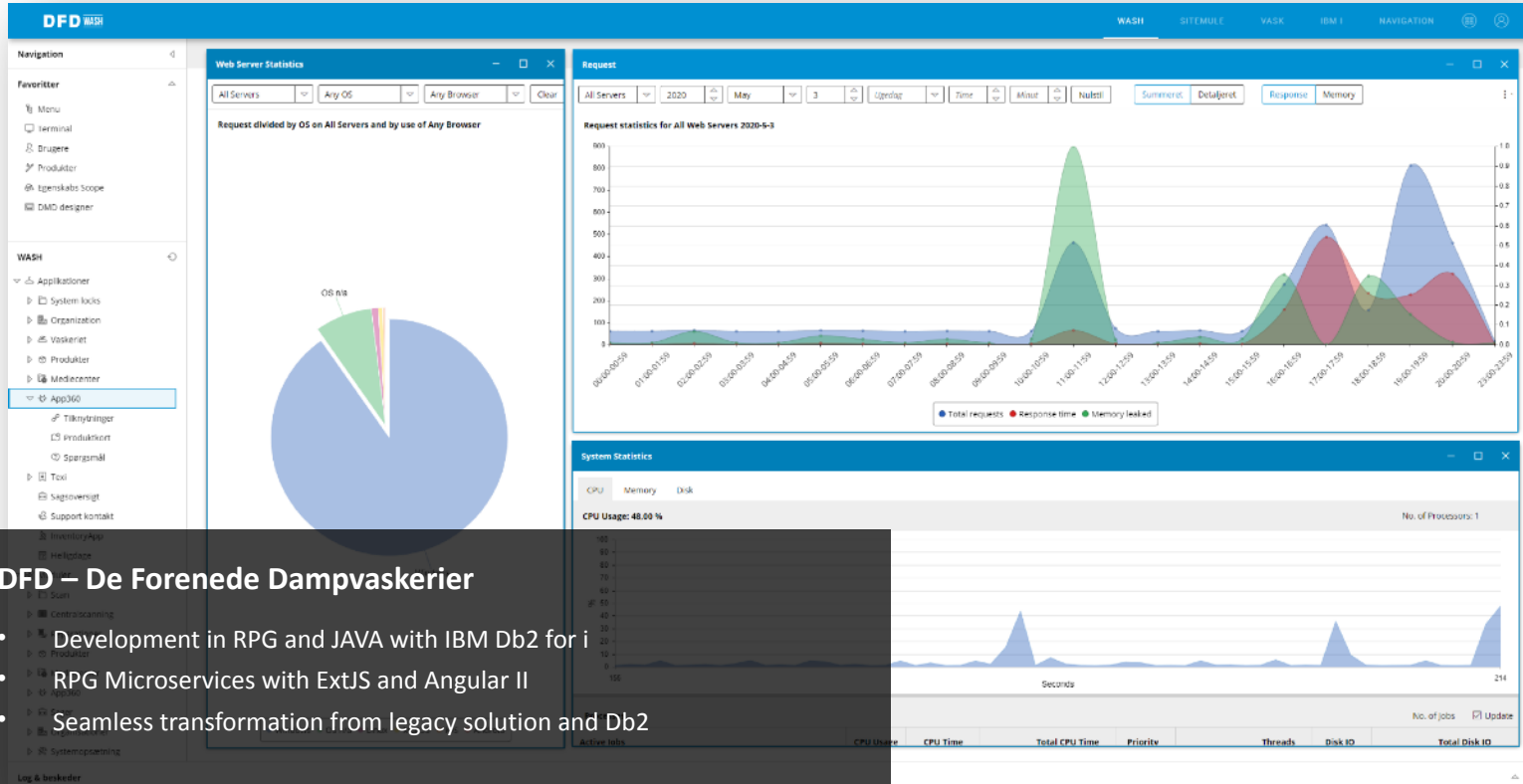
- Glue it all together with
 - [Eureka](#) and/or [Nginx](#)





Berry | Superfos

- New Applications with Microservices in JAVA
- Modifications with Microservices in RPG
- Globalization with UTF8 in Db2 and Intercompany Microservices for 14 countries in 8 partitions
- Transition from 5250 to graphical interface in ExtJS.
- Faster deployment at new factories



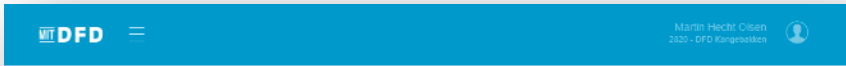
DFD – De Forenede Dampvaskerier

- Development in RPG and JAVA with IBM Db2 for i
- RPG Microservices with ExtJS and Angular II
- Seamless transformation from legacy solution and Db2



DFD – De Forenede Dampvaskerier

- Customer portal developed with RPG Microservices and Angular II



BEHOJNING Opred beholdning
 BEHOJNING Ændre beholdning
 SCAN Scan indlæs og reparation



BEKLÆDNING Åbn ordre
 BEKLÆDNING Opret ordre
 BEKLÆDNING Red og reparation



BEKLÆDNING Åbn ordre
 LINNED Opret ordre
 LINNED Rediger ordre



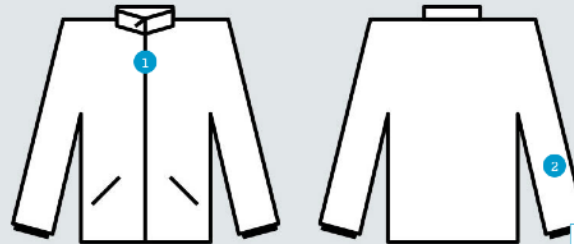
Supply & Service Opret ordre
 Supply & Service Rediger ordre
 RAPPORT Relevante rapporter

DFD De Forenede Dampvaskerier A/S - Lyngby



Reparationskort

Køkket prest singel k/æ hv



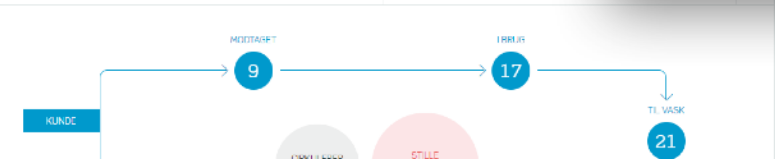
- 2. Plot
Bemærkning
- 1. Lymåle
Lymåle i stykker

- Knap
- Søm syes
- Hul
- Lomme
- Knaphul
- Plot
- Lymåle
- Selo
- Andet

ANNULLER

GEM & LUK

Vælg Navn, stilling, afdeling, lønnr, skab...





VÆRDI360 Opdater Vælg ram Katalog Kurv Katalog

- Rengøringsrum
- Personale rum
- Gartner
- Lastbil
- Hotelværelse
- Badeværelse
- Linned depot
- Køkken
- Restaurant**
- Bord
- Tjener
- Reception

Nederdel Essentia

Skjorte Essentia Dame kortærmet

Skjorte Essentia Herre langærmet

Vest Essentia

Product
F01 test emne - 1 LUK X

Skjorte Essentia Dame kortærmet

Et godt førstehåndstryk er essentielt og med til at understrege virksomhedens professionalisme, image og værdier. Med et sikkert valg fra SIGNATURE kollektionen, bliver netop professionalisme fuldent kombineret med funktionalisme i et tidsløst moderne design, uden at gå på kompromis med komfort og kvalitet.

Design

Køn Dame

Ærmelængde kortærmet

Brands

DFD varemærke Essentia

Komposition

Bomuld 62%

Polyester 35%

Flastan 3%

Kvalitet

Gram per kvadratmeter 130g/m2

LÆG I KURV

Skjorte - Slørreløsskjorta

Størrelse

32 34 36 38 40 42 44 46 48 50 52 54 SPEC

DFD – De Forenede Dampvaskerier

- Sales & Presentation App developed with RPG Microservices in Angular II



Sitemule menu Niels Lisberg

Navigation menu: << Hjem List kunder Engagement x

Favoritter: ..

YX Danmarks applikationer: Fyringsolie Skygge, Kunde søgning, Icebreak menu, Interessent søgning, Interessent, IceCap QCMD, YX 5250 menu, Kort omsætning, Generelle registre, Test liste, test engagement

Engagement	Type	Konto nr	Beskrivelse
firma	0	49.41.00	Vejgodstransport
kunde	0		Kort Stor / Kort DS / Store +25000
konto	OKSLU	6126262	Frode Frodesen A/S
kortvilkår		6126262	
olievilkår		6126262	Personale skal fakt. direkte
konto	DK I	7791890	Frode Frodesen A/S Maxi konto for storkunder
kortvilkår		7791890	
olievilkår		7791890	
konto	K I	7904501	Frode Frodesen , Frodenlunds vej , 1234 Frodiasselille
kortvilkår		7904501	
olievilkår		7904501	
konto	K	8474827	Søren Frodlund, Frodemose allé 15 , 1234 Frodricksberg
kortvilkår		8474827	
kontn	m	8485501	

Stamdata: Relationer Provision Rabat ordninger Bidrag Grupper Dokumenter

Interessent: 1006678
 CPR nr: 0
 cvmr: 29482713
 KOB-nr: 130325
 Stilling:
 Navn 1: Frode Frodesen A/S
 Navn 2:
 Person/Firma:
 Adresse 1: Frodenlundsvej 123
 Adresse 2:
 Stednavn: Froderup
 Postnr: 1234

Olie Ordreregistrering

FAK101RP NLI 7.02.19 09:16:57

3800 GRÅ FGX TANK kundenumer. : 612626-2 Disp:0000

Lev: Fa. Telefon.

Frode Saldo. 15548018,61

Roskildevej Forfald.saldo: S.lev.dto: 8.01.2019

2640 Hedehusene Rest saldo VK: 0,00 VK slut. : 0

Sted: Lev. ej fakt.: 0,00 FBF : 0

Næste G-tal. : 70081

Tekst . : Løbende måned + 30 dage **Fast pris. 0,00**

Produkt . FYX Tankplac. . F Tanktype . FR **Rabat kr . 1.350,00**

Kvantum . 1800 F Var.txt . . . **Alt.pris .**

Lev Fra . 25.01.2019 Til dato. 29.01.2019 Fra Kl. 0,00 **Til Kl. . 24,00**

Send PDF-faktura J/N . . . : **J faktura**

Tør løber. Graddag Kd: 8 20 Min. gebyr N Tankstr. : 3.800

Aut.opfy. NEJ Lev. gebyr N

Dist omr. 50 kontr.sdl. NEJ Strax faktura N Ordrenr. : 2501701

Fakt.txt. Overfor miga J Spec. betbet , 0 F4

Bemærk. 3 Miga system 5 Spec.ord.: N

1 STOP LEV. NÅR FØRSTE FLØJTE STOPPER (BEFRI 01.13)

2 2 NYE GRÅ TANK 161209 AF 1900 L - KODE 7755

Anmærk. . TØR VED 3185 LTR. 010318 14:25 VI KOMMER INDENFOR 2 TIMER

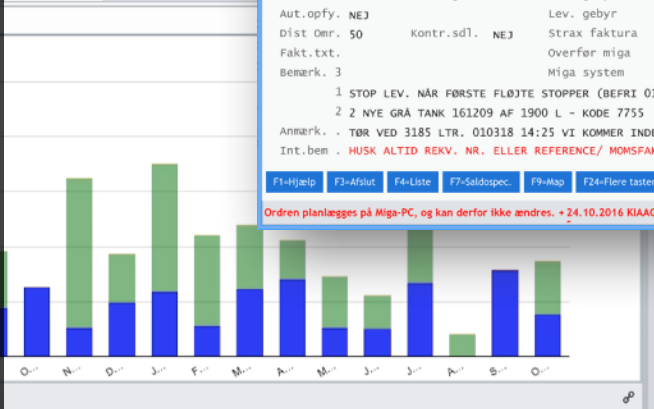
Int.bem . HUSK ALTID REKV. NR. ELLER REFERENCE/ MOMSFAKT

F1+Hjælp F3=Afslut F4=Liste F7=Saldospec. F9=Map F24=Flere talter

YX Denmark

- Integration on "The Glass":
 - In-house developed ERP and IBS Finance solution in 5250
 - Business Intelligence developed in-house
 - API phone interface
 - Roadmap from RPG to JAVA – From old to new generation
- Sitemule
- Dynamiske applikationer
- Registry
- Navigation

Microservices in JAVA and Stored Procedures in Db2
 DevOps in every part of the process



Ordren planlægges på Miga-PC, og kan derfor ikke ændres. +24.10.2016 KIAAG: FL ansatte skal oprettet på en særlig måde



RPG example



Github:

<https://github.com/sitemule/ILEastic>

- RPG:
 - ILEastic : RPG framework for embedded HTTP (opensource)
- noxDB
 - JSON, XML and CSV framework
- VSCode
 - Microsoft open source development tool
 - Sitemule RPG extension (11.000+ users !!)
 - gmake build system

Adopting Microservices at Netflix: Lessons for Architectural Design

microservices, containers, Netflix, content delivery network (CDN)



TWITTER



LINKEDIN



In some recent blog posts, we've explained why we believe it's crucial to adopt a [four-tier application architecture](#) in which applications are developed and deployed as sets of [microservices](#). It's becoming increasingly clear that if you keep using development processes and application architectures that worked just fine ten years ago, you simply can't move fast enough to capture and hold the interest of mobile users who can choose from an ever-growing number of apps.

Switching to a microservices architecture creates exciting opportunities in the marketplace for companies. For system architects and developers, it promises an unprecedented level of control and speed as they deliver innovative new web experiences to customers. But at such a breathless pace, it can feel like there's not a lot of room for error. In the real world, you can't stop developing and deploying your apps as you retool the processes for doing so. You know that your future success depends on transitioning to a microservices architecture, but how do you actually do it?

Fortunately for us, several early adopters of microservices are now generously sharing their expertise in the spirit of open source, not only in the form of published code but in conference presentations and blog posts. [Netflix](#) is a leading example. As the Director of Web Engineering and then Cloud Architect, Adrian Cockcroft oversaw the company's transition from a traditional development

NETFLIX

ABOUT NGINX

NGINX is the heart of the modern web, powering half of the world's busiest sites and applications. The company's comprehensive [application delivery platform](#) combines load balancing, content caching, web serving, security controls, and monitoring in one easy-to-use software package.

CATEGORIES

[Events](#)[News](#)[Tech](#)[Opinion](#)[More »](#)

TOP POSTS

[Creating NGINX Rewrite Rules](#)[Introduction to Microservices](#)[Building Microservices: Using an API Gateway](#)[Adopting Microservices at Netflix: Lessons for Architectural Design](#)[A Guide to Caching with NGINX and NGINX Plus](#)

STAY IN THE LOOP





Microservices and RPG?

Q: Can you make microservices in RPG?

A: Absolutley !!

Q: Microservices requires a self contained application server?

A: Look at opensource project ILElastic

Q: I need some JSON tooling if you will provide JSON I/O

A: Look a opensource project noxDB

Q: Can opens source help me?

A: Get started at : <https://github.com/sitemule/>



How we did it and are doing it

- Open Source:
 - Encapsulation of existing RPG code and application and as JSON I/O services (Sitemule - ILElastic/noxDB)
 - Create a test unit environment
 - Build new services in Spring-Boot JAVA (any modern language)
 - Run and compare RPG and JAVA service layer
 - Using “Eureka” as service discovery / balancing
 - Spreading the services over docker on power and IBM
 - Packages: yum, git, gmake, python, node, nginx



Sitemule

Sitemule is all kind of web solutions for Power and IBM

Denmark <http://sitemule.com> nl@sitemule.com



NETFLIX
EUREKA



<https://gitlab.devops.sitemule.com/demos/i-ug-uk>

<https://github.com/sitemule>



Serverless architecture and Microservices

Questions?

By:

Niels Liisberg

nli@system-method.com

System & Method A/S

