

Qlik Beyond Dashboards

Extending Qlik Capabilities

Maria Sandorova
Head of Product Development
Inphinity

Tomas Janco
Senior Developer
Inphinity



“Qlik is far more than a data visualization tool.

Far more...”



In my first 2 years as a full-time Qlik developer, I delivered two large Qlik projects with one dashboard only

Maria Sandorova

As a non-Qlik developer I delivered my company's largest Qlik-based integration to date after only few months of Qlik experience

Tomas Janco



3 bottlenecks in data processing

in a large financial company 



data preparation & cleaning
from multiple systems



manual processes in monthly
accounting closure



many different systems
in customer rating processing

3 bottlenecks in data processing

in a large financial company 

we solved with 



data preparation & cleaning
from multiple systems



manual processes in monthly
accounting closure



many different systems
in customer rating processing





data preparation & cleaning
from multiple systems

Requirements:

- 9 different data sources
- daily reload
- snapshots + data cleaning
- self-service management
- data lineage



AUTOMATION BY WRITE-BACK

L1: raw data
(data as is)



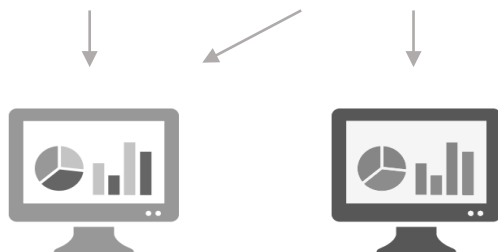
L2: clean data
(transformations)



L3: datamart



L4: apps



relevant sources, tables, fields and rows

1:1 data from data source

data consolidation, naming conventions

1:1 data from raw data



cross-table operations, advanced transformations

N:N from clean data

relevant domains, tables, fields and rows

1:1 from datamart



AUTOMATION BY WRITE-BACK

L1: raw data
(data as is)



L2: clean data
(transformations)



L3: datamart



sales data



HR data

L4: apps



relevant sources, tables, fields and rows

time savings

single source of truth

full changelog

documentation and additional information

responsibility and categories



cross-table operations, advanced transformations

N:N from clean data

relevant domains, tables, fields and rows

1:1 from datamart

Solution

Iterative improvement & optimization

11 subroutines

L1: 1 200 rows of script

L2: 500 rows of script

9 data sources, 150 tables, 2 000 fields

Loaded every day for almost 3 years up to now

...still used

#Source Systems

18

 Groups

9

#Source Tables

241

 QVDs

70

#Source Fields

3152

Fields

1125

App available here: inphinity.xyz/qw2020

DataMart

Prepare
Data manager

Analyze
Sheet

Narrate
Storytelling

Edit

1. systems specification

Selections

1. systems specification

1. systems specification

-> define relevant data sources and additional information

-> click recalculate to scan available tables & fields

2. tables selection

-> check "include" for tables you want in your datamart

-> insert inputs needed for proper loading of L1

3. fields management

-> check "include" for fields you want in your datamart

-> define transformations proper data cleaning

4. additional settings

-> enter global settings for datamart

-> define domains for assigning them to tables

Relevant data sources definition

Notice: in case of type "XLSX", the current app assumes that "system" is the file name and there is a sheet in the file called Sheets containing a list of sheet names

clear	save	recalculate	export					
ID	system	type	connection name	note	importance	changed by	last change	
1	sales_budget	XLSX	lib://datamart/excels/	internal simulations		einsetin	2020-04-22 15:44:46	
2	risk_settings	XLSX	lib://datamart/excels/	settings for monthly closure		inphinity	2020-04-22 16:58:25	
3	HR_data	XLSX	lib://datamart/excels/			inphinity	2020-04-22 16:58:25	
4	budget	MariaDB	lib://MariaDB	final budgeting data		hendricks	2020-04-22 16:58:25	
5	Tasks_management	XLSX	lib://datamart/excels/			netwon	2020-04-22 16:58:58	

6

Full changelog of data sources definition

ID	Q	log.system	Q	log.type	Q	log.connection	Q	log.note	Q	log.importa...	Q	log.actual	Q	log.change...	Q	log.change_date	Q
1		sales_budget		XLSX		lib://datamart/excels/		internal simulations		0		1		einsetin		2020-04-22 15:44:46	
1		sales_budget		XLSX		lib://xlsx/		internal simulations		0		0		newton		2020-04-22 15:43:44	
2		risk_settings		XLSX		lib://datamart/excels/		settings for monthly closure		1		0		inphinity		2020-04-22 16:07:49	
2		risk_settings		XLSX		lib://datamart/excels/		settings for monthly closure		1		1		inphinity		2020-04-22 16:58:25	
3		HR_data		XLSX		lib://datamart/excels/				2		0		doe		2020-04-22 16:09:08	
3		HR_data		XLSX		lib://datamart/excels/				3		1		inphinity		2020-04-22 16:58:25	
4		budget		MariaDB		lib://MariaDB		final budgeting data		1		0		hendricks		2020-04-22 16:27:26	
4		budget		MariaDB		lib://MariaDB		final budgeting data		2		1		hendricks		2020-04-22 16:58:25	
5		Tasks_management		XLSX		lib://datamart/excels/				2		1		netwon		2020-04-22 16:58:58	






Selections

1. systems specification

- > define relevant data sources and additional information

->click recalculate to scan available tables & fields

2. tables selection

-> check "include" for tables

-> insert inputs needed for

3. fields management

-> check "include" for fields

- > define transformations

⚙ 4. additional settings

-> enter global settings for

```
datamart
> define domains for
```

system_type	system_importance	system_name	table_name	field_type	field_name
-------------	-------------------	-------------	------------	------------	------------

Tables selection and L1 loading parameters

clear  save  recalculate  export 

[illegible]

App available here: inphinity.xyz/qw2020

☰

DataMart

Prepare

Data manager

Analyze

Sheet

Narrate

Storytelling

🔖

Edit

3. fields management

🏠

Selections

table_name

4 of 177

3. fields management

1. systems specification

2. tables selection

3. fields management

4. additional settings

→ define relevant data sources and additional information

→ click recalculate to scan available tables & fields

→ check "include" for tables you want in your datamart

→ insert inputs needed for proper loading of L1

→ check "include" for fields you want in your datamart

→ define transformations proper data cleaning

→ enter global settings for datamart

→ define domains for assigning them to tables

system_type

system_importance

system_name

table_name

field_type

field_name

Fields management & cleaning transformations

Notice: if the field name in the datamart should be the same as in the data source, use '=' in DM field name

clear

save

recalculate

export

system name	table name	field name	include	DM field name	mapping	L2 transformation (qlik syntax)
HR_data	Employee	emp_id	<input checked="" type="checkbox"/>	%Employee_id	<input type="checkbox"/>	
HR_data	Employee	gender	<input checked="" type="checkbox"/>	Gender	<input type="checkbox"/>	
HR_data	Employee	hire date	<input type="checkbox"/>		<input type="checkbox"/>	
HR_data	Employee	job title	<input checked="" type="checkbox"/>	Job_title	<input type="checkbox"/>	
HR_data	Employee	name	<input checked="" type="checkbox"/>	Employee_name	<input type="checkbox"/>	
HR_data	Employee	office_id	<input checked="" type="checkbox"/>	%Office_id	<input type="checkbox"/>	
HR_data	Employee	reports to	<input type="checkbox"/>		<input type="checkbox"/>	
HR_data	Employee	team	<input type="checkbox"/>		<input type="checkbox"/>	
HR_data	Employee	updated	<input type="checkbox"/>		<input type="checkbox"/>	
HR_data	Employee	year salary	<input type="checkbox"/>		<input type="checkbox"/>	
sales_budget	portfolio	Product Group	<input checked="" type="checkbox"/>	Products_group	<input type="checkbox"/>	if(len(trim("Product Group"))=0,'N/A','Product Group')
sales_budget	portfolio	Product name	<input checked="" type="checkbox"/>	%Product	<input type="checkbox"/>	
sales_budget	sales_LY	#Sales_LY	<input checked="" type="checkbox"/>	=	<input type="checkbox"/>	round(#Sales_LY,'0.01')
sales_budget	sales_LY	Country	<input checked="" type="checkbox"/>	%Country	<input type="checkbox"/>	
sales_budget	sales_LY	Month_LY	<input type="checkbox"/>		<input type="checkbox"/>	
19 rows			<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	

App available here: inphinity.xyz/qw2020



4. additional settings

1. systems specification

-> define relevant data sources and additional information

-> click recalculate to scan available tables & fields

2. tables selection

-> check "include" for tables you want in your datamart

-> insert inputs needed for proper loading of L1

3. fields management

-> check "include" for fields you want in your datamart

-> define transformations proper data cleaning

4. additional settings

-> enter global settings for datamart

-> define domains for assigning them to tables

Global datamart settings

clear save recalculate export

L1 path

lib://datamart/L1_raw

L2 path

lib://datamart/L2_clean

system_name

table_name

field_name

DM_field_name

Define the map for field values: M_prod_descr

If the map should be applied, be sure you use applymap() function in L2 transformation.

original value

new value



Dumbbells

Dumbbells



exercise ball

Exercise Ball



Sports bag

Sports Bag



Bottle

Steel Bottle



bottle

Steel Bottle



towel

Microfibre Towel



microtowel

Microfibre Towel



Domains management

Domains are used for assigning responsibilities on tables level

clear save export

domain name	icon	owner	last change
sales		Pinocchio	2020-04-22 15:40:30
risk		Peter Pan	2020-04-22 16:07:14
human resources		Cinderella	2020-04-22 16:07:14
accounting		Peter Pan	2020-04-22 16:08:39
R&D		Snow White	2020-04-22 16:08:39

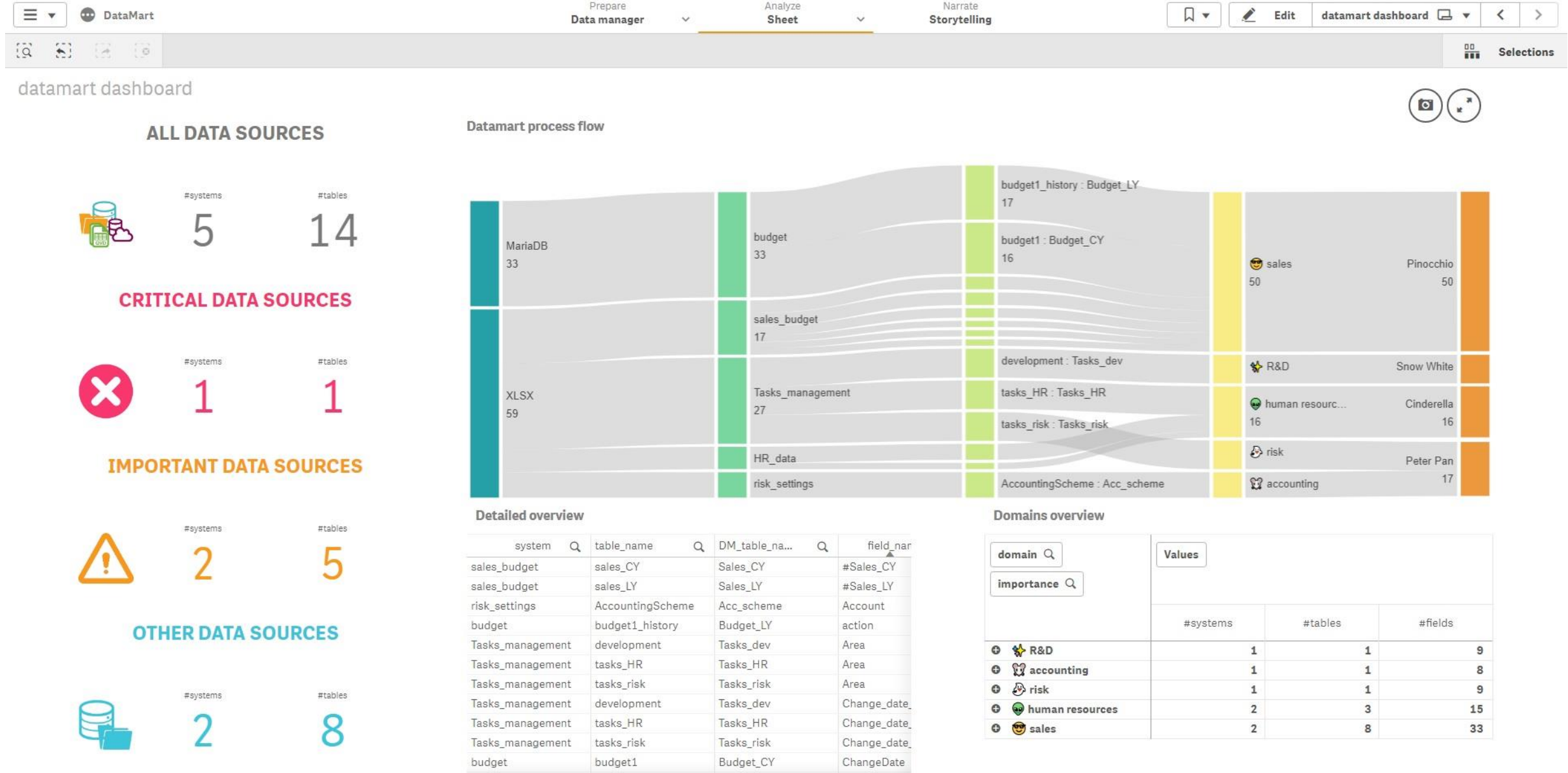
Developed by Maria Sandorova, Inphinity for QlikWorld 2020

Available for download: <https://inphinity.xyz/qlikworld2020>

Ideas for you to improve the app:

- load available data connections for systems definition
- add scripts for "snapshot" and "incremental" load types
- add new types of data sources
- load field values for maps definition
- load users from AD or other lists for domains management
- separate different steps to standalone qlik applications
- define your own approval process for datamart updates

App available here: inphinity.xyz/qw2020





manual processes in monthly
accounting closure

Requirements:

single source of truth

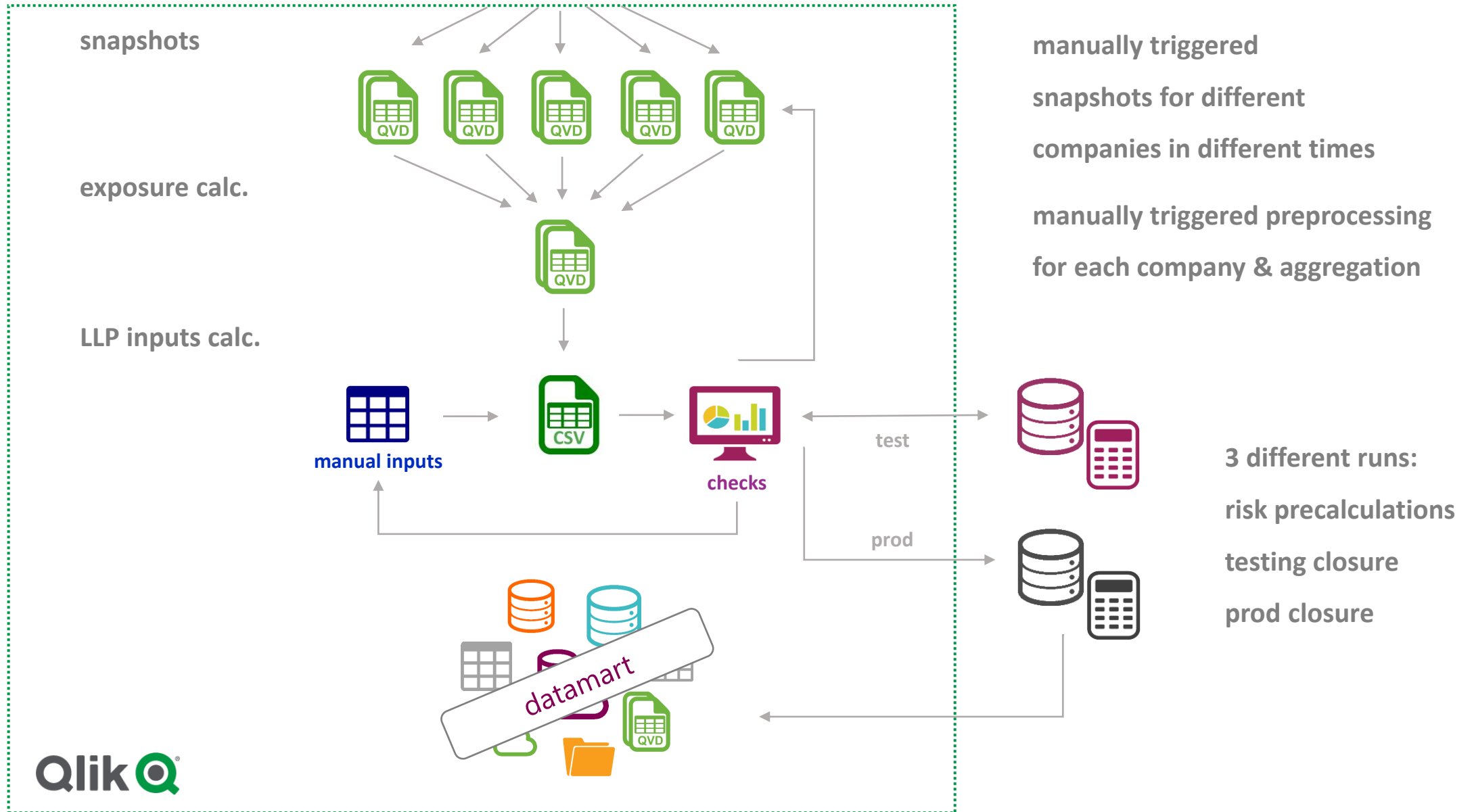
automated process across 3 departments

avoid many excel files in processing

self-service management

master data storage: datamart QVDs

Process:





Challenge

3 departments coordination

minimum apps

time efficiency

manually triggered steps

complex calculations

sending data to external DB



Solution

QS apps in one common stream

the very same apps with parameters

processing status (+Qlik Alerting)

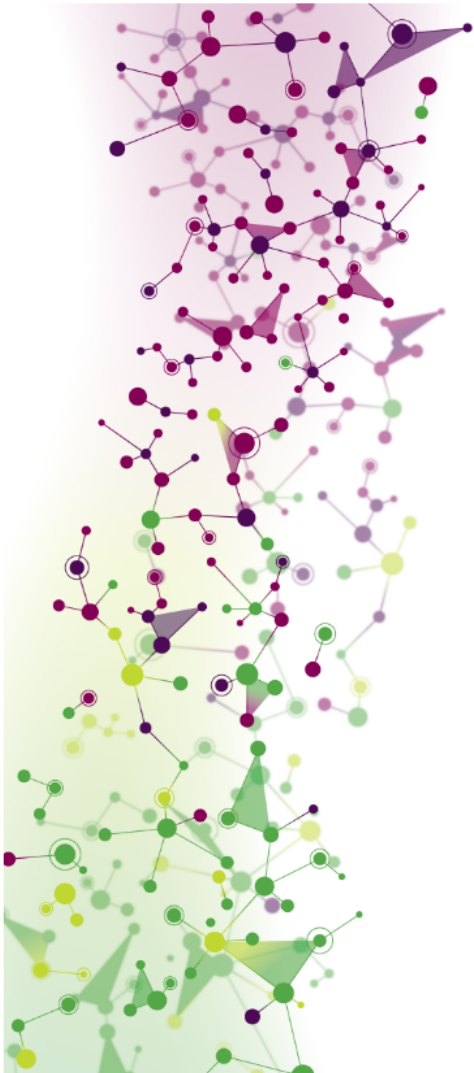
Reload Task Button (for free)  BUTTON & STATUS

more than a thousand rows of script

bulk insert app with a control (pre)run

set errormode = 0

Accounting Closure Management



Welcome Inphinity in Accounting Closure App,
Today is 4/21/2020.
Please set parameters and click on Reload.

General settings

Extract date (DD.MM.YYYY)

No data after the date will be used

Load type

Please select a load type

Data storage

Based on your settings generated data will
be stored here:

E:\data\risk\QVDs\05052020

1. Snapshots

Company

Start

Last reload: 4/21/2020, 8:55:46 AM
Status: Success

2. Exposure calculation

Data about different companies will be concatenated. Please run
this step after all companies have been snapshotted.

Calculate

Reloading.

3. LLP inputs

The specific dataset for LLP calculation will be generated and
stored into an XML file.

Generate

Last reload: 9/5/2019, 1:11:12 PM
Status: Failed

4. Import to external calculator

Database type

Import

Status: Never started



many different systems
in customer rating processing

Requirements:

less than 1 hour response time

user interface: JIRA

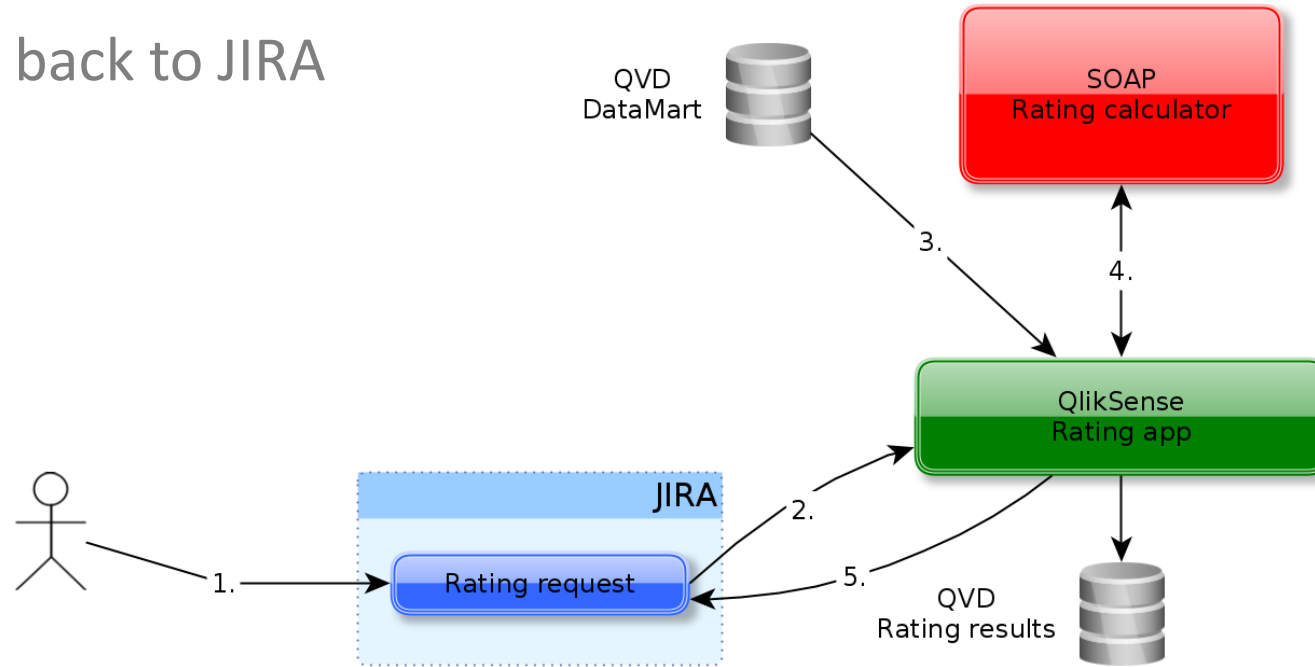
2 different rating calculators: SOAP services

full changelog

master data source: Datamart QVDs

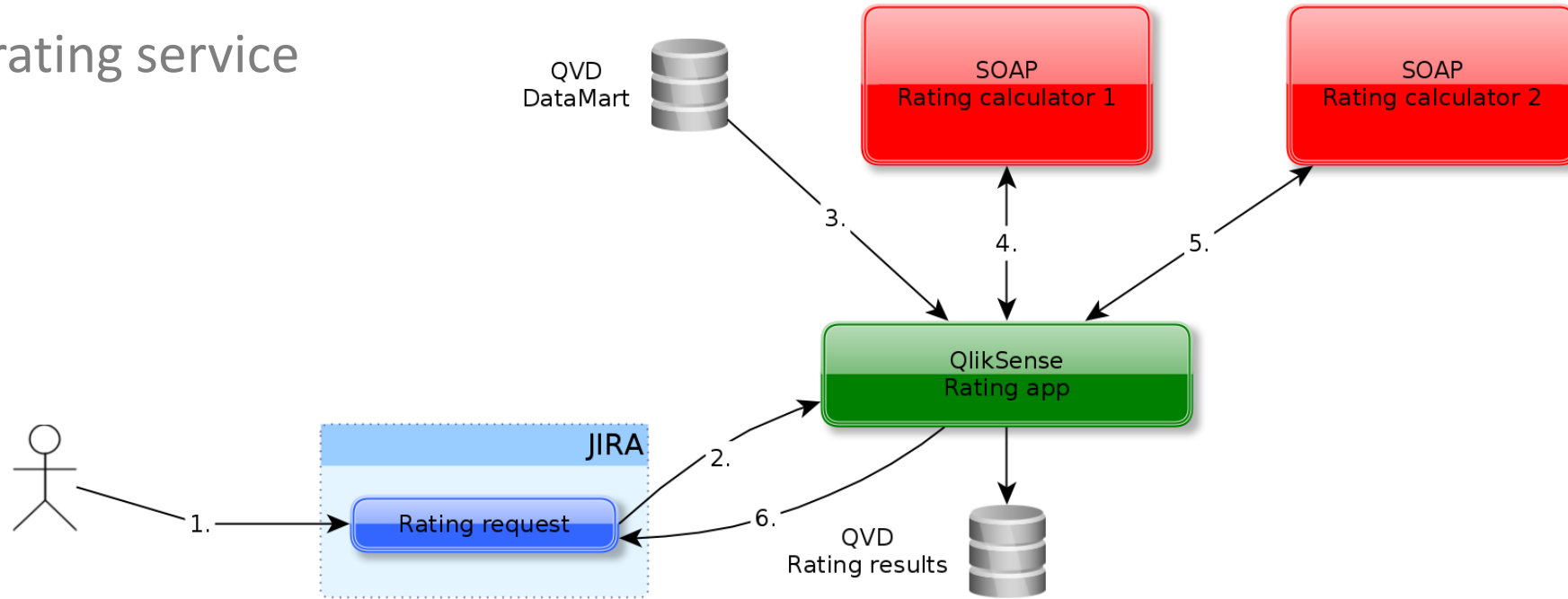
Request:

- write results back to JIRA



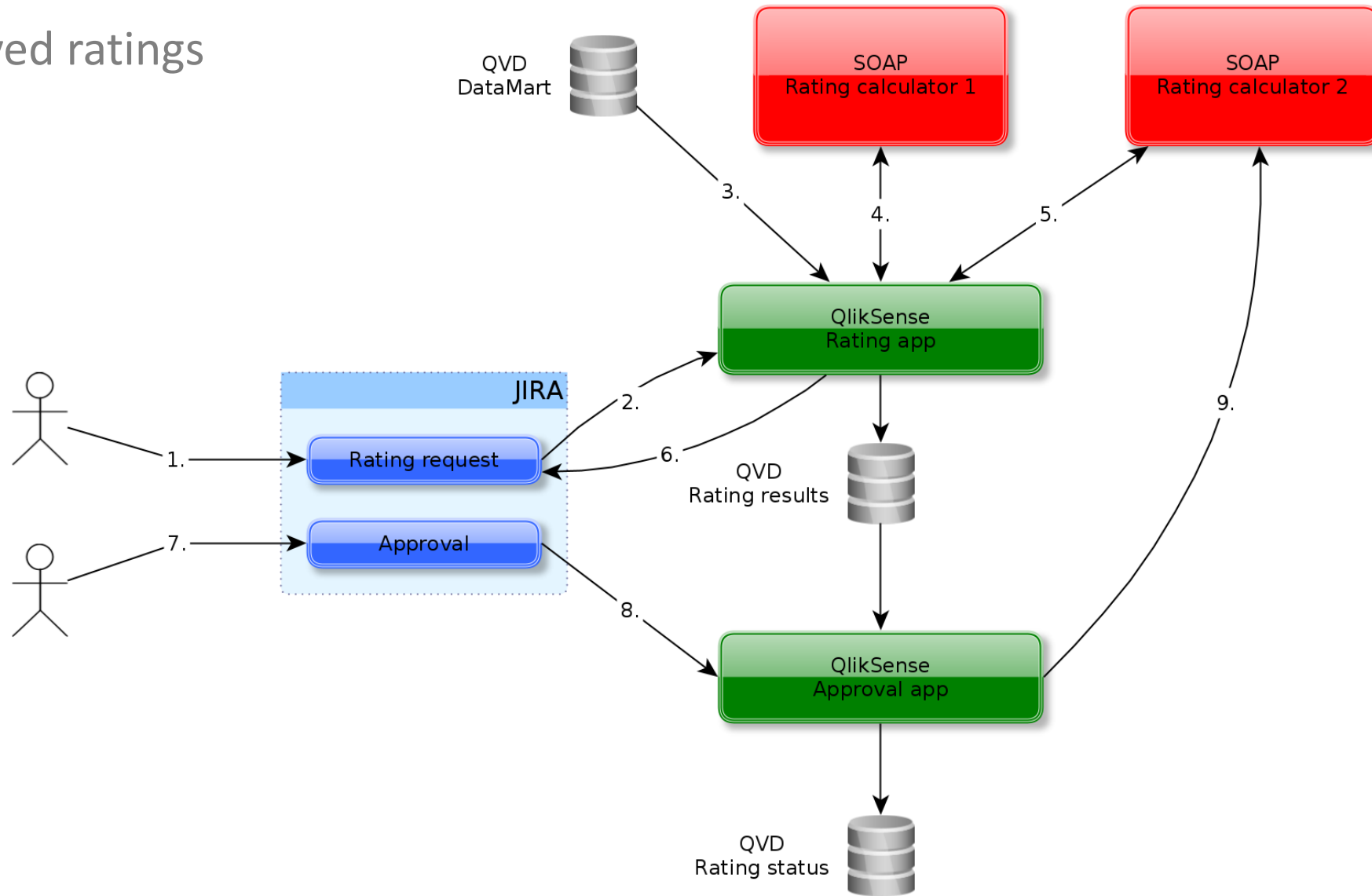
Request:

- add second rating service



Request:

- store approved ratings





Master data

- ordinary boring stuff
- LOADs, MAPPING LOADs, JOINs, GROUP BYs etc...
- 5 QVDs, 40 fields, 250 LOC

Approval workflow – wait for approval

- calculated ratings are stored in input QVD by the first app
- approval status stored in output QVD by second app
- all pending requests are re-read from JIRA for approval status
- all approved but not yet processed are stored

Avoids write conflicts between apps



- reading from JIRA
 - REST connector can do fine
- writing back to JIRA - HTTP PUT
 - custom HTTP connector, 1000 Java LOC

```
SELECT * FROM HTTP {  
  "url": "${vJiraURL}/rest/api/2/issue/${vIssueId}",  
  "method": "PUT",  
  "timeout": 30000,  
  "httpProxy": "10.10.10.1:8080",  
  "data": "{\"fields\":{\"customfield_11901\": \"${vPD}\", \"customfield_11902\": \"${vRating}\"",  
  "headers": {  
    "Content-type": "application/json",  
    "Authorization": "Basic ${vHttpAuthToken}"  
  },  
  "ignoreSSL": true  
};
```

- running on demand
 - NodeJS webhook listener triggers reload via API



SOAP Rating calculators

SOAP is XML over HTTP

- XML generating

```
[T_DSS_Request]:  
LOAD  
    FirstValue([Request ID]) as [Request ID],  
    Concat(  
        '< '&XMLName&'>'&  
        Replace(Replace(Replace(Replace(Replace(Value, '&', '&amp;'), '...  
        &'</ '&XMLName&'>'  
        , '' ) as Request  
RESIDENT T_DSS_DataTable  
WHERE [Request ID] = $(vIssueId) AND NOT(IsNull(XMLName));
```

- Custom authentication mechanism
→ just another SOAP call
- Access via network proxy
→ custom HTTP connector

- XML parsing

```
LOAD  
    $(vIssueId) as [Request ID],  
    responseBody as DSSReason,  
    TextBetween(responseBody, 'PD value=', '') as PD,  
    TextBetween(responseBody, 'Model value=', '') as Model,  
    TextBetween(responseBody, 'Rating value=', '') as Rating,  
    TextBetween(responseBody, 'reasonCode>', '</') as DSSReasonCode,  
    TextBetween(responseBody, 'returnCode>', '</') as DSSReturnCode,  
    TextBetween(responseBody, 'reasonText>', '</') as DSSMessage,  
    Now() as DSSTime,  
    '$(vDSSRequest)' as DSSRequest,  
    1 as @DSS_Loaded  
;
```



Fine tuning

Early exit

```
IF '$(vEnableEarlyExit)' = '1' THEN
  RequestCountCheck:
  LOAD Count(DISTINCT [Request ID]) as RequestCount
  RESIDENT R_JIRA_MasterTable
  WHERE NOT ISNULL([__KEY_issues]);

  Let vRequestCount = Peek('RequestCount');
  TRACE >> Request count: $(vRequestCount) <<;

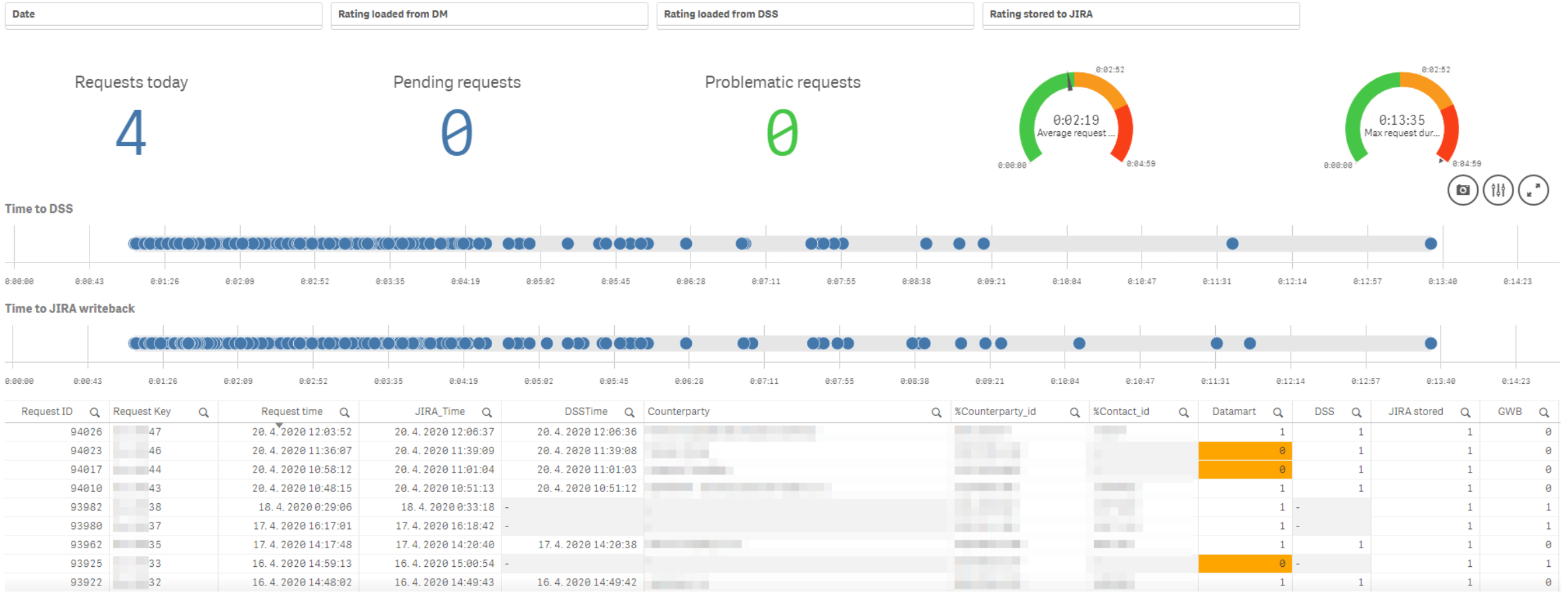
  IF Peek('RequestCount') = 0 THEN
    DROP TABLE RequestCountCheck;
    DROP TABLE R_JIRA_MasterTable;
    [T_Results_Buffer]:
    NOCONCATENATE LOAD * FROM [lib://$(vQVD)/JIRA_Results.qvd] (qvd);
    TRACE >> No request to process - ABORT <<;
    Exit Script;
  END IF;

  DROP TABLE RequestCountCheck;
END IF;
```

Fine tuning

Status dashboard

Dashboard



3 bottlenecks in data processing

in a large financial company 

we solved with 



data preparation & cleaning
from multiple systems



manual processes in monthly
accounting closure



many different systems
in customer rating processing



“Qlik is far more than a data visualization tool.

Far more...”



Maria Sandorova

✉ sandorova@inphinity.xyz

in [maria-sandorova](#)

🐦 [sandorovaMaria](#)



Tomas Janco

✉ janco@inphinity.xyz

in [jancotomas](#)



inphinity

www.inphinity.xyz/contact