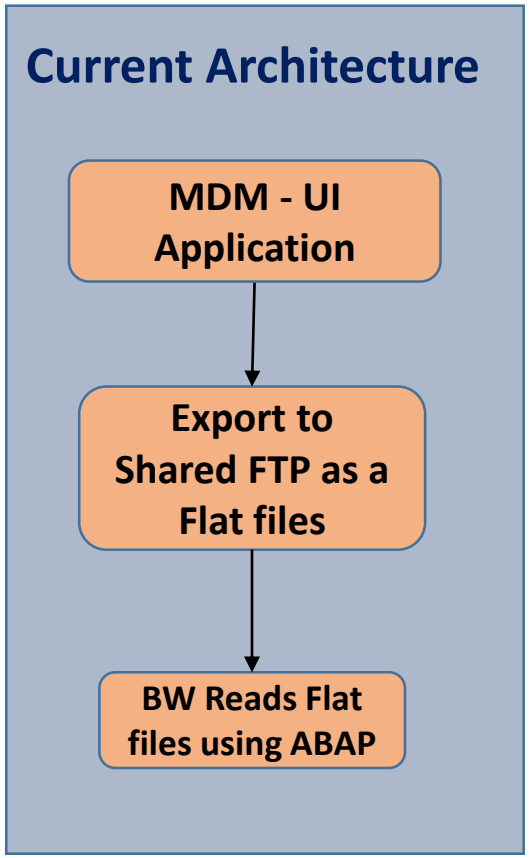
MDM Alternate Solution Options

Joseph Yeruva

MDM Application - Project Scope

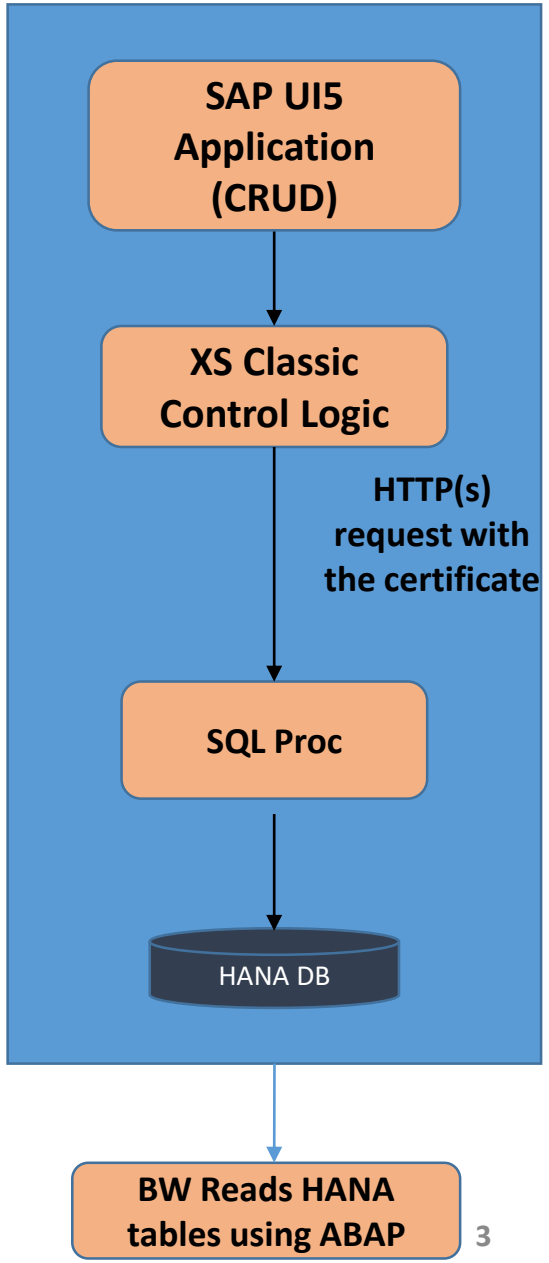
Current Architecture and Process :

- MDM Application used to enter master data using SAP MDM Data manager 7.1
- Maintain tables in MDM → Export to shared drive Excel format→ Read the files in BW ABAP
- Approximate no of tables - 140
- Avg no of columns in tables - 10
- Avg no of records in tables – 1000
- How frequently updated – Daily once ;
- Where all these MDM data is used list → BW,
- Who updates? Nick, Emily's Team .
- No of users? 10
- While creating/ updating data in tables Search help(descriptions) are there or not? Drop down and text available for few.
- How the flat file export is done – leapftp? Frequency of export? hourly
- Do we have Audit log/change history with details like who updated and when? No
- MDM tool Version - SAP MDM Data manager 7.1
- Main data sets ? 10 -20



Option: 1 SAP HANA XSC - Native Application

- SAP HANA Extended Application Services (SAP HANA XS), classic model, embeds a full-featured application server, Web server, and development environment within SAP HANA.
- Applications can be developed and deployed directly on SAP HANA XS, which exposes them to end users through a web interface.



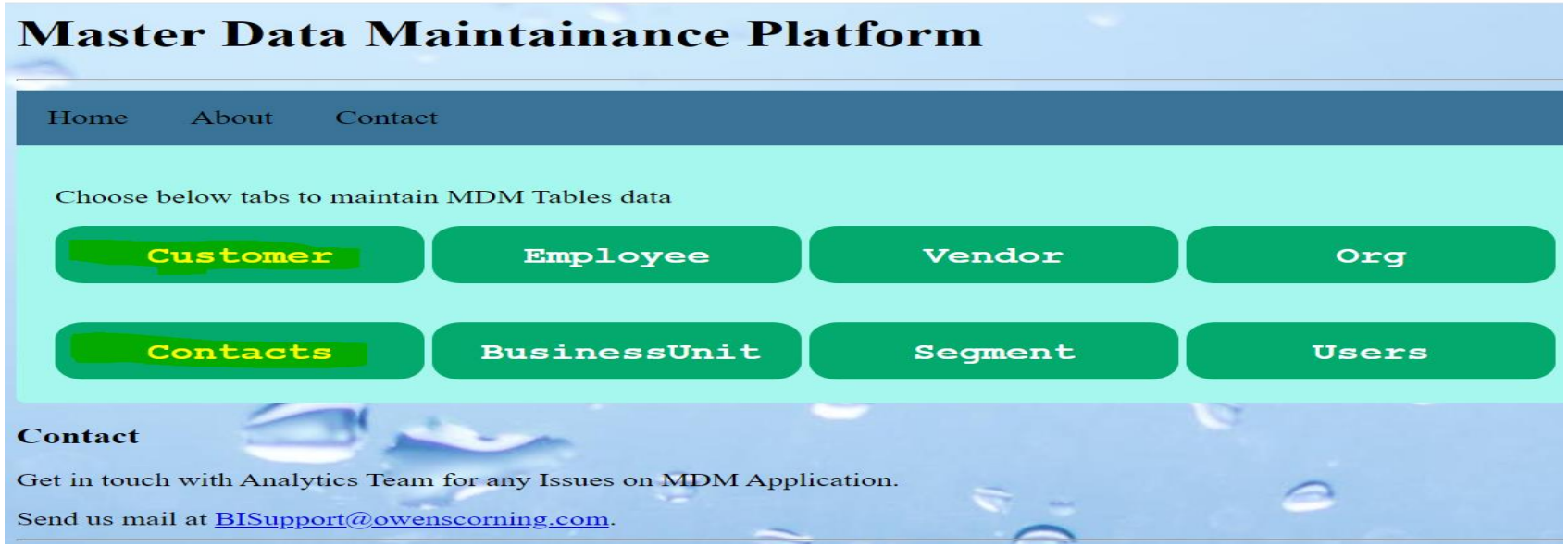
Pros and Cons:

- Our Environment is on XS Classic.
- This architecture aligns with future S/4 HANA embedded analytics
- Based on complexity Skillset needed – Java scripting, UI5, HANA PL/SQL
- Migration to XSA needed to sustain in future . XS Classic is already removed from HANA As A Service and HANA Cloud.
- This solution can be leveraged to maintain any tables data in future, avoid manual load of adhoc Flat files .(ex:frieght table, Inflation /Unit cost).

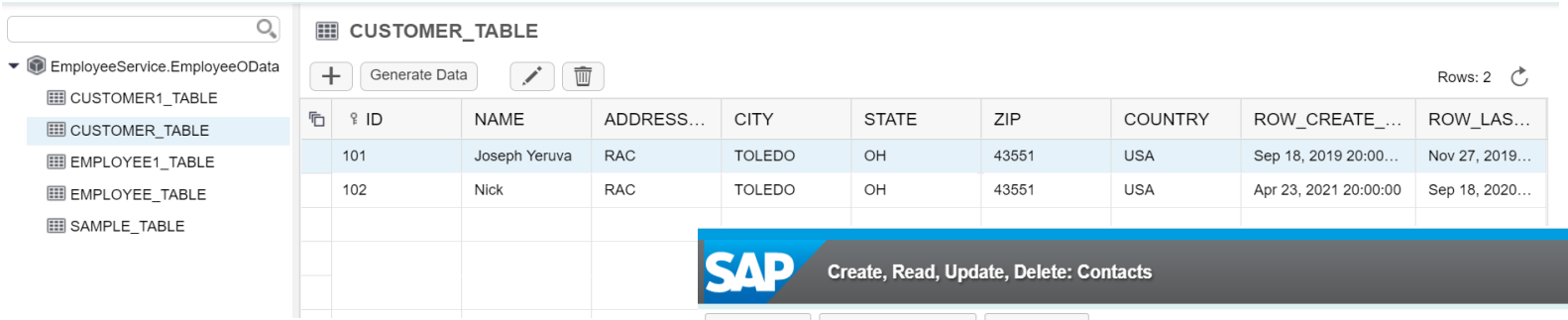
Demo Link:

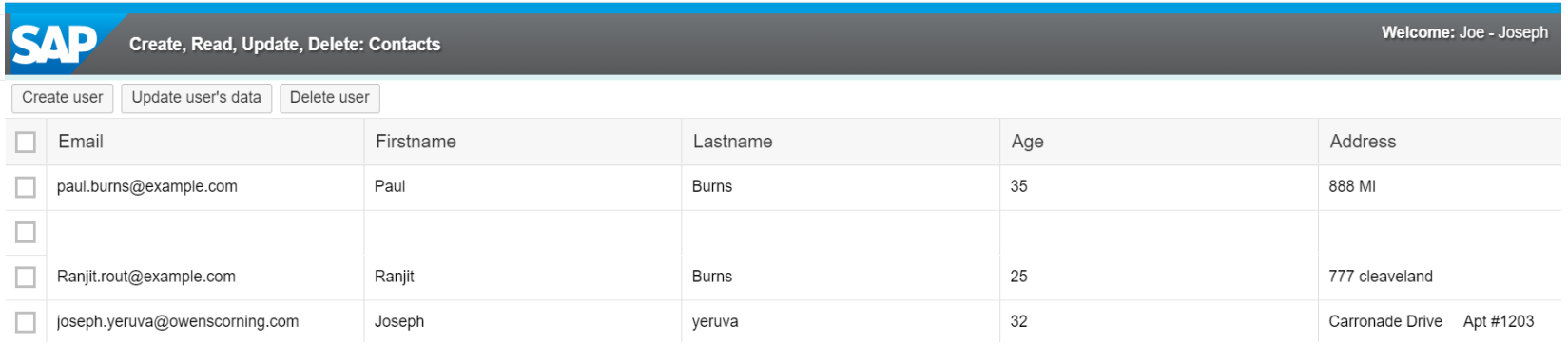
- https://saph9dv.owenscorning.com:4335/EmployeeService/index1.html

Demo : https://saph9dv.owenscorning.com:4335/EmployeeService/index1.html



1. Click on above link
2. Login using H9D Credentials
3. Click on Customer, Contacts Tabs
4. Select table and enter/edit the data



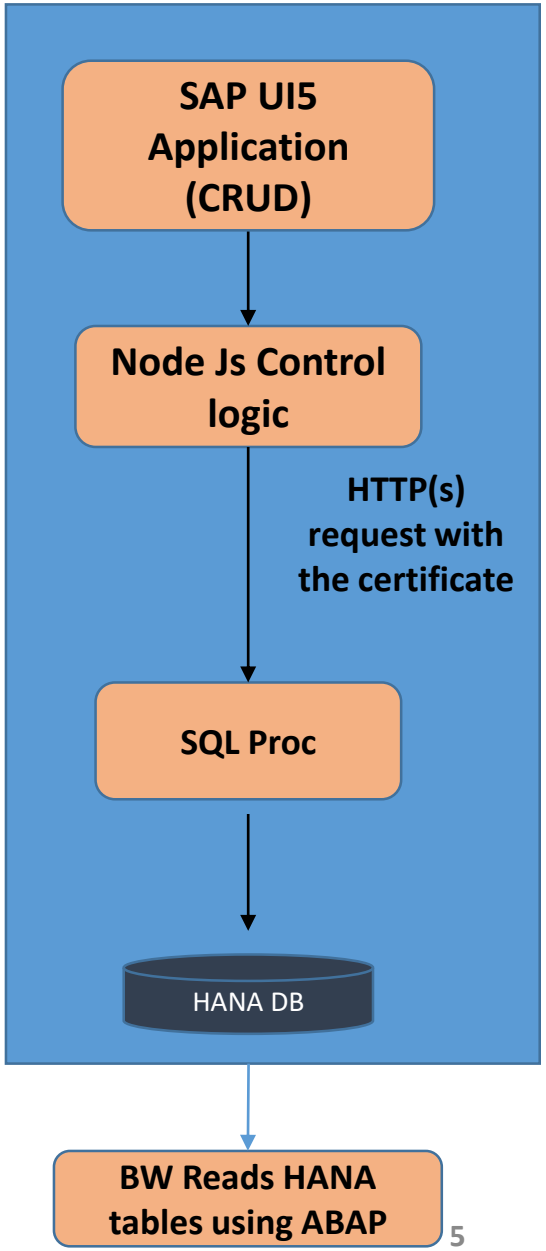


Option: 2 SAP HANA XSA - Native Application

- SAP HANA Extended Application Services (SAP HANA XSA), XS Advanced model, embeds a full-featured application server, Web server, and development environment within SAP HANA.
- Applications can be developed and deployed directly on SAP HANA XS, which exposes them to end users through a web interface.

Pros and Cons:

- Currently we are not on XSA,XSA need to be installed.
- Based on complexity Skill set needed – Node.js, Java scripting, UI5, HANA PL/SQL

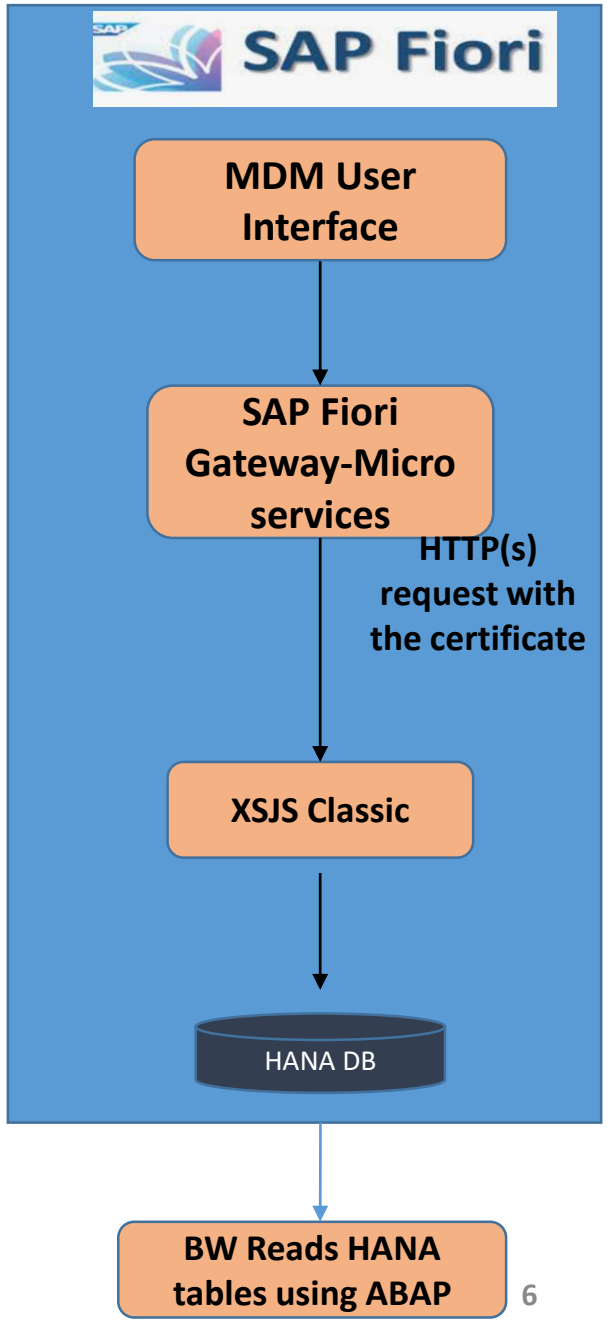


Option 3 : SAP Fiori

- Application(User interface) can be developed Using Firori/UI5 technology and deployed directly on SAP HANA XS, which exposes them to end users through a web interface.

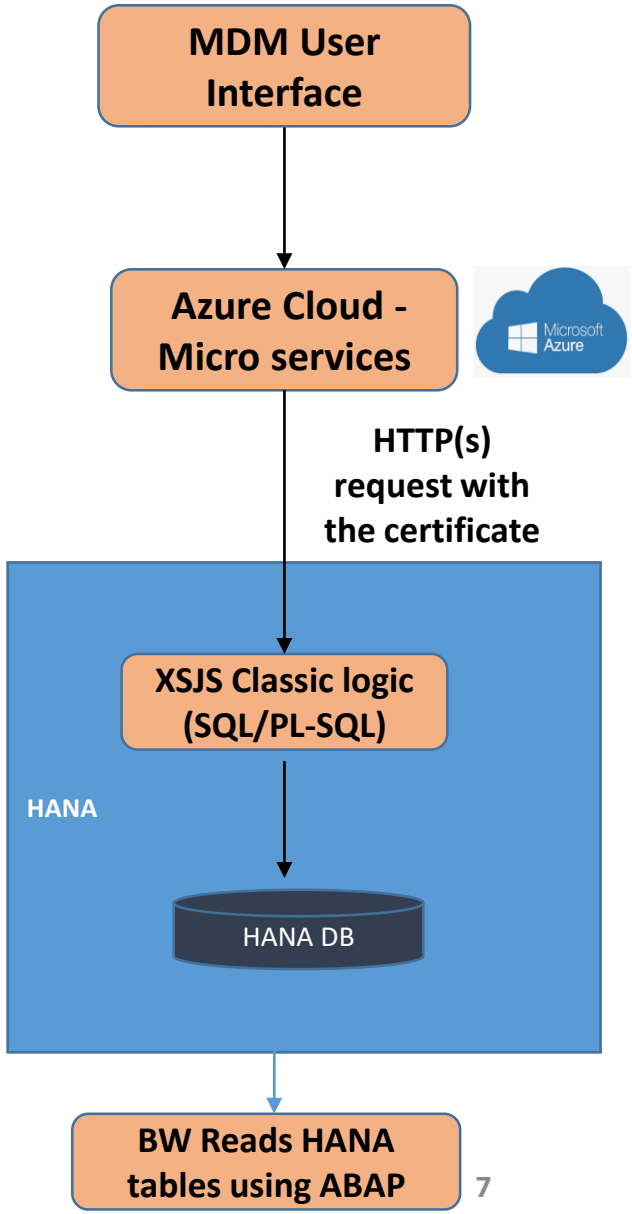
Pros and Cons:

- Need Fiori Gateway server/Front end UI5 .
- Based on complexity Skill set needed – Fiori UI5,Java scripting, HANA PL/SQL



Option: 4 Azure AND HANA XSJS - Classic Native Application

- Application(User interface) can be developed in .NET/Java and deployed directly on cloud.
- SAP HANA XS API'S Need to be created in SAP HANA and can be exposed to front end UI.



Pros and Cons:

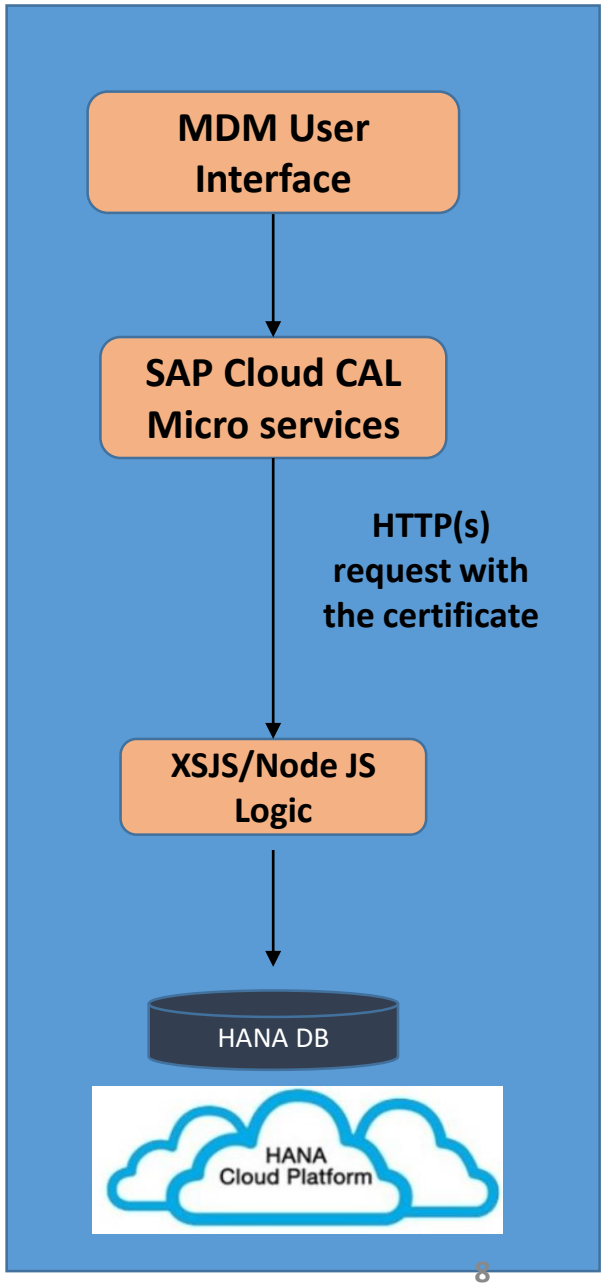
- Need to spin up Cloud instance for Application(User interface)
- Based on complexity Skill set – UI Using Java/.net and XSJS/ HANA PL/SQL

Option 5 : SAP Cloud Application (CAL)

- Applications can be developed and deployed directly on SAP cloud platform , CAL will exposes UI to end users through a web interface.

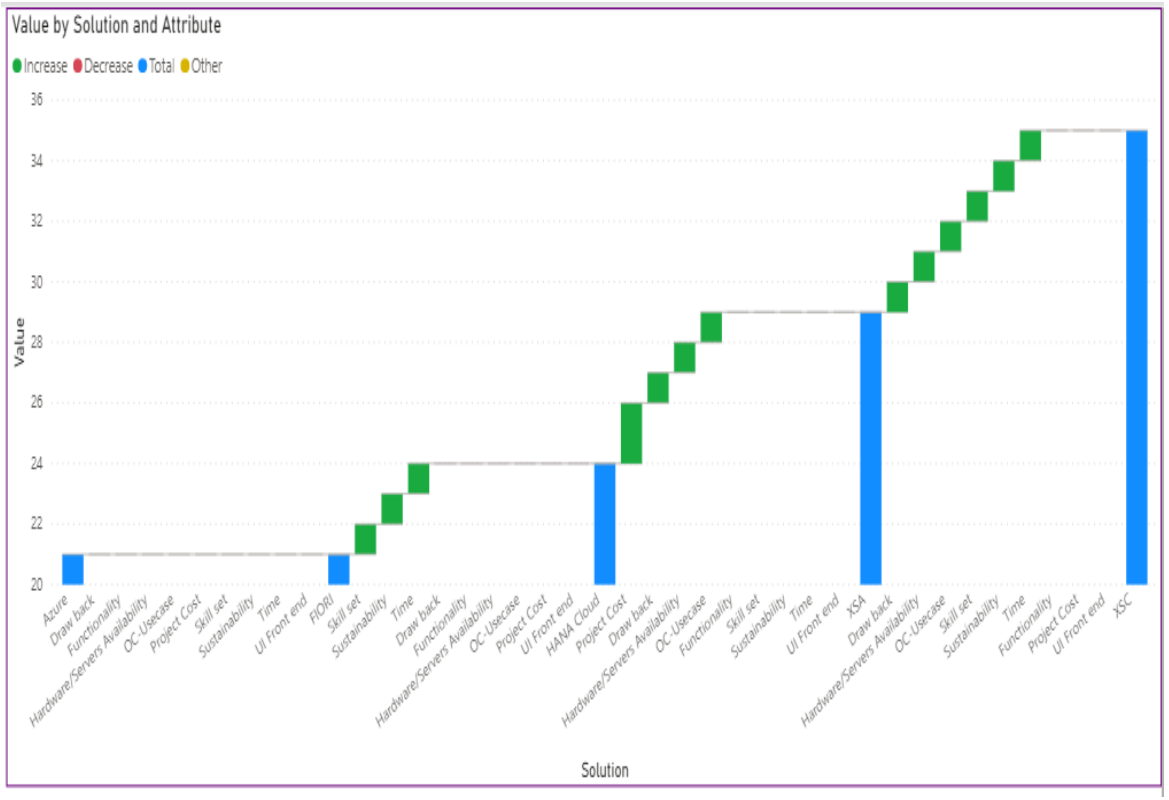
Pros and Cons:

- We need to spin of cloud instance .
- our data will be in cloud . Consumption in BW will be difficult.
- Based on complexity Skill set needed – Node.js, Java scripting, UI5, HANA PL/SQL



Comparison of each Solution:

Note: More points Means Good. 5 is Max.



Sno	Aspect	XSC	XSA	HANA Cloud	FIORI	Azure
1	Functionality	4	4	4	4	4
2	UI Front end	3	3	3	3	3
3	Project Cost	4	4	2	2	2
4	Hardware/Server Availability	4	3	2	2	2
5	Skill set	4	3	3	2	2
6	Time	4	3	3	2	2
7	Sustainability	4	3	3	2	2
8	OC-Usecase	4	3	2	2	2
9	Draw back	4	3	2	2	2
	Total Points	35	29	24	21	21

Steps for HANA XSC Solution:

- Create tables in HANA
- Load initial data from flat files
- Create XSJS End points
- Create User interface
- Replace the Flat file look up with HANA Table in BW transformations

Implementation cost Analysis:

No License cost (We have Enterprise HANA)

For Combination (50 simple+ 30 Medium + 20 High):
2320 hrs = 290 days = 14 months

Project Cost:
2 off shore + 2 Consultants = 7 months = 186K

UI5 Training cost 4000 $ for future Support.

Simple Interface (Low code)	Medium Complex	High Complex
Just CRUD Operation, No fancy UI, No backend processing(Transformations)	CRUD + Fancy UI(Ex: search help,Value help),No processing Logic(Transformations)	CRUD + Fancy UI(Ex: search help, Value help), with processing Logic(Transformations)
16 hrs /table/one resource for complete development	24hrs /table/one resource for complete development	40hrs /table/one resource for complete development
for 100 tables/one resource = 1600 hrs = 200 days = 9.5 months	for 100 tables /one resource = 2400 hrs = 300 days = 14 months	for 100 tables/one resource = 4000 hrs = 500 days = 24 months
We can do internally 2 offshore developers (@ 50 $/hr)	One UI5 Consultant , 5 months (@ 100 $/hr) 2 offshore developers (@ 50 $/hr)	One UI5 Consultant for 6 months (@ 100 $/hr) One Java/Node JS Consultant for 6 months(@ 100 $/hr) 2 offshore developers (@ 50 $/hr)
Total Cost = 5 months = 80K	Total Cost = 7 months = 200K	Total Cost = 8 months = 390K

Next Steps:

- Analyze the required tables, features and complexity on existing MDM Tables
- Any other Tables to be included in the scope.? (ex:frieght table, Inflation /Unit cost).
- Segregate them into complexity level to estimate the exact Project cost
- Leverage any other efficient options (Ex: Power apps/ SAP MDM) on this solution.

Thank You

Any Questions ?

Contact : joseph.yeruva@owenscorning.com