RiskShield ESA

DRAFT

Table of Contents

References

Type	Link to reference
Program / Local Board initiative	TBA
Theme	TBA
EPIC One Pager(s)	
QBR Memo	
Current State Architecture	
Target State Architecture	
Applicable Reference Architecture	
Product Architecture Roadmap	
Related EPIC Solution Approaches	RiskShield Implementation details

Stakeholders

Role	Responsibility	Name
Tribe Lead		
IT (Area) Lead	Informed	
Feature Engineer	Consulted	

Domain Architect	Responsible (Author)	Patrick Faragou
Enterprise Architect	Accountable	
Platform Architect	Consulted	
Information Architect	Consulted	
Security Architect	Consulted	
Software Engineer	Consulted	

Value Proposition & Scope

RiskShield is the SAAS that allow to evaluate the risk of different operations that could be done by our customers when they used the service provided by our applications. The goal of integrate this tool is to enhance our capacity to detect and prevent fraudulent operations and so reduce the economic and reputational negative impacts that suffer the bank when such events happen. RiskShield evaluates the risk of an operation "on demand" according to a large set of data (pre loaded and in the current context) that should be transmitted to the service and that should be up to date to allow Riskshield to perform evaluations as precise as possible.

The objective of this ESA is to propose a high level architecture reference in order to implement a solution to the feature described in RiskShield Implementation details as the Data Feeding use case.

The data feeding is performed through several steps:

- **initial load**: allow to retrieve and copy all necessary data needed to risk evaluation from our local data stores ("source") to the RiskShield data store system ("destination").
- **continuous feeding**: allow to continuously updates "destination" accordingly to all changes performed in the "source" data so source and destination has each a consistent snapshot of the data.
- **reconciliation**: allow to periodically check the previously commented "consistency" and, if necessary, perform the corrective operations needed to get back to a consistent state between "source" and "destination".

The data feeding can be done thanks to 2 kind of process:

- **batch with files**: thanks to periodic data extraction into files and transmission of such files from source to destination where it should loaded.
- **near real time thanks to events**: capturing data state changes on "source" and sending it to a Kafka topic that will be consumed by and loaded into "destination".

Key Assumptions, Decisions, Risks, Constraints & Debts

Key Assumptions

Provide a list of assumptions on which solution approach is developed

ASSUME ID	Description	Status
KA01	Bacth processes will be performed by the Big Data Platform (BDP). Use of the BDP is still under discussion.	PENDING
KA02	All the data that will be transferred to "destination" and necessary for risk evaluation is currently available in the BDP own data store (HDFS - Hadoop Distributed File System), FOS (Oracle customer database).	PENDING
KA03	Most of the information needed to be transmitted through the near real time process will be available thanks to currently existing event sent to Kafka topics and Rabbit Queues.	PENDING

Key Design Decisions

Provide a list of foundational design decision on which solution could be developed (? can we create a link to Decision registry, there could be a lot more solution specific decisions, less general decisions, but not have duplicate registry could be preferred)

Decision ID	Decision Description	Status	Submission date	Decision date
KD01	One one hand, batch data feeding will be performed with files sent to RiskShield. On the other hand, real time data feeding will be performed thanks to events that will be sent to RiskShield on a specific topic that RiskShield is ready to consume.	CONFIRMED		
KD02	Continuous data feeding will be performed thanks to both processes previously defined with a preference to the near real time transmission thanks to event.	CONFIRMED		
KD03	Use of events is recommended in order to perform continuous feeding of data, particularly for data that have a great impact on the risk evaluation and also for data that have less impact. Although, in case information is not available in a current existing events, it should be considered if it worth it to create a new event or if it is more efficient to use file batch process type.	CONFIRMED		
KD04	For the near real time process, a unique avro schema will be used to sent a "standard" event to a unique topic that Rsikshield will consume. Such event will have to be composed and generated on a new building block in charge of consuming all the necessary local events that contain the relevant information to be sent to RiskShield. According to the information needed in the standard event to be sent to Riskshield, enrichment may be necessary to compose the standard event.	CONFIRMED		

Key Risks

Provide a list of risks that may impact the creation of this approach, delivery and should be mitigated..

RISK ID	Description	Status
KR01	As RiskShield currently require the standard event to be composed of the complete information, it may trigger redundant request, particularly in case of event enrichment. Such redundant operation could lead to overload some datasources. Technical solution such as caching should be considered if such use scenario is detected. Such problem should be confirmed before being considered in the solution.	PENDING
KR02	As the same data could be sent to Riskshield by event or by file, the receptor of the information (RiskShield) will have to deal with race condition scenario such as loading data file while events are received in the same time.	PENDING

Key Constraints

Provide a list of constraints that may hamper the creation of this approach, technological constraints, delivery constraints...

CONS ID	Description	Status
KC01	File transmission between source and destination should be done with XFB protocol as soon as possible.	PENDING
KC02	Once generated, files should be encrypted.	PENDING
KC03	Files should be sent from BDP to XFB Sender by SFTP	PENDING
KC04	RiskShield Event Component Building Block (find a better name...) should be TPA compliant. Use of the TPA sidecar is recommended.	PENDING

Architectural Debts

<< this list can be the same as Architectural Debt Item Service registry, architect should be able to register new Debt items inside this page.. and only the ones related to this solution should be listed here.. They will be visible also in global registry >>

DEBT ID	Description	Tactical solution	Target solution	Cause	Risk	Planned Closure Date
DB001	This debts is registered as a possible enhancement and is not in scope of the target solution.	File transfer to RiskShield should be done in 2 steps. First step deposit the file generated by BDP task in an NFS, second step is to retrieve and send the file via XFB.	Send the file via XFB directly from the BDP task.	Functionality currently not available in BDP but may be developed in future.	File generated is copied in NFS in the first step without applying XFB that should be the standard way to transfer file in a secure way.	?
DB002	This debts is registered as a possible enhancement and is not in scope of the target solution.	File transfer to RiskShield should be done in 2 steps. First step deposit the file generated by BDP task in an NFS, second step is to retrieve and send the file via XFB.	Send the file to an S3 document manager instead of an NFS.	Functionality currently not available	Unavailability of integrated functionality of such a specialized service like (versioning, high availability,...)	?

Logical View

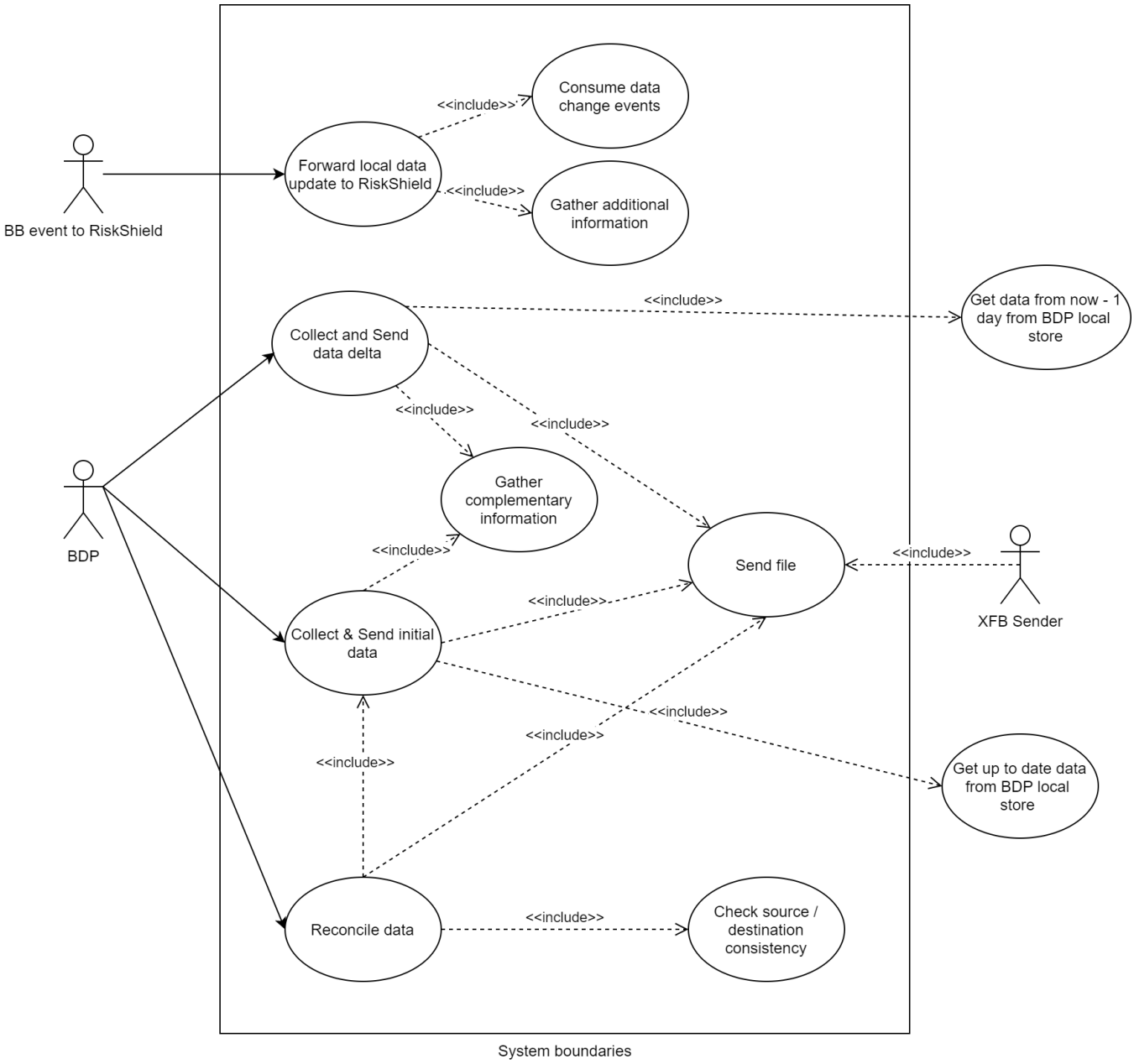
The system is composed of 3 actors.

BB event to RiskShield: until we find it a better name, it is the building block that will listen and consume event that are relevant to riskshield in order to update dynamic/critical data that sould be up to date as soon as possible in "destination" in order to perform a risk evaluation as pertinent as possible.

BDP: execution platform of the process that will

- retrieve data needed to initialize the "destination", generates file and trigger the file transmission.
- retrieve "delta data" each days, generates file and trigger the file transmission.

XFB Sender: execution platform of the process that will send via XFB the files generated by BDP.

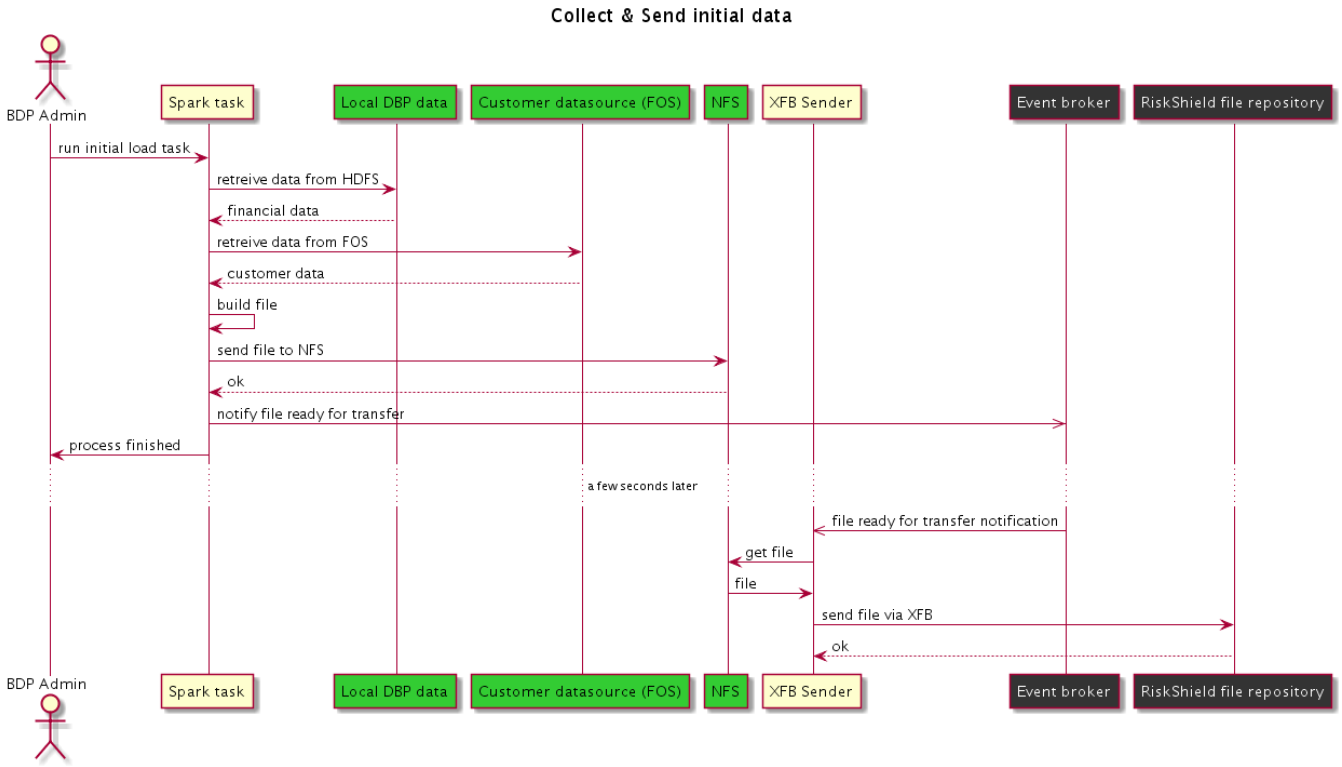


Key Use cases / Features Identified and Acceptance Criteria

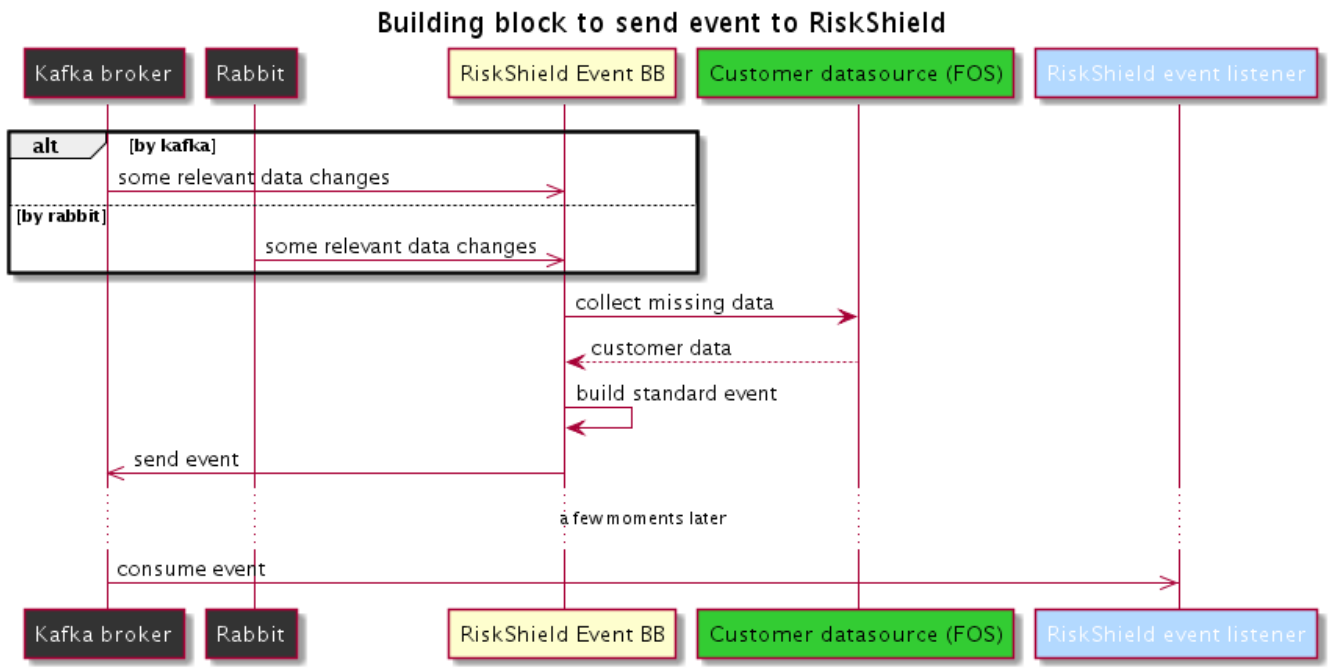
Use Case / Feature ID	Description	Acceptance Criteria
UC01	**Collect and send initial data**: represent the process that is in charge to generate the file that should be build with all the data needed by RiskShield	
UC02	**Gather complementary information**: represent the process part that retrieves from others datasource the data that is not directly available from the BDP local datastore (HDFS)	
UC03	**Send File**: represent the process part that is in charge of sending the file to RiskShield via XFB protocol	
UC04	**Forward local data update to RiskShield**: process that listen and consume the relevant events, compose and build the event to send to Riskshield and send it.	

UC05	**Consume data change events**: represents the part of the BB that will listen to all topics and queues necessary to forward real time data change to RiskShield.	
UC06	**Gather additional information**: represents the part that will be in charge to retrieve the additional information needed in case enrichment would be necessary to compose "standard" event.	
UC07	**Get up to date data from BDP local store**: represents the process currently offered by the BPD that allow to retrieve financial data (from Profile BOS) an make it available through the BDP local data store. Out of the scope of the project but described here for a better understanding.	
UC08	**Collect and send delta data:** represent the process that will periodically (each day) retrieve the data updated since the previous day and send a file with the changes detected.	
UC09	**Get data from now -1 day from BDP local store**: represents the functionality currently offered by the BPD that allows to retrieve financial "delta information" according to a timestamp. Out of the scope of the project but described here for a better understanding.	
UC10	**Reconcile data:** represent the process that check for data consistency and trigger a data correction process in case the consistency level is considered as insufficient.	
UC11	**Check source / destination consistency:** process that calculate the consistency rate and trigger a new "collect and send initial data" use case if consistency is considered as unsatisfactory.	

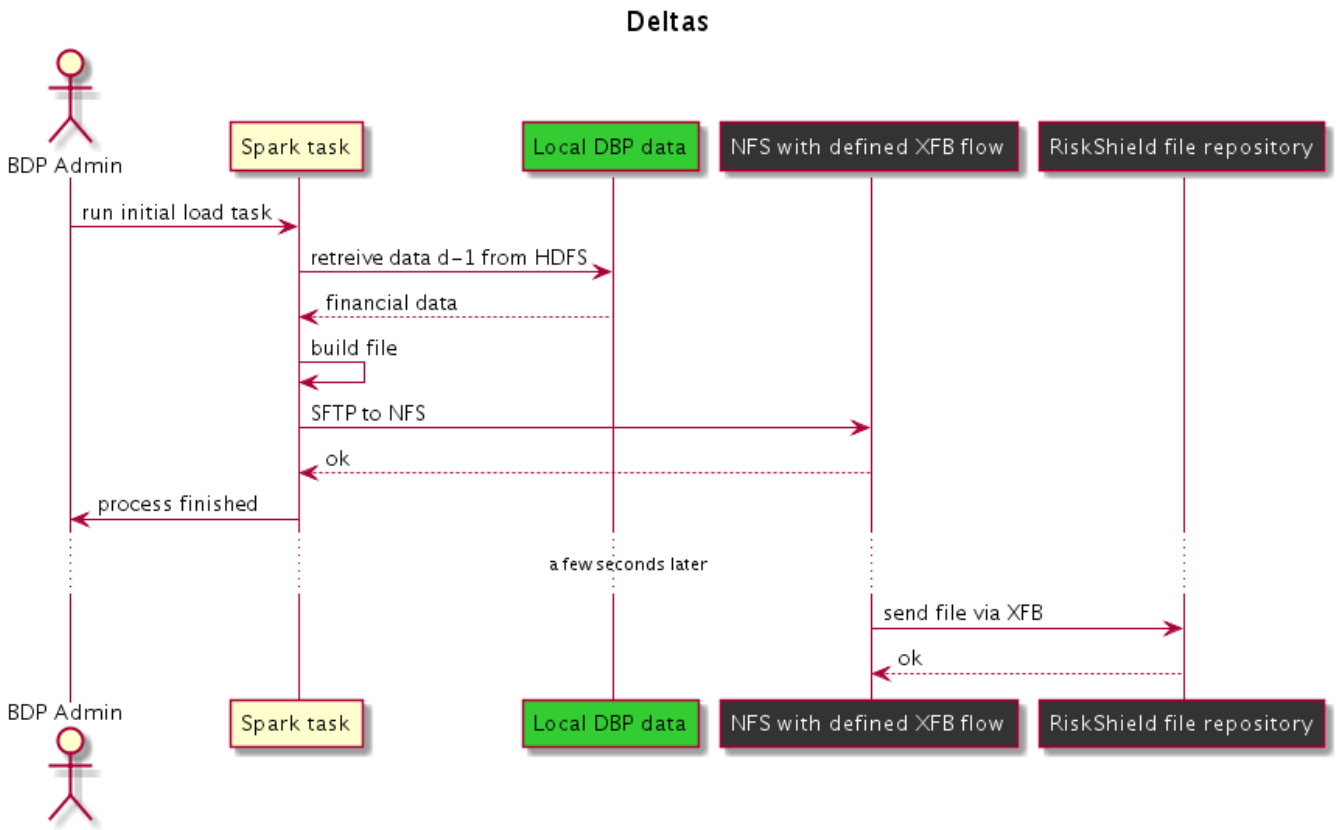
Collect & send data use case sequence diagram.



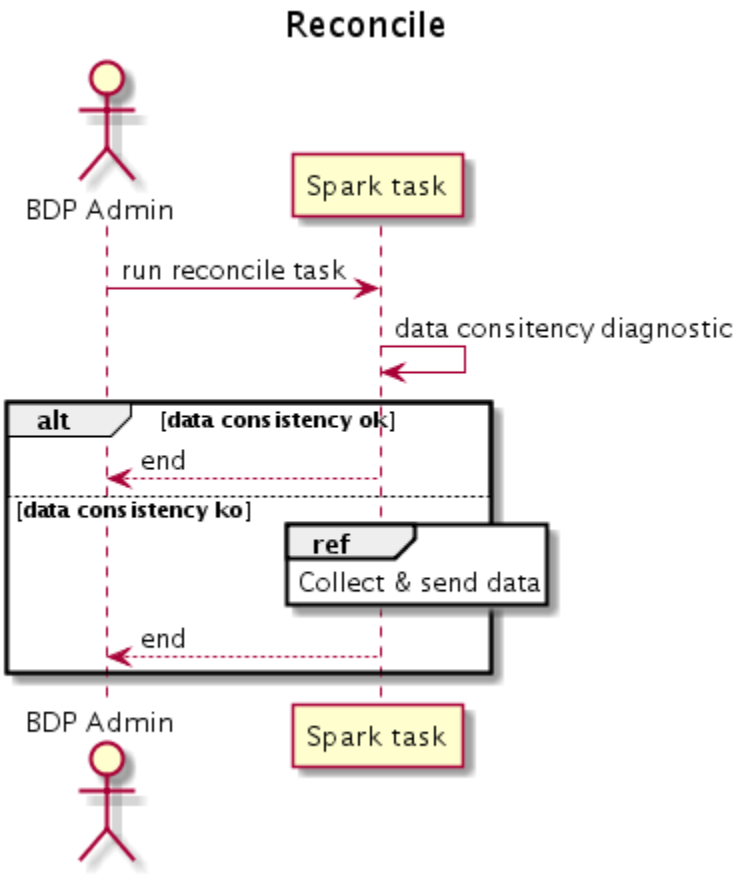
Forward local data update to RiskShield use case sequence diagram



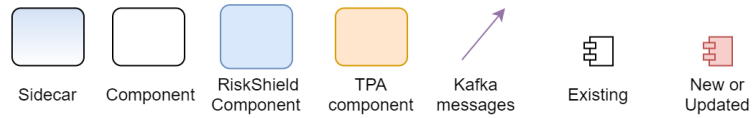
Collect & Send data delta



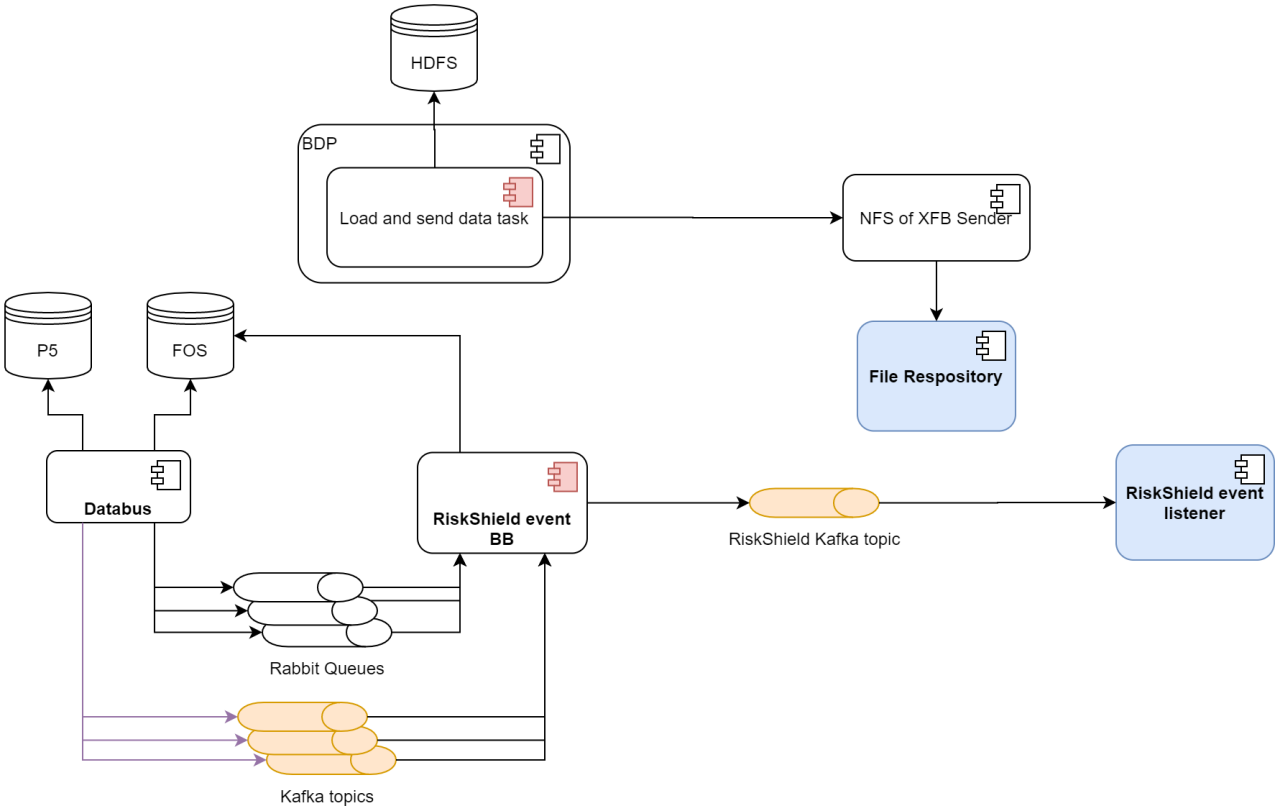
Reconcile data

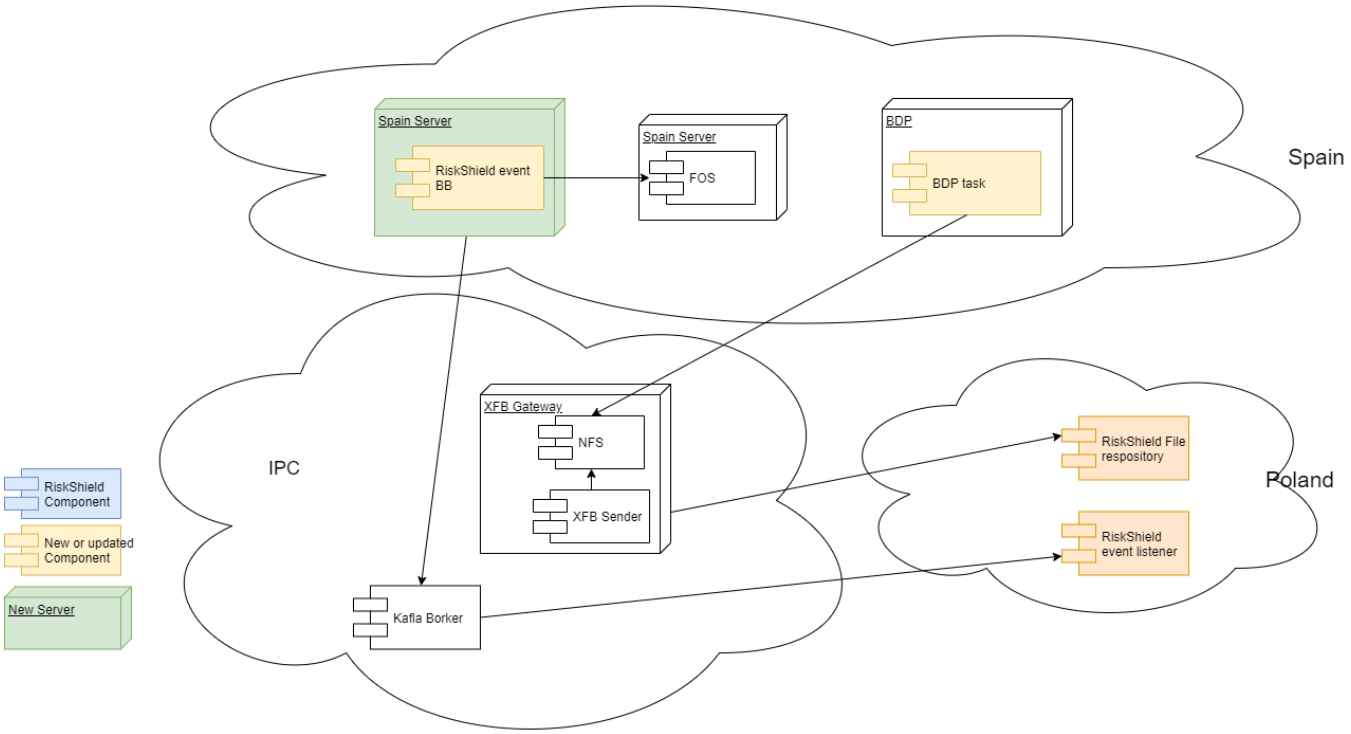


Logical Architecture



Target solution





Still not confirmed (WIP):

- **Not yet confirmed what kind of component is the XBF Sender. It should be a task run by a task planner able to listen a kafka event to trigger the start of the task (send process via XFB).**
- **Physical location of all component is not yet confirmed. Network performance should be considered taking into account connection with Poland, Spain and IPC.**
- **BDP should be confirmed as the execution platform of the UC01, UC08, UC10**
- **Reconcile process should be detailed particularly yo determine what component should calculate the consistency rate and how it is calculated**
- **Missing sequence diagram for 2 use case.**
- **Check if BDP task can retrieve data from Cassadra database**
- **Check the better way to access FOS data (use of Parties BB)**

Non-Functional Requirements

NFR ID	Description	Type	KPIs / Criterias / Condition
NFR001	File sending will be made thanks to XFB Gateway. A service that is provided globally. To do so, sending flow should be describe in Excel template form to request such process. Once authorized, a remote dedicated server file system directory will be available to send by SCP o SFTP the file we want to transfer. Once deposited, XFB sending process is performed automatically.		

Security & Risk

CIA Rating

C	I	A

<< CIA rating determines the applicable security controls to be implemented around the solution, Confidentiality: prevent unauthorised access of the data, Integrity: prevent unauthorised change of the data, Availability: prevent unwanted loss / unavailability of the data >>

User Authentication & Authorization

<< for user-access an identification/authentication procedure needs to be in place, Preferable based on single sign on (SSO) or based on username /password, In some situations (privileged access) multi-factor authentication is required>>

<< specify your types of users, related authentication and authorization methods, directory and number of factors for authentication... >>

Data Security

<< The security controls are applicable for both data-in-rest and data-in-motion and security controls should be applied to both the server-side (backend) and the user-device (frontend) >>

<< Security controls are also applicable for the end-to-end data-exchange between solutions. this include the availability/ resilience of a chain of solutions delivering end-to-end continuity. >>

<< Security Event Monitoring ?? not sure to be mentioned or reminded here, could be part of the security view >>

Minimum Security Controls Identified

ID	Minimum Standard	Control Objective	IT Control	Acceptance Criteria
1				*GIVEN ... AND / OR ... WHEN ... THEN ...*

Workload / Capacity

<< estimates on workload could be provided here >>

Expected Workload Details

Parameters	Forecasted Values

Capacity Suggestions

Component	Capacity

Availability

<< Availability objective : % 99,91 ; Availability Rating : A : ; Business Continuity Requirements 5/8 or 7/24 or daily once or periodical >>

Load Management

<< describe the details of the load balancing, is the system stateful or stateless: describe where in the architecture the state is managed and how >>

<< describe redundancy , clustering , deployment options, scaling options in horizontal or vertical requirements >>

Performance / Fail Over Scenarios

<< mention any critical performance issues, make suggestions on what and how to test the system performance, what type of performance tests suggested like load test, stress test, endurance test, etc..>>

<< service loss: Discuss how solution design handles service losses. Discuss the solution architecture for applications and critical services being unavailable. >>

<< site loss: Although less likely, discuss what will happen and what is required in a site loss scenario - for example if primary datacenter was unavailable. >>

Recovery Objectives

<< What is the recovery strategy and availability for this solution. Worse case scenario, how much data could be lost if the primary system was not available. >>

Application Name	Scenario	Recovery Technique	Recovery Point Objective (hours)	Recovery Time Objective (hours)
<< Application X >>	*e.g. data corruption*	*e.g. backup*	*4*	*4*

Operational Monitoring

<< list any services, logs, identified for monitoring, any performance KPIs for monitoring could be mentioned here also >>

- *<< compliance state management >>*
- *<< logging normalisation aggregation >>*
- *<< performance monitoring >>*
- *<< ?? security event monitoring >>*
- *<< business event monitoring >>*

Useful Links / Standards / Related Readings / References

Description	Link