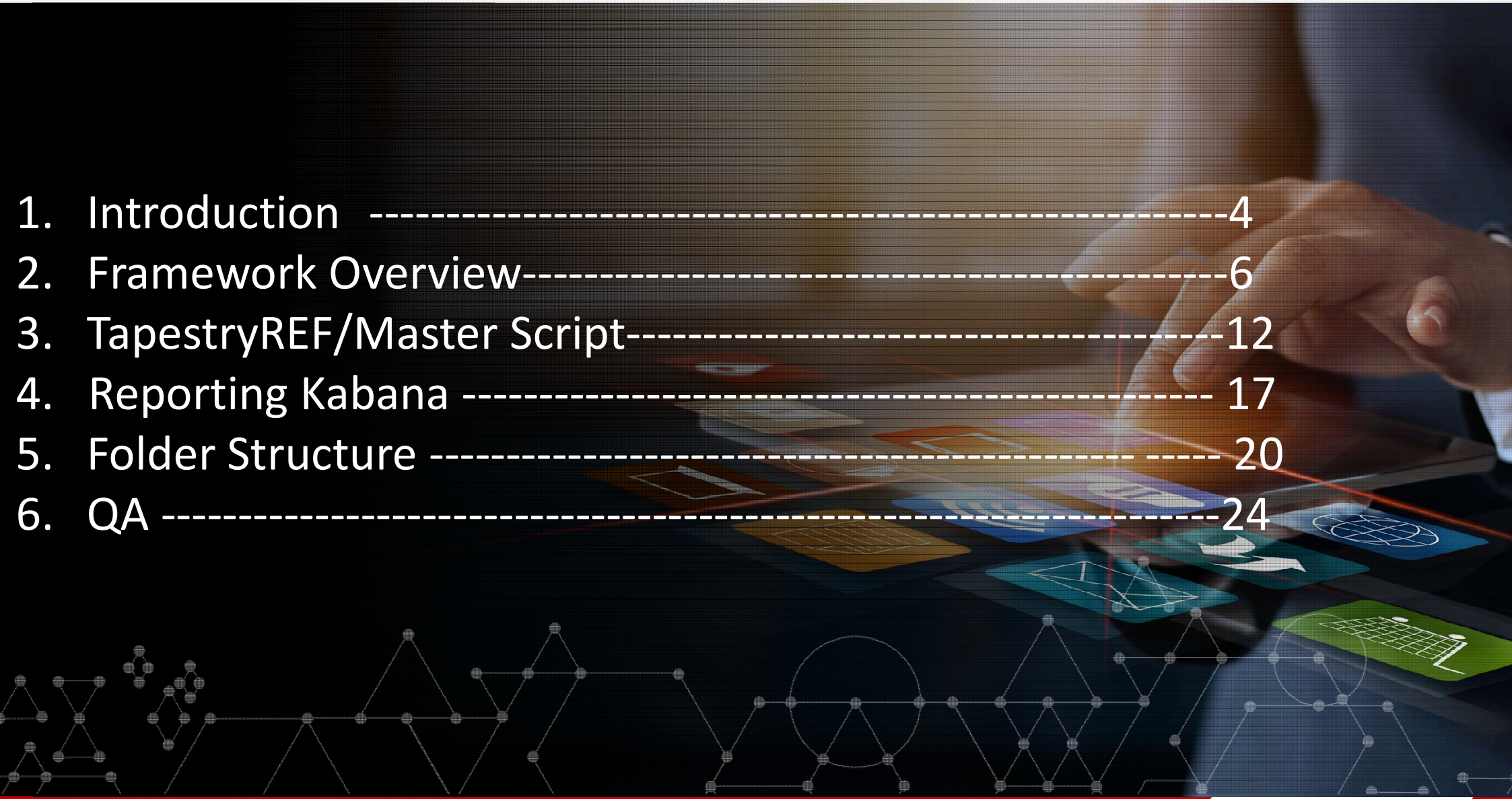
Solution Design Document (SDD)

SoftStone | North America

Presenters and Prepared by : Jinwei Pan (Tech Lead)

Prepared for Tapestry phase 2

ble of Contents



ocument History

Date	Version	Author	Description
4/15/2020	1.0	Jinwei Pan	Initial draft of ssd.
4/20/2020	1.1	Reto Meier	Add additional descriptions
4/23/2020	1.2	Jinwei Pan	Completed ssd with addition content, update the template, add history, table contents, sql explanation, queue explain etc.

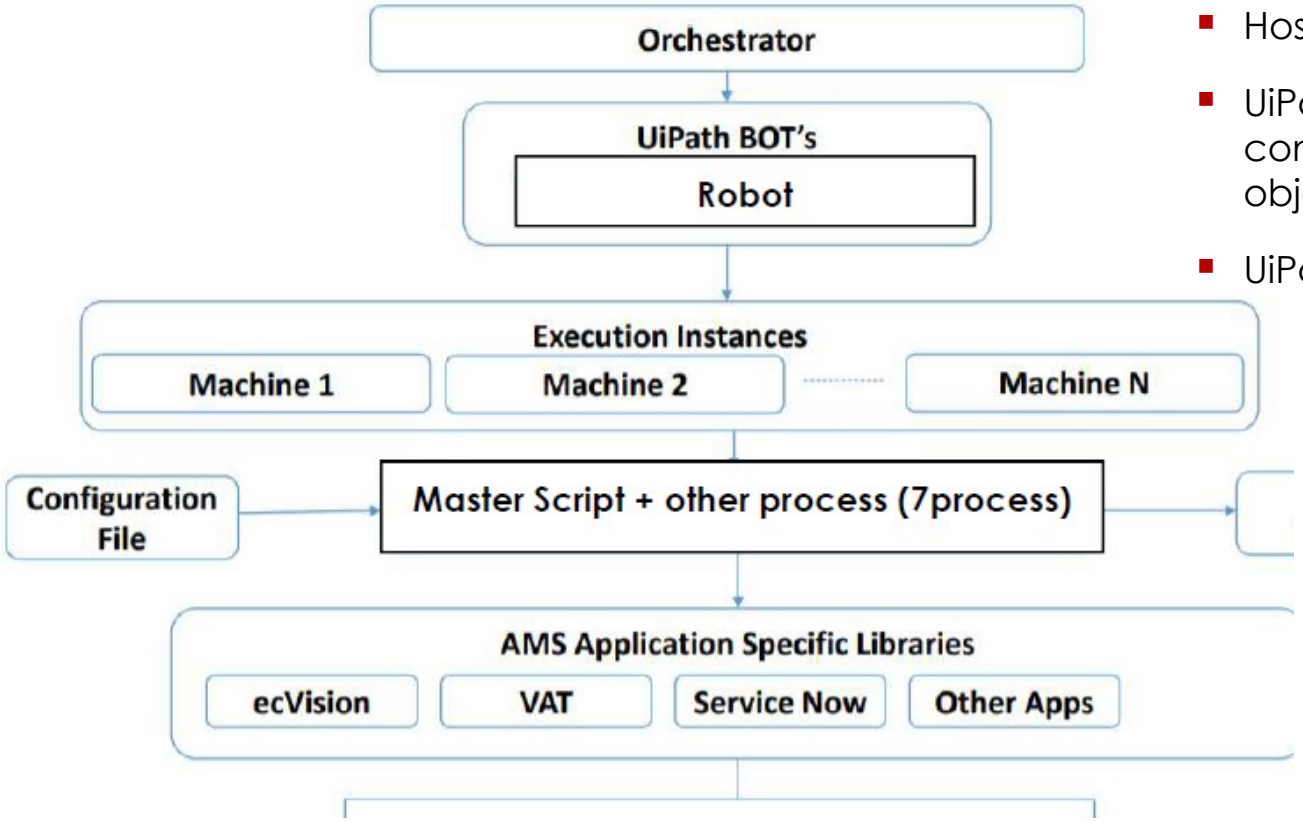
1. Design Introduction

1. Introduction

- This document is filled in by the RPA Solution Architect and RPA developer who automates the business process and reviewed by the RPA Solution Architect. This document is meant to guide RPA COE, IT operations and process owners by providing the snapshots of the automated process details and component as well as to allow developers on having a quick glance at the setup before diving into the code, to troubleshoot or update the workflow.

2. Framework Overview

2.1 Automation Framework



- 3 Studio, Orchestrator, 1 Unattended Robot
- 3 environments (Development, QA/Pre-Production, Production)
- Hosted by Tapestry in its AWS cloud environment
- UiPath REFramework (Robotic Enterprise Framework) comprising master script, application scripts, generic object libraries, configuration files asset.
- UiPath Queues will not be used because:
  - For production there will be only 1 robot to run 7 process, with queue, there will be 8 processes to run in 1 bot, there will be some schedule conflict ex: master script and queue process both would be run multiple times daily
  - Masterscript transaction is based on ticket level, queue is based on use case level.
  - Use case data is stored in attribute excel file and prepared by AMS team. Data could contain human errors after put into the queue
  - Significant development effort

2.2 All Use Cases

ase	Description
ction/SubCollection Request	Process of Collection/SubCollection
/SubClass Request	Process of Class/SubClass
uct Segmentation Request	Process of Product Segmentation
rial Request	Process of Material Request
Group Request	Process of Style Group Request
de Request	Process of Attitude Request
uette Request	Process of Silhouette Request
Rename Request	Process of Style Rename Request
Update Request	Process of DDP Update Request
word Reset	Process of Password Reset
hly Ticket Report	Process of Monthly Ticket Report
ly Ticket Report	Process of Weekly Ticket Report
Access	Process of New users need Access
Off Board	Process of User off Board
n /product refresh	Process of season /product refresh
log	Process of FGPO log

- Most use cases have been implemented and operationalized in several production releases between June and December 2019 (phase 1). Users have created a product backlog with additional requirements to enhance and improve the existing use cases (phase 2). In addition, stakeholders have provided requirements for reporting (operations and business metrics) and email alerts.
- Separately, phase 2 will include resolving existing production defects and other software issues (tec debt).
- Description for each use case will be include in deployment operational documentation handou (DOD)

2.3 All Processes

Production Process Name
TapestryREF
Monthly Ticket Report
WeeklyReportingandDistributing
AccessRequest
Off Board Request
Product Refresh
FGPO Log

- A user request is assigned to the relevant use co in Orchestrator. Orchestrator initiates the assigne use case, which is executed through one of sev (7) defined processes (either through the processes governed by the Master Script or as a standalone process) to create transactions (or jobs). The unattended robot executes the transactions following these defined processes. Master Script comprises multiple use cases, including all attribute requests (e.g., Collection/SubCollection Request)
- .Phase 2 includes the redesign of the Master Scr and incorporating additional use cases to impro development and testing, solution performance and stability, as well as ongoing maintenance.

2.4 Process Structure Organization

Case	Production Process Name
ction/SubCollection Request	TapestryREF
/SubClass Request	TapestryREF
uct Segmentation Request	TapestryREF
erial Request	TapestryREF
Group Request	TapestryREF
ude Request	TapestryREF
uette Request	TapestryREF
Rename Request	TapestryREF
Update Request	TapestryREF
word Reset	TapestryREF
thly Ticket Report	Monthly Ticket Report
kly Ticket Report	Weekly Ticket Report
Access	AccessRequest
Off Board	Off Board Request
on /product refresh	Product Refresh
O log	FGPO Log

- The first 10 use cases will be combined inside t master script-TapestryREF process, because all use case are on ticket/incidents level and hav same schedule requirements, so we can put th one process to reduce the complexities .
- Remaining use cases will be standalone proce to different business need and different sched time requirements.

2.5 All Libraries

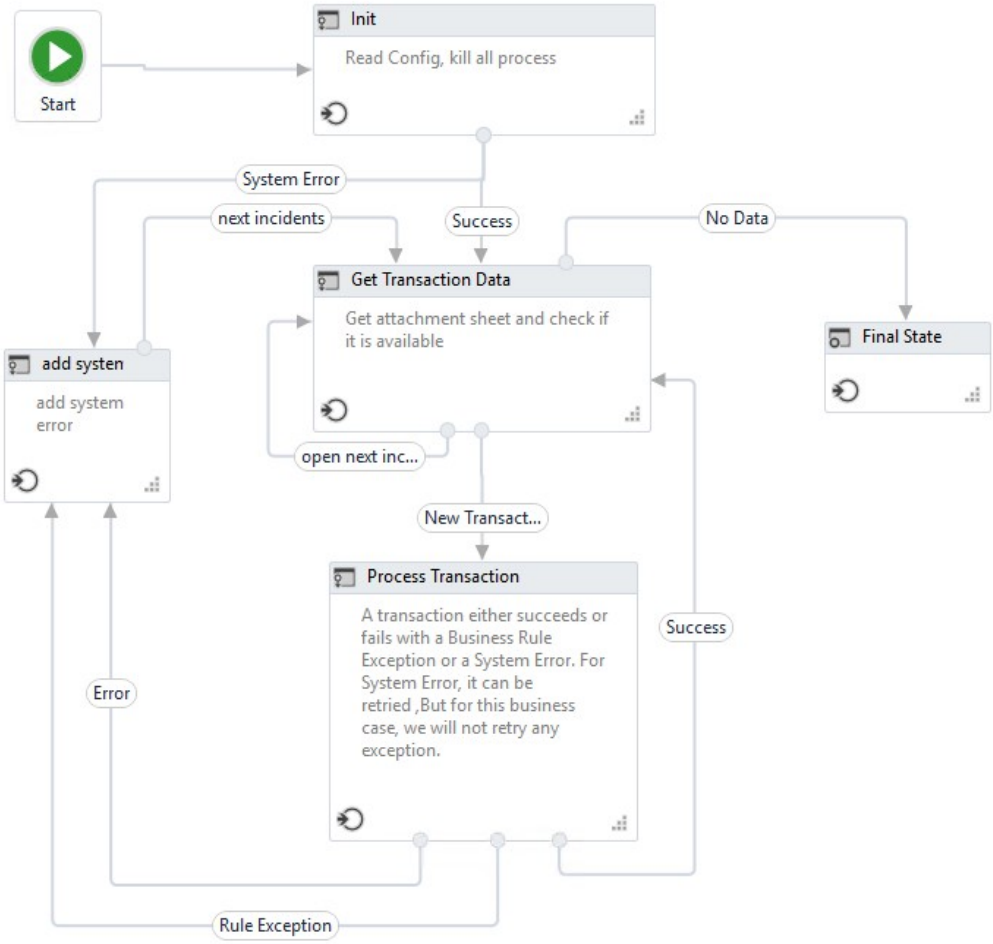
Name	Description
Vision	Steps of login ecVision
cVision	Steps of logout ecVision
NowLogin	Steps of open SN
NowDownloadAttch	Steps of download attribute requrest form attached from SN
NowCloseComment	Steps of add close/work notes and resolve/leave blank for incidents
t	Steps of login VAT
/at	Steps of logout VAT
VisionAdmin	Steps of Login ecVision admin portal
cVisionAdmin	Steps of Logout ecVision admin portal
eCode_3digit	Steps of Generate 3 digit code
eEmail	Steps of sending outlook email
ess	Steps of Kill process by searching for process name
eenShot	Steps of take screenshot

- Generic object libraries comprise objects and components that are re-usable across applic scripts. Most libraries will be stored in the Orchestrator library section and called from th tenant packages section in the UiPath Studio.

3. TapestryREF And Process Flow

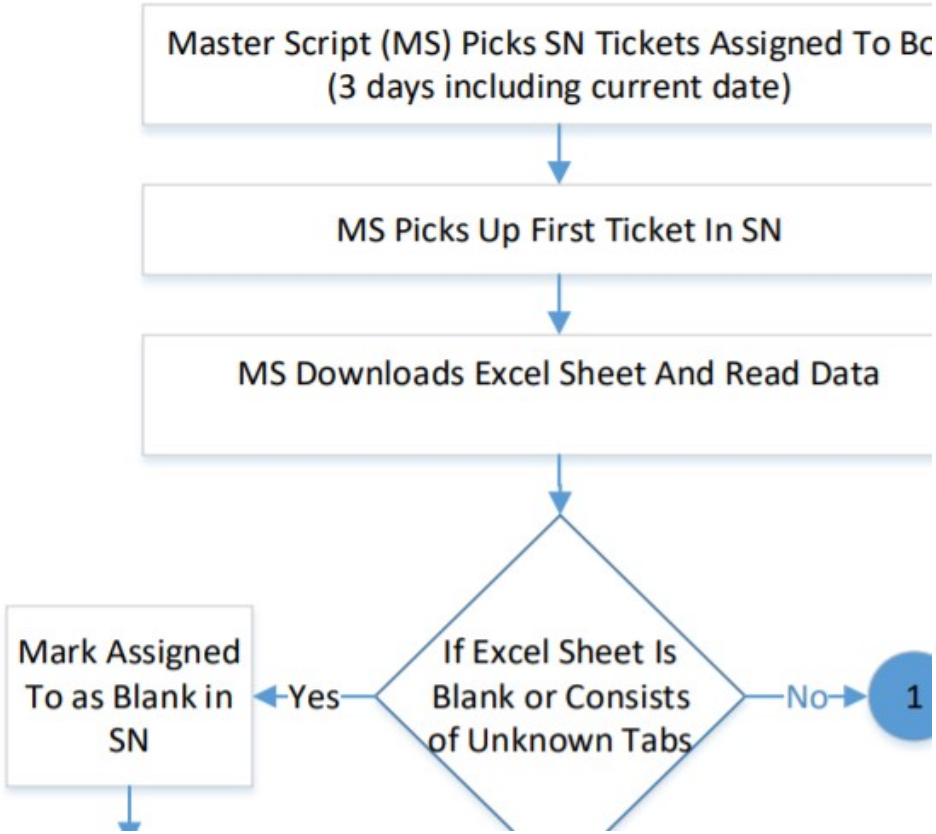
3.1 TapestryREF/ Master Script

State	Description
Start	The start of the workflow.
Init	Initializing system configurations, constants and assets. Kill all process to start the session fresh such as Internet Explorer.
Get Transaction Data	Run query with multi type calls from service now, and get all ticket ID, pick 1st ticket and download attachment
Process Transaction	Process the current ticket following the process list base on the tab section of the attachment sheet flow while handling business and system exceptions, once complete will update status and mark comment in the service now ticket. then loop back to next ticket id to repeat the process transaction
End Process	The end of the workflow when there is no more data to be process or system error.



bove figure depict the main overview of the masterscript in the
EFramework used in TapestryREF. Use case for the automation flow.

3.2 Process Flow



Below steps shows the masterscript TapestryREF automation process flow

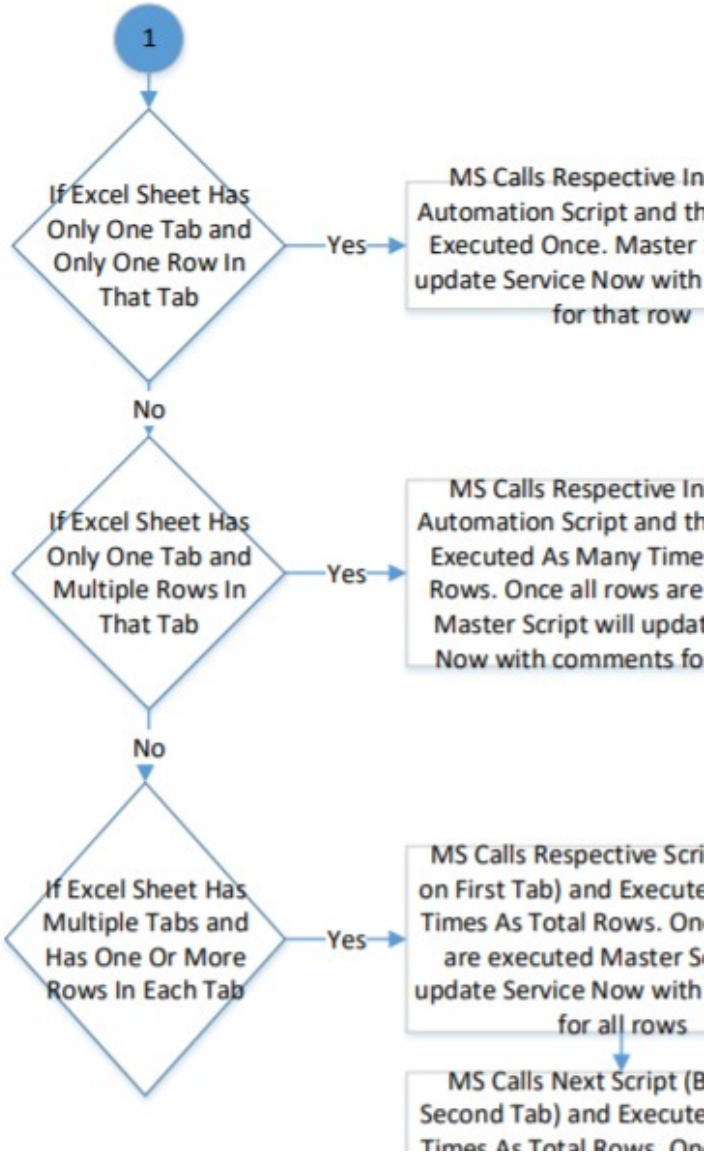
1. Initial Stage:

- Get all settings from orchestrator asset and configuration file

2. Get transaction data state:

- Open ServiceNow and navigate to query link,
- Click first incidents,
- Download the attachment file and check if it is available,
- Save the tab name which is not empty to collection list, the list is transaction Data
- Output each item in the transaction data list as transaction Item

3.2 Process Flow Continue



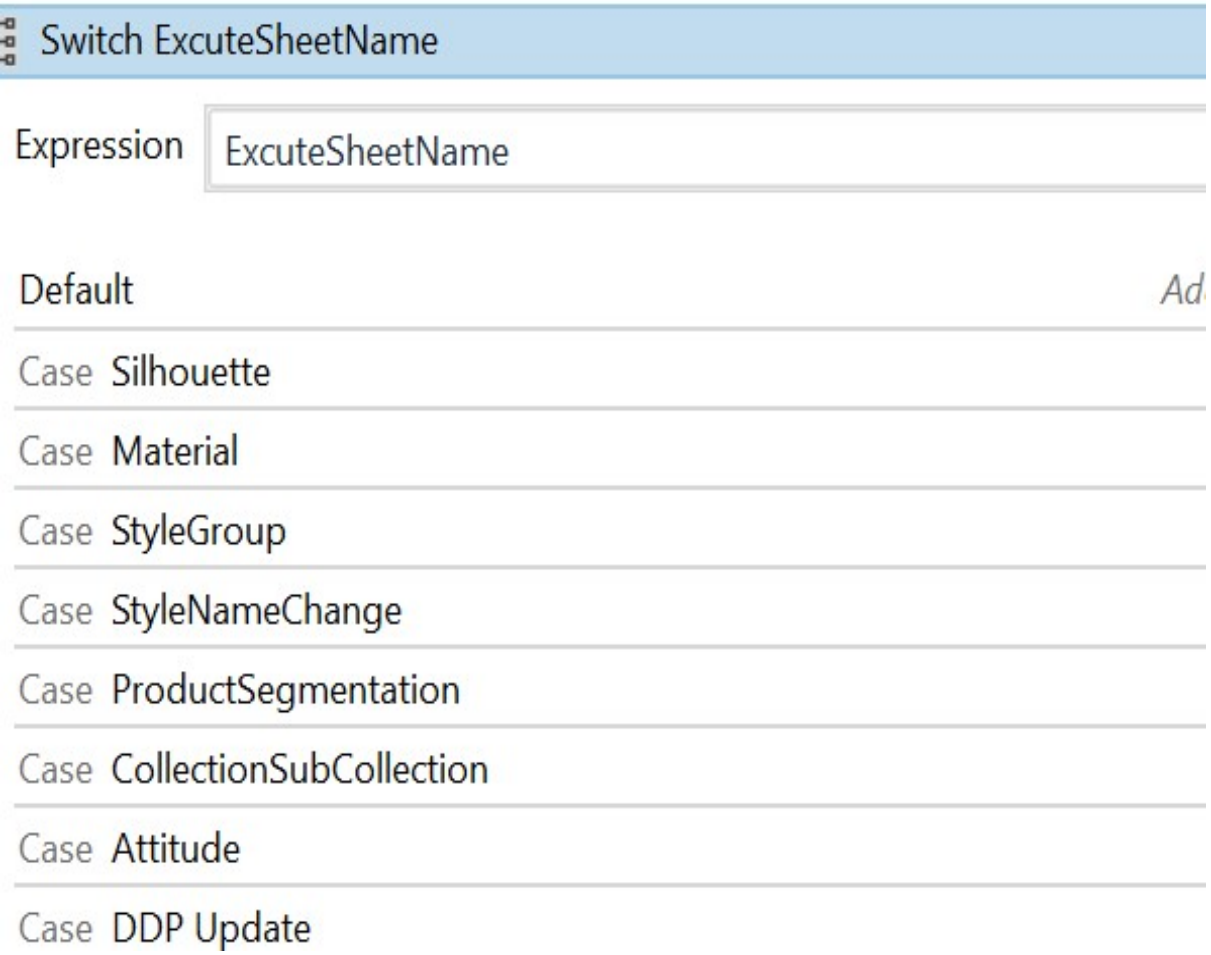
1. Initial Stage:

- Get all settings from Orchestrator asset and configuration file

2. Get transaction data state:

- Open ServiceNow and navigate to query link,
- Click first incidents,
- Download the attachment file and check if it is available,
- Save the tab name which is not empty to collection list, the list is transaction data
- Output each item in the transaction data list as transaction Item
- 3 way to execute the excel sheet
  - one tab one row
  - one tab multiple rows
  - Multiple tabs multiple rows

3.2 Process Flow Continue



3. Process State:

- According to the transaction item switch and invoke different use case workflow, the masterscript will execute each sub use case follow below sequence:

4. Add System Error State:

- If a system exception occurrs, then add system error to ServiceNow website.

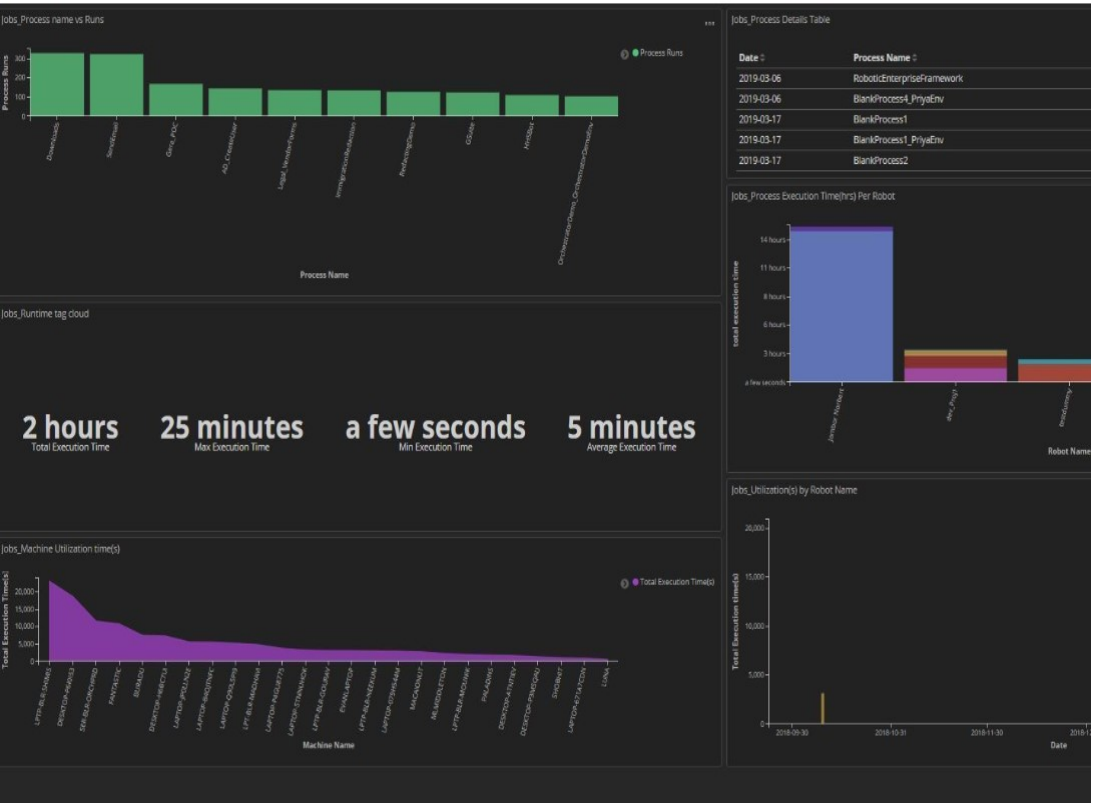
4. Final State:

- Process will loop back to ServiceNow, if no incidents in ServiceNow, then the automation will be finished.

4. Reporting- Kibana Elasticsearch

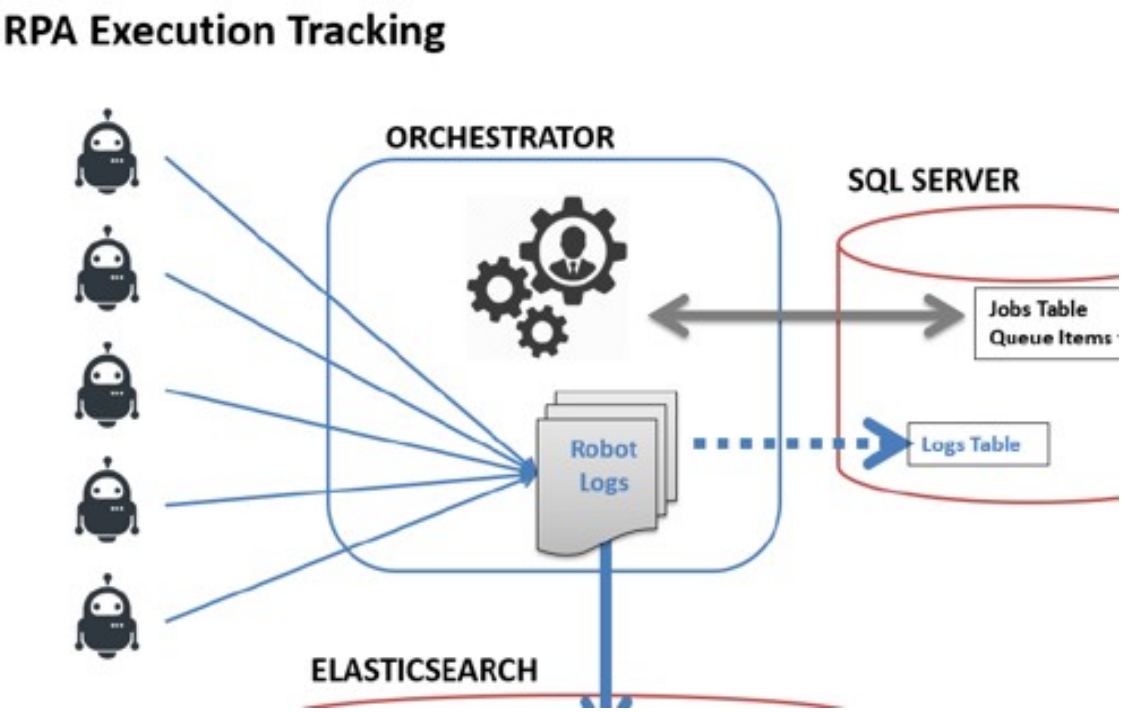
4.1 Reporting- Kibana Elasticsearch

Media



- We recommend to design and build the reporting and en capabilities using Kibana and Elasticsearch to enable bot operations and business metrics, as well as email alerts. UiP supports several reporting solutions, including Kibana, Pow and Tableau, as well as its own Insights module. There are various integration, configuration and licensing considera for each solution. We understand Tapestry prefers Kibana Elasticsearch, which has been enabled in Tapestry's AWS environment for RPA.

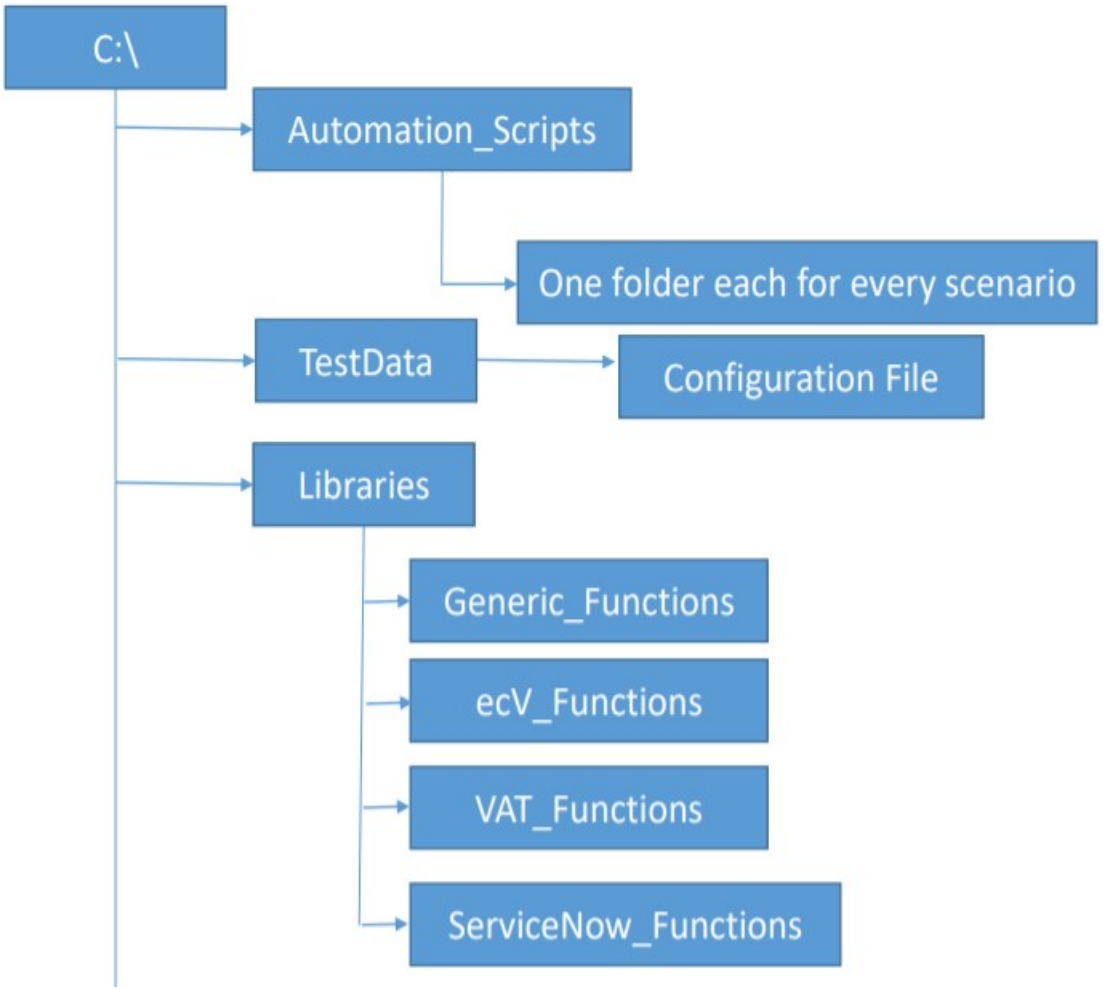
4.2 Kibana Elasticsearch Step up



- SQL server and Kibana Elasticsearch are required.
- Elasticsearch will receive all execution logs from Orchestrat be able to create the reporting dashboard.
- SQL server will restore content tables and detailed applica logs from Orchestrator, such as: all users, assets, robots, machines names, all info level logs for debugging etc.

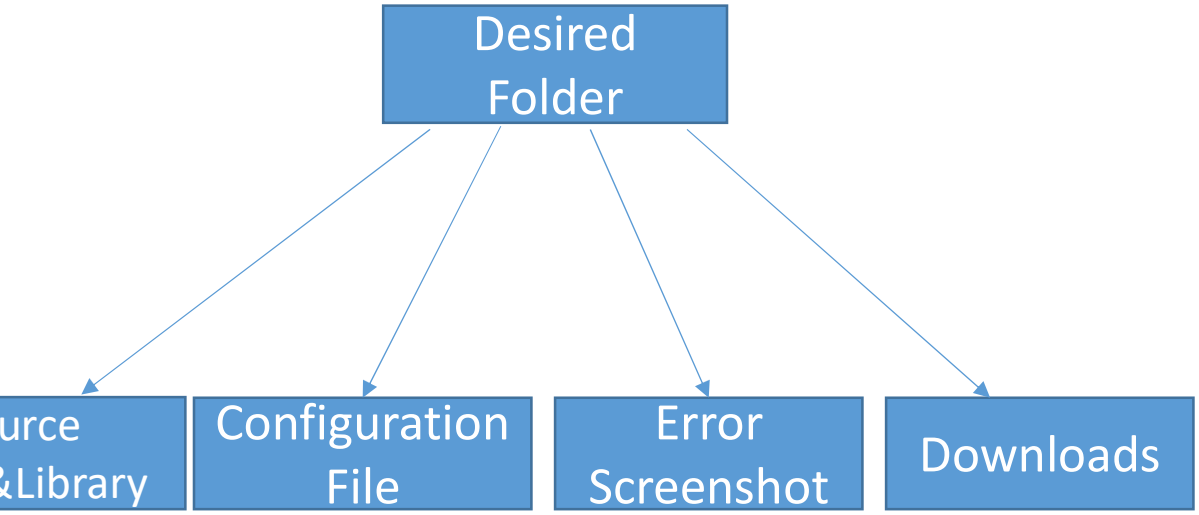
5. All Folder Structure

5.1 Old Folder Structure



- Left is the old Phase 1 folder structure which is not efficient and does not reflect best practice
  - Hardcoded the folder structure location with
    drive
  - Too many folder for each use cases: reframe
    issue
  - Cannot just deploy packages
  - Libraries are not fully utilized and not stored in
    orchestrator, some library can not be shared
    among other use cases
  - Some case rely on download folders to send
    reports out

5.2 New Folder Structure



- Left is the new Phase 2 folder structure
- No more Hardcoded the folder structure location
  - Reframe the folder structure for each use ca
  - No more hardcode
  - Folder structure much simpler
  - Libraries are fully utilized, shared and most ar stored in Orchestrator library
  - Reports and Attachments will send through emails

.3 Deployment Operation Documentation (DOD

DOD Handout (user operations guide) will be provide for each process

- A documentation includes all necessary information of each process such as :
  - Environment requirement
  - Schedule time
  - Assets and libraries file path
  - Invoke process name and description
  - Process descriptions
  - Manual exception handle method
  - How to resume process if process encounters application exception or interruption.

All sending out emails currently using ISS email sever (https://usmail.isoftstone.com) **need to be switched to tapestry's email server**

- Email sever has been updated as an asset and not hardcoded. Can be changed after deployment
- Tapestry Email account
- Tapestry Email password
- Tapestry Email sever name

6. QA