UiPath

Solution Architect

academy.uipath.com

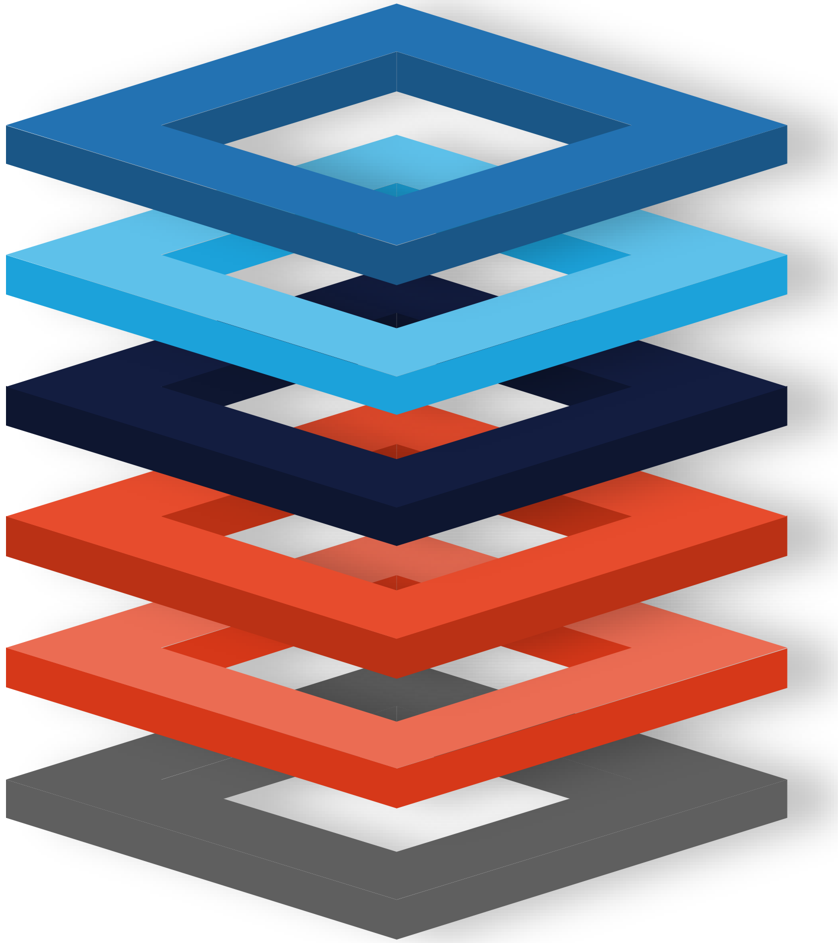
UiPath

Solution Architect

Solution Design

Solution Design

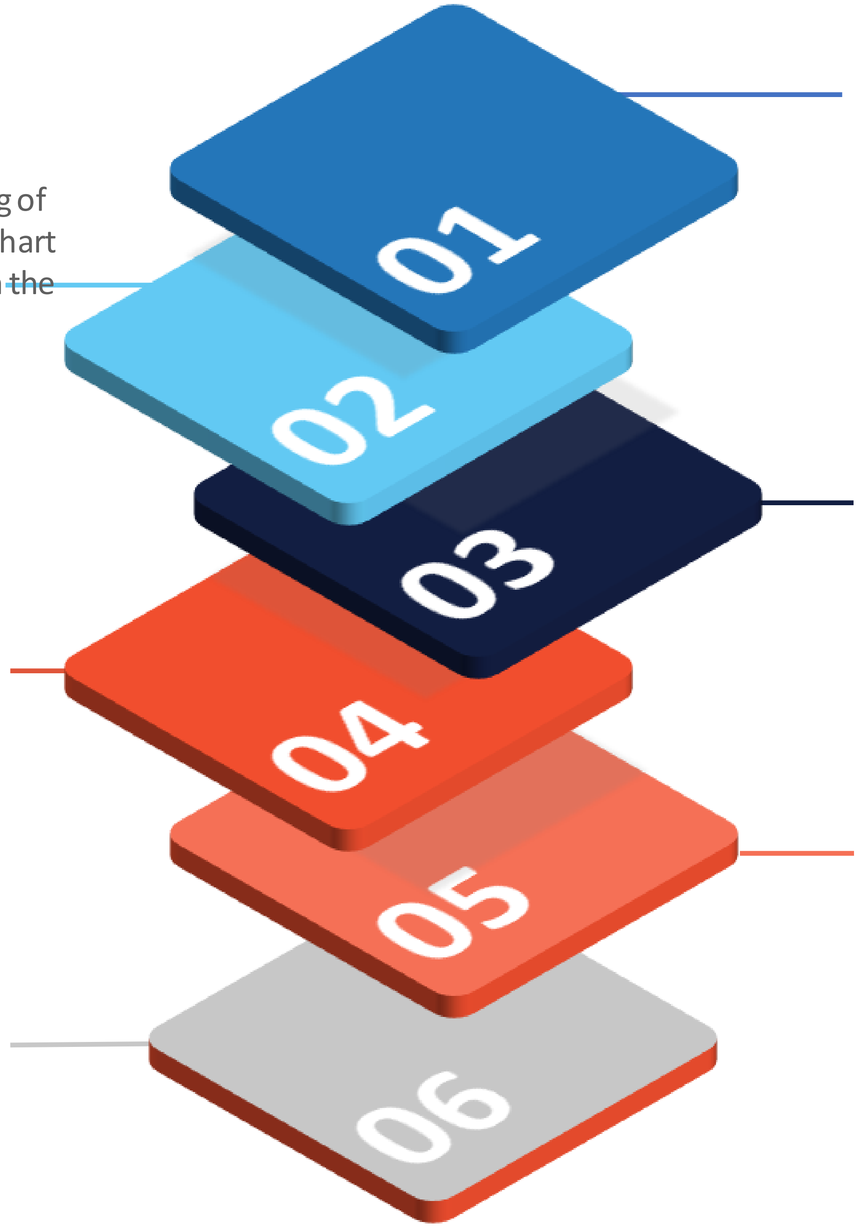
- Attended vs. Unattended
- Input and Output data
- Sub-processes
- Number of robots based on business requirements
- Process and sub-process scheduling
- High level interaction between workflow components



- Source Control project names
- Type and data format – E. g. Queues and queue items specific content
- High level diagram of the automated process (will go to DSD).
- Additional descriptions

RPA Workflow Abstraction

- Abstraction is as important in RPA as it is generally in software development.
- We suggest **six** layers of abstraction
  1. The Framework layer
  2. The Business Process layer
  3. The Services layer
  4. The Application Process layer
  5. The Application Screen layer - highly reusable
  6. The Data layer - reusable
- The SA identifies the workflow components and assigns them to the appropriate level.



1. **Framework layer:** High level State Machine using the REFrameWork with overall exception handling.

2. **Business Process layer:** Modeling of the business process rules - a flowchart integrating other components from the other levels.

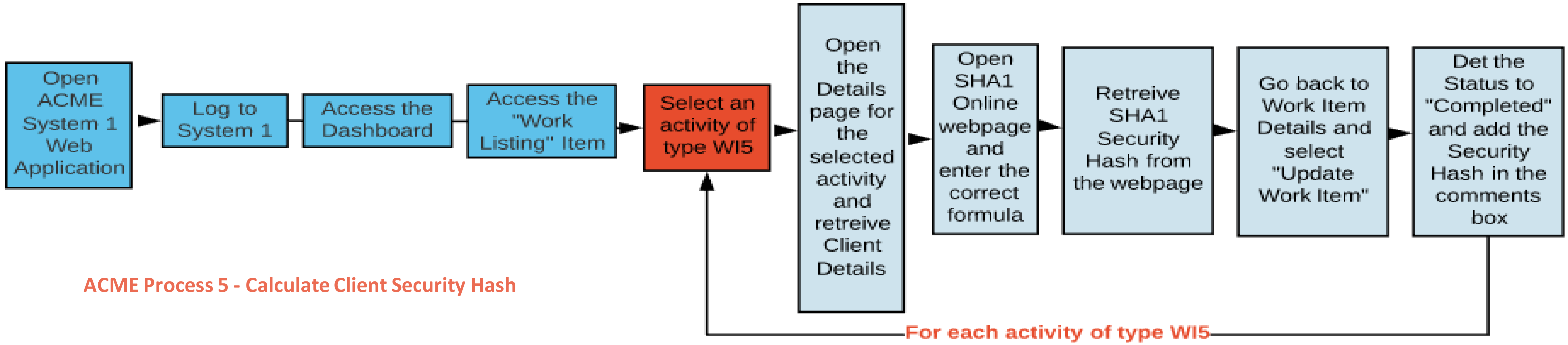
3. **Services layer:** Additional service components with internal exception handling. State machine template.

4. **Application Process layer:** Components working with applications and implementing process rules. The argument names can be related to the business process. This component will navigate, write and extract.

5. **Application Screen layer:** Generic components where arguments are not related to the business process rules. Separate workflows can be created to navigate, input, and extract data.

6. **Data layer:** Interact only with easily accessible (local) data. Orchestrator activities allowed. Not for application interaction.

Workflow Abstraction & Documentation Example


Open ACME System 1 Web Application
Log to System 1
Access the Dashboard
Access the "Work Listing" Item
Select an activity of type WI5
Open the Details page for the selected activity and retreive Client Details
Open SHA1 Online webpage and enter the correct formula
Retreive SHA1 Security Hash from the webpage
Go back to Work Item Details and select "Update Work Item"
Det the Status to "Completed" and add the Security Hash in the comments box
For each activity of type WI5

ACME Process 5 - Calculate Client Security Hash

System	Name	Screen	Description	Arguments	Abstraction Layer	Location	Developer	Status
System1	System1_Login	Desktop	…….	In: Credential – String In: URL – String	App Screen	TFS Project 1 …	Developer 1	Done
System1	System1_Close	Any Page	…….	N/A	App Screen	TFS Project 1 …	Developer 1	Done
System1	System1_ExtractClientInfo	WI Details	…….	Out: Client ID - String Out: ClientName - String Out: ClientCountry - String	App Process	TFS Project 1 …	Developer 2	In progress
System1	System1_ExtractWIsDataTable	WI List	…….	Out: dt_Wis – DataTable	App Screen	TFS Project 1 …	Developer 2	In progress
System1	System1_NavigateTo_Dashboard	Any Page	…….	N/A	App Screen	TFS Project 1 …	Developer 2	In progress
System1	System1_NavigateTo_WIDetails	Any Page	…….	In: WIID – String	App Screen	TFS Project 1 …	Developer 2	In progress
System1	System1_NavigateTo_WorkItems	Dashboard	…….	N/A	App Screen	TFS Project 1 …	Developer 1	In progress
System1	System1_UpdateWorkItem	WI Details	…….	In: Comment – String	App Screen	TFS Project 1 …	Developer 1	In progress
SHA1	SHA1Online_Open	Desktop	…….	In: URL – String	App Screen	TFS Project 1 …	Developer 1	Done
SHA1	SHA1Online_Close	Any Page	…….	N/A	App Screen	TFS Project 1 …	Developer 1	Done
SHA1	SHA1Online_ProcessHash	Process Hash	…….	In: Formula – String Out: HashResult – String	App Screen	TFS Project 1 …	Developer 2	Not started
-	Process	-	…….	In: TransactionItem – DataRow	Business Process	TFS Project 1 …	Developer 2	Not started
-	Framework components	-	…….	-	Framework	TFS Project 1 …	SA1	In progress

Thank You!

UiPath

Solution Architect

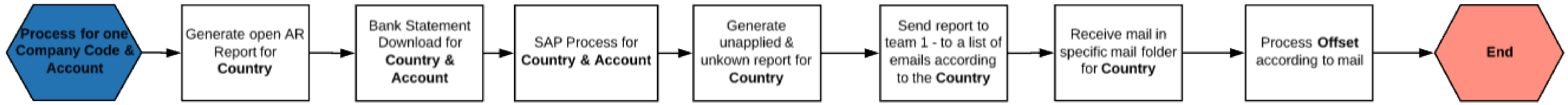
Solution Design

Reusable Components Approach

Local File Storage	Shared File Storage	Shared Components Package	Custom Packaging Method
Store the reusable components in the Source Control System. Sync files in shared location. Add the network path to the Library in Studio.(RECOMMENDED).	Use a file share location to store the reusable components and invoke them remotely.	Create one package containing shared components and distribute it using the Orchestrator embedded provisioning mechanism.	Use a third party solution to create packages that also provision the reusable components.
PROS: ✓ Easiest to implement ✓ Most secure **CONS:** ▪ In case a reusable needs to be changed, manual re-publishing and re-deploying are required	**PROS**: ✓ Easy to implement ✓ Calling by reference **CONS:** ▪ In case of network failure, the robots are not able to run ▪ The robots can run more slowly due to network latency (or even trigger exceptions) ▪ Security risk (access to shared folder)	**PROS:** ✓ Calling by reference ✓ Version control **CONS:** ▪ Harder to implement ▪ Project path needs configuration	**PROS:** ✓ An alternative to the Shared Components approach **CONS:** ▪ The most complex approach and the hardest to implement and maintain ▪ Dependent on third party components

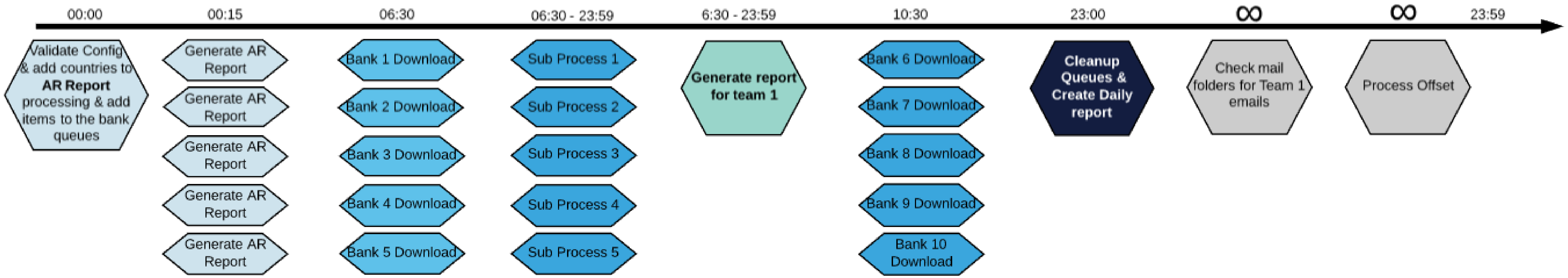
Overall Solution Diagram – Example 1

Basic steps for each Application



Process for one Company Code & Account → Generate open AR Report for **Country** → Bank Statement Download for **Country & Account** → SAP Process for **Country & Account** → Generate unapplied & unkown report for **Country** → Send report to team 1 - to a list of emails according to the **Country** → Receive mail in specific mail folder for **Country** → Process **Offset** according to mail → **End**

Process Scheduling



00:00
Validate Config & add countries to **AR Report** processing & add items to the bank queues

00:15
Generate AR Report
Generate AR Report
Generate AR Report
Generate AR Report
Generate AR Report

06:30
Bank 1 Download
Bank 2 Download
Bank 3 Download
Bank 4 Download
Bank 5 Download

06:30 - 23:59
Sub Process 1
Sub Process 2
Sub Process 3
Sub Process 4
Sub Process 5

6:30 - 23:59
Generate report for team 1

10:30
Bank 6 Download
Bank 7 Download
Bank 8 Download
Bank 9 Download
Bank 10 Download

23:00
Cleanup Queues & Create Daily report

∞
Check mail folders for Team 1 emails

∞
Process Offset

23:59

Robot 1 Dispatcher

Robot 2,3,4,5,6

Robot 7,8,9,10,11

Robot 2,3,4,5,6
(can be scaled to any multiple of 5)

Robot 1

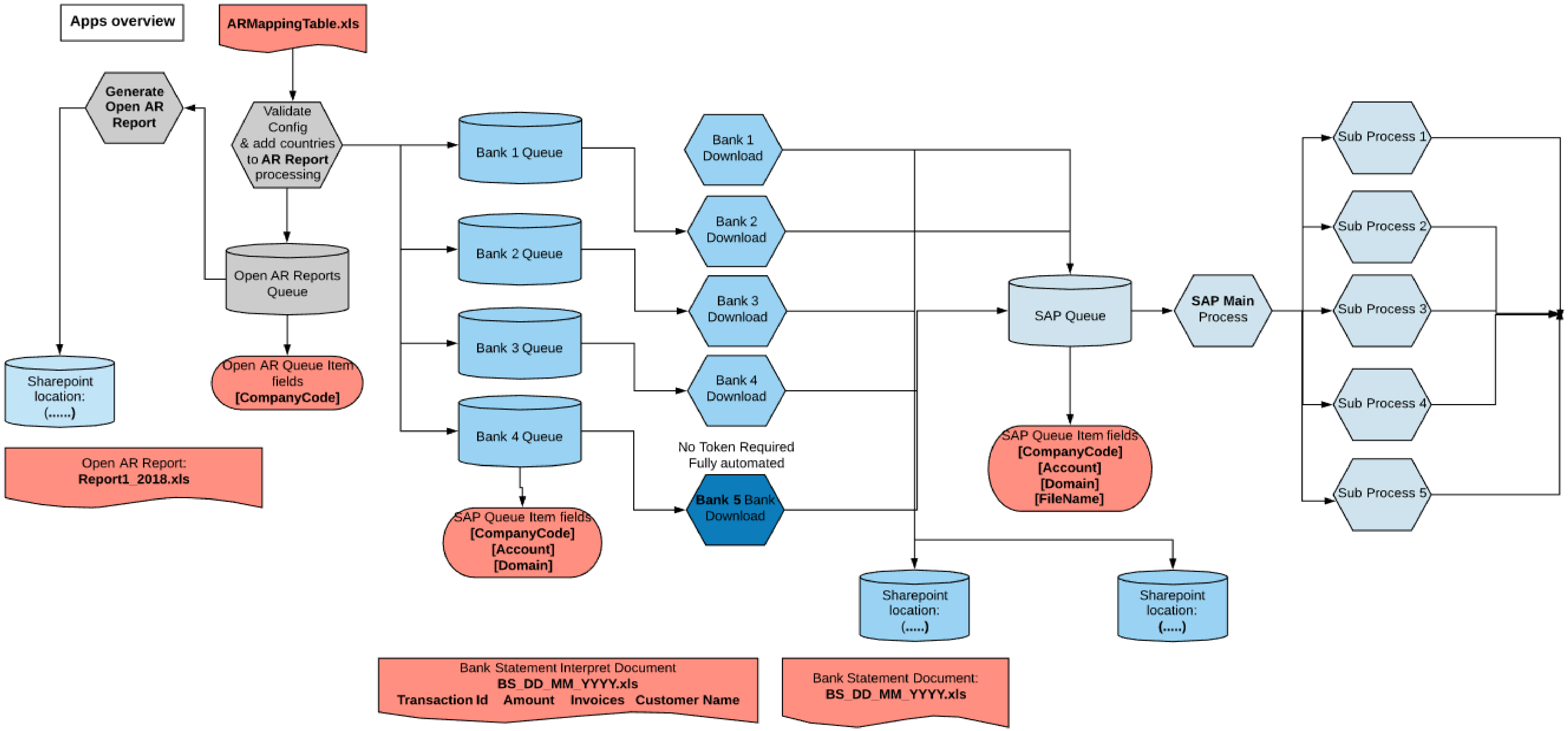
Robot 7,8,9,10,11

Robot 1

Robot 12

Robot 13
(can be scaled to any number)

Overall Solution Diagram – Example 2



Apps overview

ARMappingTable.xls

Generate Open AR Report

Validate Config & add countries to **AR Report** processing

Open AR Reports Queue

Open AR Queue Item fields **[CompanyCode]**

Sharepoint location: (……)

Open AR Report: **Report1_2018.xls**

Bank 1 Queue

Bank 2 Queue

Bank 3 Queue

Bank 4 Queue

SAP Queue Item fields **[CompanyCode] [Account] [Domain]**

Bank 1 Download

Bank 2 Download

Bank 3 Download

Bank 4 Download

No Token Required
Fully automated

Bank 5 Bank Download

SAP Queue

SAP Queue Item fields **[CompanyCode] [Account] [Domain] [FileName]**

SAP Main Process

Sub Process 1

Sub Process 2

Sub Process 3

Sub Process 4

Sub Process 5

Sharepoint location: **(…..)**

Sharepoint location: **(…..)**

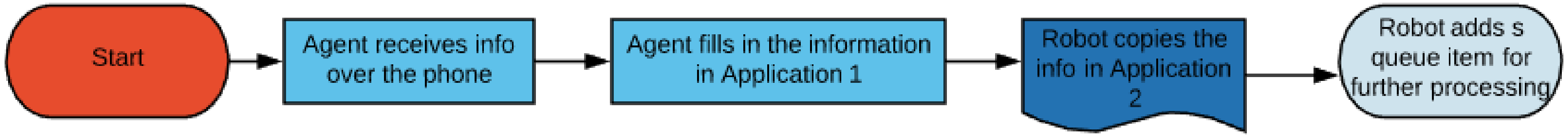
Bank Statement Interpret Document
BS_DD_MM_YYYY.xls
Transaction Id Amount Invoices Customer Name

Bank Statement Document:
BS_DD_MM_YYYY.xls

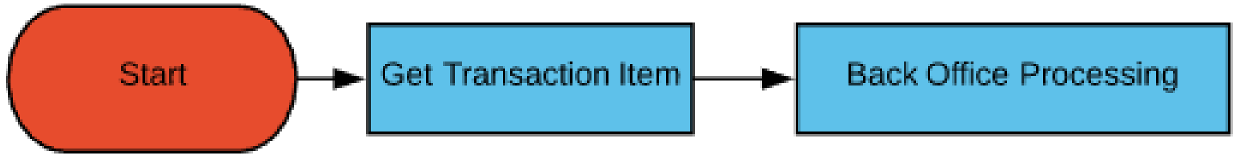
Attended vs Unattended in solution design

- Front office processes can make use of back office processing
- Data sync between processes using Orchestrator queues

Attended


Start
Agent receives info over the phone
Agent fills in the information in Application 1
Robot copies the info in Application 2
Robot adds s queue item for further processing

Unattended



BENEFITS

1 Fast execution for the Attended robot

2 Scalable

UiPath

Thank You!

Solution Architect

Build RPA

RPA Build – Developing workflows

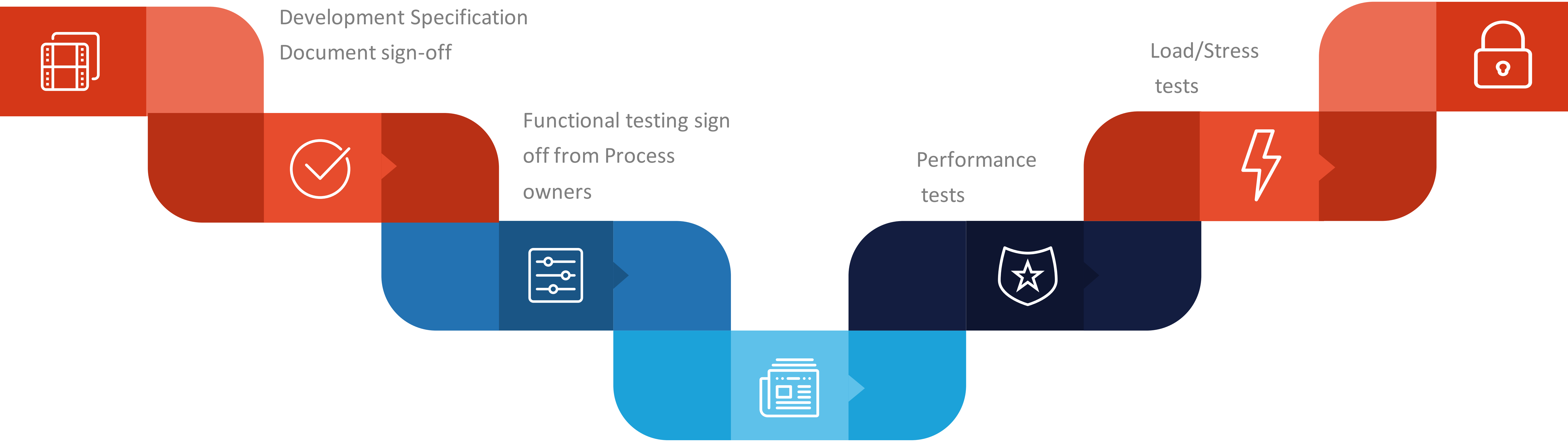
- Review and sign off extra requirement list after PDD sign-off
- Constant mentoring and increasing the capabilities of the RPA Developers
- Troubleshooting and debugging
- Logging and reporting
- Source control owner of the solution

- Performance improvements and quality check
- Managing risk identification and risk mitigation strategies
- Solving complex technical issues
- Assisting in Unit and Functional testing

Solution Architect

Test RPA

RPA Test – Quality assurance



Code review audit

Development Specification Document sign-off

Functional testing sign off from Process owners

REFrameWork automated test framework

Performance tests

Load/Stress tests

Robustness tests

RPA Test – Quality assurance

Code review

- Thorough code review – activity level
- Follow the agreed standard and guidelines
- Naming strategy
- Optimal techniques
- Hard coded values vs Config
- Assets vs config files
- Duplicated code
- Scalability and Maintainability
- Clean code

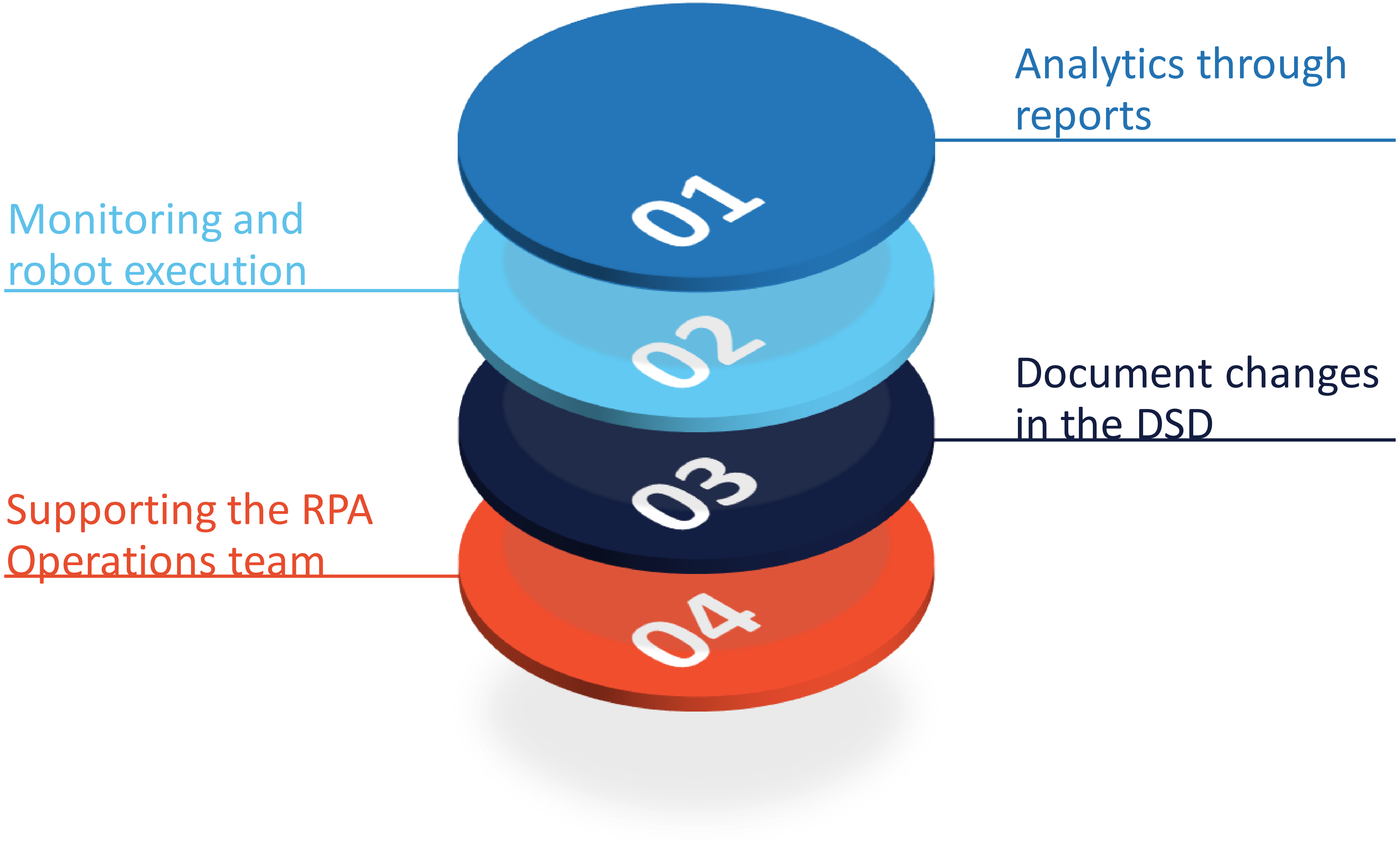
Audit

- Credentials for applications usage
- Scope limitation of credentials
- Check against exposing sensitive information (send mail, save file, etc.)
- Check against modifying configuration parameters

Solution Architect

Sustain RPA

RPA Sustain – Hypercare (Warranty)



Analytics through reports

Monitoring and robot execution

Document changes in the DSD

Supporting the RPA Operations team

Thank You!