

Functional Requirements Specification

Customer Onboarding

|  |  |
| --- | --- |
| Authors: | Phillip Caruthers, Samuel Hagadorn, Bernice Templeman, Malykai Mejia |
| Approved By: | Prakash Agrawal |
| Version: | v.2 |
| Date: | December 1, 2021 |
| Approved (Y/N): | Yes |

Table of Contents

[1.0 Introduction 3](#_Toc89241576)

[1.1 Background 3](#_Toc89241577)

[1.2 Purpose 3](#_Toc89241578)

[1.3 Scope 3](#_Toc89241579)

[2.0 Onboarding Process 5](#_Toc89241580)

[2.1 Process 5](#_Toc89241581)

[2.2 Domain Specification 5](#_Toc89241582)

[2.3 Transform Maps 5](#_Toc89241583)

[2.4 Service Level Agreements 6](#_Toc89241584)

[3.0 Design 7](#_Toc89241585)

[3.1 Locations 7](#_Toc89241586)

[3.1.1 Import Table Fields 7](#_Toc89241587)

[3.2 Departments 7](#_Toc89241588)

[3.2.1 Import Table Fields 7](#_Toc89241589)

[3.3 Cost Centers 7](#_Toc89241590)

[3.3.1 Import Table Fields 7](#_Toc89241591)

[3.4 Business Units 8](#_Toc89241592)

[3.4.1 Import Table Fields 8](#_Toc89241593)

[3.5 Users 8](#_Toc89241594)

[3.5.1 Import Table Fields 8](#_Toc89241595)

[3.6 Groups 8](#_Toc89241596)

[3.6.1 Import Table Fields 8](#_Toc89241597)

[3.7 Group Members 8](#_Toc89241598)

[3.7.1 Import Table Fields 9](#_Toc89241599)

[3.8 User Roles 9](#_Toc89241600)

[3.8.1 Import Table Fields 9](#_Toc89241601)

[3.9 Windows Server 9](#_Toc89241602)

[3.9.1 Import Table Fields 9](#_Toc89241603)

[3.10 Linux Server 9](#_Toc89241604)

[3.10.1 Import Table Fields 9](#_Toc89241605)

[3.11 Unix Server 10](#_Toc89241606)

[3.11.1 Import Table Fields 10](#_Toc89241607)

[3.12 Oracle / SQL 10](#_Toc89241608)

[3.12.1 Import Table Fields 10](#_Toc89241609)

[4.0 Service Level Agreements 11](#_Toc89241610)

[4.1 Response Conditions: 11](#_Toc89241611)

[4.2 Resolution Conditions: 11](#_Toc89241612)

[4.3 Holiday Schedule 12](#_Toc89241613)

[5.0 Summary 14](#_Toc89241614)

# 1.0 Introduction

## 1.1 Background

Systems are a critical component of efficient business operations. Without them, businesses run the risk of losing productivity as a result of missed work, breached deadlines or duplicated efforts. By creating a system for tasks that need to be repeated, such as onboarding a new customer, businesses can ensure that all the data is entered completely and timely, maximizing time and resources for higher productivity and ultimately higher profits.

## 1.2 Purpose

This Functional Requirements Specification document will develop a repeatable step-by-step process to bring a new customer into the ServiceNow environment. The onboarding process defined here will help ensure that new customers are set up, in the separate domains (in a multi-tenant instance), and with their data imported so that the customer can immediately begin utilizing ServiceNow.

The following system tables will be updated via transform maps:

|  |  |
| --- | --- |
| **Name** | **System Table** |
| Locations | cmn\_location |
| Departments | cmn\_department |
| Cost Centers | cmn\_cost\_center |
| Business Units | business\_unit |
| Users | sys\_user |
| Groups | sys\_user\_group |
| Group Members | sys\_user\_grmember |
| User Roles | sys\_user\_has\_role |
| Windows | cmdb\_ci\_win\_server |
| Linux | cmdb\_ci\_linux\_server |
| Unix | cmdb\_ci\_unix\_server |
| Oracle/SQL | cmdb\_ci\_database |

## 1.3 Scope

The successful update of the preceding tables constitutes onboarding of a new customer. Additionally, Service Level Agreements (SLAs) and required schedules will be created as received from the customer and re-iterated in Section 4.

Finally, the customer specified the following number of entries be created for each table listed:

Location – 10

Department – 10

Cost Center – 10

Business Units – 5

Users - 20

Groups – 5

Group Members - 2 per group (minimum)

User Roles - at least 15 users should have roles

SLA - 10 (5 Response and 5 Resolution)

cmdb\_ci\_win\_server - 5 items

cmdb\_ci\_linux\_server - 5 items

cmdb\_ci\_unix\_server - 5 items

cmdb\_ci\_database - 10 items (5 Oracle / 5 SQL)

# 2.0 Onboarding Process

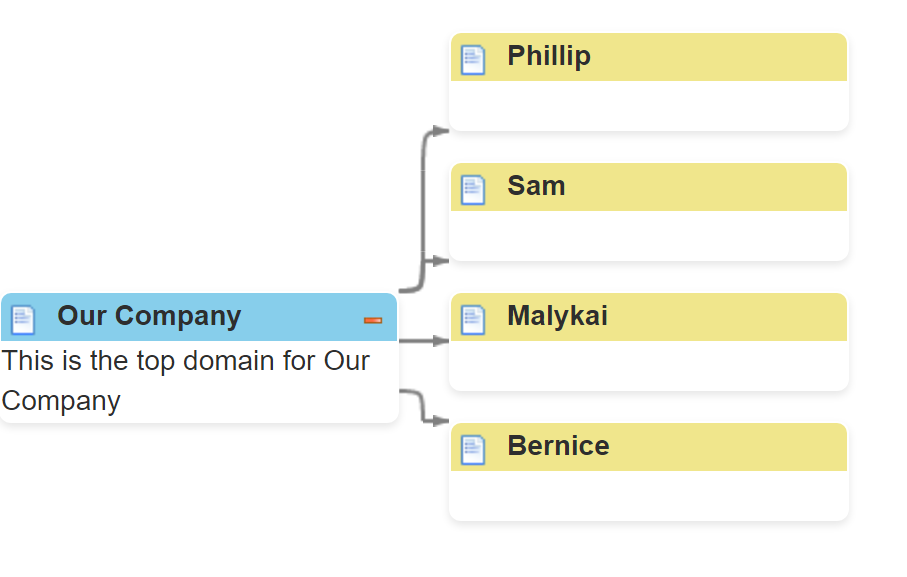
## 2.1 Process

The following process is to be followed to onboard a new customer:

1. Domain Specification: Create customer domain as child of our client (parent/top) domain
2. Transform Map: Create transform maps for each system table
3. Create Service Level Agreements (SLAs)

## 2.2 Domain Specification

The domain hierarchy contains the top/parent domain (Team 3) and child domains (Phillip, Sam, Malykai, Bernice), each will contain the customer to be onboarded into the ServiceNow platform.



Domain Map

## 2.3 Transform Maps

This process should incorporate the use of multiple customized transform maps. These maps will take incoming information from external sources and translate them in a manner where they can be read and defined by the parent instance. These translations begin after the incoming data is loaded into an import set, when the map is originally constructed. It links fields already found in ServiceNow tables to parameters defined in the new data. Within each transform map, specific fields will be coalesced to ensure the uniqueness of each input. Coalescing fields allows information to be correctly read, updated and created based on how it interacts with the predefined table information. In this process, transform maps will be constructed by using fields from already existing tables and specifying the correlating system table field. Once sent through the transforms, the new information will be accurately inserted into the system and available for use.

## 2.4 Service Level Agreements

Service Level Agreements are the timeframe within which a specified task is required to be completed. Failure to complete the task within the timeframe will result in a breach of contract, which will trigger reviews for the assigned user.

# 3.0 Design

In order to import customer data, the following system tables will be used in transform maps to correctly associate customer data with system fields. For each table, the specified fields are required either by the system or for meaningful input into the system.

Note: System Mandatory fields are indicated in red. **The remaining fields indicated for each table are the *minimum* recommended datum for a healthy record system.**

## 3.1 Locations

Table: cmn\_location

### 3.1.1 Import Table Fields

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Label** | **Field Name** | **Data Type** | **Reference To** | **Field Referenced** | **Coalesce** | **Choice Action** |
| Name | name | String |  |  | True |  |
| Company | company | Reference | sys\_company | Name | True | Reject |
| Street | street | String |  |  | True |  |
| City | city | String |  |  | True |  |
| State | state | String |  |  | True |  |
| Zip | zip | String |  |  | True |  |
| Contact | contact | Reference | sys\_user | Email | True | Ignore |
| Phone | phone | String |  |  | True |  |

## 3.2 Departments

Table: cmn\_department

### 3.2.1 Import Table Fields

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Label** | **Field Name** | **Data Type** | **Reference To** | **Field Referenced** | **Coalesce** | **Choice Action** |
| Name | name | String |  |  | True |  |
| Company | company | Reference | sys\_company | Name | True | Reject |
| ID | id | String |  |  |  |  |
| Cost Center | cost\_center | Reference | cmn\_cost\_center | Name |  | Ignore |

## 3.3 Cost Centers

Table: cmn\_cost\_center

### 3.3.1 Import Table Fields

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Label** | **Field Name** | **Data Type** | **Reference To** | **Field Referenced** | **Coalesce** | **Choice Action** |
| Name | name | String |  |  | True |  |
| Location | location | Reference | cmn\_location | Name | False | Ignore |
| Company | company | Reference | sys\_company | Name | False | Reject |
| Manager | manager | Reference | sys\_user | Email |  | Reject |

## 3.4 Business Units

Table: business\_unit

### 3.4.1 Import Table Fields

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Label** | **Field Name** | **Data Type** | **Reference To** | **Field Referenced** | **Coalesce** | **Choice Action** |
| Name | name | String |  |  | True |  |
| Company | company | Reference | sys\_company | Name | False | Reject |

## 3.5 Users

Table: sys\_user

### 3.5.1 Import Table Fields

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Label** | **Field Name** | **Data Type** | **Reference To** | **Field Referenced** | **Coalesce** | **Choice Action** |
| User ID | user\_id | String |  |  | True |  |
| First Name | first\_name | String |  |  | False |  |
| Last Name | last\_name | String |  |  | False |  |
| Company | company | Reference | sys\_company | Name | True | Reject |
| Email | email | Email |  |  | True |  |
| Department | department | Reference | cmn\_department | Name | False | Ignore |
| Phone | phone | Phone Number |  |  | False |  |
| Location | location | Reference | cmn\_location | Name | True | Ignore |
| Password | user\_password | Password |  |  |  |  |
| Password Needs Reset | password\_needs\_reset | Yes/No |  |  |  |  |

## 3.6 Groups

Table: sys\_user\_group

### 3.6.1 Import Table Fields

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Label** | **Field Name** | **Data Type** | **Reference To** | **Field Referenced** | **Coalesce** | **Choice Action** |
| Name | name | String |  |  | True |  |
| Company | company | Reference | sys\_company | Name | False | Reject |
| Manager | manager | Reference | sys\_user | Email | False | Ignore |

## 3.7 Group Members

Table: sys\_user\_grmember

### 3.7.1 Import Table Fields

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Label** | **Field Name** | **Data Type** | **Reference To** | **Field Referenced** | **Coalesce** | **Choice Action** |
| User | user | Reference | sys\_user | Email | True | Ignore |
| Group | group | Reference | sys\_user\_group | Group | True | Ignore |
| Company | company | Reference | sys\_company | Name | False | Reject |

## 3.8 User Roles

Table: sys\_user\_has\_role

### 3.8.1 Import Table Fields

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Label** | **Field Name** | **Data Type** | **Reference To** | **Field Referenced** | **Coalesce** | **Choice Action** |
| Name | name | Reference | sys\_user | Email | True | Ignore |
| Role | role | Reference | sys\_user\_role | Name | True | Ignore |

## 3.9 Windows Server

Table: cmdb\_ci\_win\_server

### 3.9.1 Import Table Fields

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Label** | **Field Name** | **Data Type** | **Reference To** | **Field Referenced** | **Coalesce** | **Choice Action** |
| Name | name | String |  |  | True |  |
| Company | company | Reference | sys\_company | Name | True | Ignore |
| Location | location | Reference | cmn\_location | Name | False | Ignore |
| Serial Number | serial\_number | String |  |  | True |  |
| Asset Tag | asset\_tag | String |  |  | True |  |
| Class | class | String |  |  |  |  |

## 3.10 Linux Server

Table: cmdb\_ci\_linux\_server

### 3.10.1 Import Table Fields

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Label** | **Field Name** | **Data Type** | **Reference To** | **Field Referenced** | **Coalesce** | **Choice Action** |
| Name | name | String |  |  | True |  |
| Company | company | Reference | sys\_company | Name | True | Ignore |
| Location | location | Reference | cmn\_location | Name | False | Ignore |
| Serial Number | serial\_number | String |  |  | True |  |
| Asset Tag | asset\_tag | String |  |  | True |  |
| Class | class | String |  |  |  |  |

## 3.11 Unix Server

Table: cmdb\_ci\_unix\_server

### 3.11.1 Import Table Fields

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Label** | **Field Name** | **Data Type** | **Reference To** | **Field Referenced** | **Coalesce** | **Choice Action** |
| Name | name | String |  |  | True |  |
| Company | company | Reference | sys\_company | Name | True | Ignore |
| Location | location | Reference | cmn\_location | Name | False | Ignore |
| Serial Number | serial\_number | String |  |  | True |  |
| Asset Tag | asset\_tag | String |  |  | True |  |
| Class | class | String |  |  |  |  |

## 3.12 Oracle / SQL

Table: cmdb\_ci\_database

### 3.12.1 Import Table Fields

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Label** | **Field Name** | **Data Type** | **Reference To** | **Field Referenced** | **Coalesce** | **Choice Action** |
| Name | name | String |  |  | True |  |
| Support Group | group | Reference | sys\_user\_group | Name | True | Ignore |
| Location | location | Reference | cmn\_location | Name | False | Ignore |
| Serial Number | serial\_number | String |  |  | True |  |
| Asset Tag | asset\_tag | String |  |  | True |  |
| Company | company | Reference | sys\_company | Name | True | Ignore |
| Class | class | String |  |  |  |  |
| Type | type | String |  |  |  |  |

# 4.0 Service Level Agreements

Table: contract\_sla

## 4.1 Response Conditions:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Priority | Duration | Schedule | Start Condition | Stop Condition |
| 1 | 15 minutes | 24x7 | Active is true  --AND--  Priority is 1 | Assigned to is not empty |
| 2 | 1 hours | 24x7 | Active is true  –AND--  Priority is 2 | Assigned to is not empty |
| 3 | 2 hours | Monday – Friday  8am - 5pm | Active is true  --AND--  Priority is 3 | Assigned to is not empty |
| 4 | 4 hours | Monday – Friday  8am - 5pm | Active is true  –AND--  Priority is 4 | Assigned to is not empty |
| 5 | 8 hours | Monday – Friday  8am - 5pm | Active is true  –AND--  Priority is 5 | Assigned to is not empty |

## 4.2 Resolution Conditions:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Priority | Duration | Schedule | Start  Condition | Stop  Condition | Pause Condition |
| 1 | 2 hours | 24x7 | Active is true  –AND--  Priority is 1 | State is closed  –OR--  cancelled | State” field is on hold  --AND--  on-hold reason is one of the following:  Awaiting Caller, Awaiting vendor, Awaiting Change or Awaiting Problem |
| 2 | 8 hours | 24x7 | Active is true  –AND--  Priority is 2 | State is closed  –OR--  cancelled | State is on hold  –AND--  on-hold reason is one of the following:  Awaiting Caller, Awaiting vendor, Awaiting Change or Awaiting Problem |
| 3 | 1 day | Monday – Friday  8am - 5pm | Active is true  –AND--  Priority is 3 | State is closed  –OR--  cancelled | State is on hold  --AND--  on-hold reason is one of the following:  Awaiting Caller, Awaiting vendor, Awaiting Change or Awaiting Problem |
| 4 | 2 days | Monday – Friday  8am - 5pm | Active is true  –AND--  Priority is 4 | State is closed  –OR--  cancelled | State is on hold  --AND--  on-hold reason is one of the following:  Awaiting Caller, Awaiting vendor, Awaiting Change or Awaiting Problem |
| 5 | 5 days | Monday – Friday  8am - 5pm | Active is true  –AND--  Priority is 5 | State is closed  –OR--  cancelled | State is on hold  --AND--  on-hold reason is one of the following:  Awaiting Caller, Awaiting vendor, Awaiting Change or Awaiting Problem |

## 4.3 Holiday Schedule

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Holiday** | **Occurs On** | **Days** | **Start Time** | **End Time** |
| New Year's Day | Every year on Jan 1 | 1 | 2021-01-01 12.00.00 AM | 2021-01-01 11.59.59 PM |
| Martin Luther  King, Jr. Day | Every year on 3rd Mon of Jan | 1 | 2021-01-16 12.00.00 AM | 2021-01-16 11.59.59 PM |
| Valentine's Day | Every year on 14th of Feb | 1 | 2021-02-14 12.00.00 AM | 2021-02-14 11.59.59 PM |
| Presidents’ Day | Every year on 20th of Feb | 1 | 2021-02-20 12.00.00 AM | 2021-02-20 11.59.59 PM |
| National PI Day | Every year on March 14 | 1 | 2021-03-14 12.00.00 AM | 2021-03-14 11.59.59 PM |
| National Goof Off Day | Every year on March 22 | 1 | 2021-03-22 12.00.00 AM | 2021-03-22 11.59.59 PM |
| Fun at Work Day | Every year on Apr 1 | 1 | 2021-04-01 12.00.00 AM | 2021-04-01 11.59.59 PM |
| Equal Pay Day | Every year on Apr 12 | 1 | 2021-04-12 12.00.00 AM | 2021-04-12 11.59.59 PM |
| Cinco de Mayo | Every year on May 5th | 1 | 2021-05-05 12.00.00 AM | 2021-05-05 11.59.59 PM |
| Memorial Day | Every year on last Mon of May | 1 | 2021-05-28 12.00.00 AM | 2021-05-28 11.59.59 PM |
| Juneteenth | Every year on June 19 | 1 | 2021-06-19 12.00.00 AM | 2021-06-19 11.59.59 PM |
| Donut Day | Every year on June 4 | 1 | 2021-06-04 12.00.00 AM | 2021-06-04 11.59.59 PM |
| Independence Day | Every year on July 4 | 1 | 2021-07-04 12.00.00 AM | 2021-07-04 11.59.59 PM |
| Embrace your Geekness day | Every year on July 13 | 1 | 2021-07-13 12.00.00 AM | 2021-07-13 11.59.59 PM |
| Pretend to be a time traveler day | Every year on Aug 8 | 1 | 2021-08-08 12.00.00 AM | 2021-08-08 11.59.59 PM |
| Festivus | Every year on Aug 24 | 1 | 2021-08-24 12.00.00 AM | 2021-08-24 11.59.59 PM |
| Labor Day | Every year on 1st Mon of Sep | 1 | 2021-09-01 12.00.00 AM | 2021-09-01 11.59.59 PM |
| Positive Thinking Day | September 13 | 1 | 2021-09-13 12.00.00 AM | 2021-09-13 11.59.59 PM |
| Columbus Day | Every year on 2nd Mon of Oct | 1 | 2021-10-09 12.00.00 AM | 2021-10-09 11.59.59 PM |
| Halloween | Every year on 31st of October | 1 | 2021-10-31 12.00.00 AM | 2021-10-31 11.59.59 PM |
| Thanksgiving Day | Every year on 4th Thu of Nov | 1 | 2021-11-22 12.00.00 AM | 2021-11-22 11.59.59 PM |
| Veterans Day | Every year on Nov 11 | 1 | 2021-11-11 12.00.00 AM | 2021-11-11 11.59.59 PM |
| Christmas Day | Every year on Dec 25 | 1 | 2021-12-25 12.00.00 AM | 2021-12-25 11.59.59 PM |

# 5.0 Summary

By utilizing this document, the person(s) responsible for integrating new customers into ServiceNow will be able to quickly and succinctly complete the task. Domain Separation allows a SNOW team member to bring in customers on a multi-tenant instance and give them their own domains without any data security issues. With Import Sets, a SNOW team member can use data given to them by the customers and place it within an existing table in ServiceNow. Lastly, SLAs allow both the customer and the SNOW service provider to keep abreast of the amount of time remaining on a ticket.