Function Specification (FncS)

()

Document Type	Function Specification (FncS)	
Document ID	547923	
Document Location	VSEM Rich Client, VSEM Active Workspace	
Document Owner	Jayaraman, Vijay V. (vjayara5)	
Document Version	B	
Document Status	Released	
Date Issued	03-Mar-2022 15:20	
Date Revised	10-Mar-2022 16:54	
Document Classification	GIS1 Item Number:	
	GIS2 Classification:	

Document Approval			
Person	Role	Email Confirmation	Date

CONTENTS

List of Figures

List of Tables

1 INTRODUCTION

1.1 Purpose

The Function (Group) Specification (FS) specifies an individual function / a group of functions.

To get more information about the concept of feature, function and component level abstraction refer to the Ford RE Wiki.

1.2 Scope

The following set of functions from the Global Feature & Function List is described in this specification.

Table 1: Functions described in this specification

Function ID	Function Name	Owner	Reference

1.3 Audience

The FS is authored by the owners of the individual functions. All Stakeholders, i.e., all people who have a valid interest in the functions and their behavior should read and, if possible, review the FS. It needs to be guaranteed, that all stakeholders have access to the currently valid version of the FS.

The following table lists all stakeholders, who should be involved in the creation and maintenance of this FD. Refer to the Roles & Responsibilities page in the in the Ford RE Wiki for a list of common Ford roles and responsibilities.

1.3.1 Stakeholder List

For the latest list of the feature stakeholder and their roles & responsibilities refer to <TBD VSEM Link>.

1.4 Document Organization

1.4.1 Document Context

Refer to the Specification Structure page in the Ford RE Wiki to understand how the FS relates to other Ford Requirements Documents and Specifications.

1.4.2 Document Structure

The structure of this document is explained below:

Section 1 – Introduction how to use this document including responsibilities and requisite documents. Explains the terminology. Gives a clarification of the definitions, concepts and abbreviations used in the document.
Section 2 – Function Group Description. States briefly the background and the purpose of the function group.
Section 3 – Functional Architecture: Specifies the overall functional architecture of the function group
Section 4 – Funciton Requirements: Specifies each function of the function group in detail
Section 5 – List of Open Issues
Section 6 – Traceability Matrix
Section 7 Revision history including a list of new or modified requirements. The requirements in this document are tagged, and this section contains different types of tables listing all, new, or changed requirements by their title and page no.

1.5 References

1.5.1 Ford Documents

List here all Ford internal documents, which are directly related to the feature.

Table 2: Ford Internal Documents

Reference	Title	Doc. ID	Revision
[aaa]			

1.5.2 External Documents and Publications

The list of external documents could include books, reports and online sources.

Table 3: List of External Documents

Reference	Document / Publication
[bbb]	Refer to IEEE Citation Reference for how to format.

1.6 Terminology

#Hint: Terms, concepts and abbreviations used in the document can be defined and illustrated here. Note that changes to terms and/or concepts described in this section tend to cause major updates to this document.

1.6.1 Definitions

#Hint: The tables below have feature specific definitions and abbreviations. For additional, non-feature specific terms please refer to the RE Glossary

Table 4: Definitions used in this document

Definition	Description

Abbr.	Stands for	Description
FS	Function Requirements Specification / Function Group Specification	The document describing, collecting and developing the requirements of a function or a group of functions.

1.6.2 Abbreviations

Table 5: Abbreviations used in this document

1.7 Notation

1.7.1 Requirements Templates

Each requirement, use case or scenario in this specification shall follow the corresponding template given in the document template *Specification_Macros.dotm* on Wiki page “Specification Templates”. This document template also provides macros to insert the requirement templates. Refer to “How to use the Specification Templates” on how to enable the macros and the requirements templates in this specification.

The requirements macro and requirements templates also enable the import of the specification to VSEM (refer to "How to import specifications into VSEM as separate requirements").

1.7.1.1 Identification of requirements

The unique requirement ID given in the headline of any requirement follows the requirement throughout the development process. The requirement ID format follows a well-defined syntax.

All identifiers in a Function Spec shall be composed of 5 parts:

- A leading letter FNC (= Function).
- Followed by the function name
- Followed by a letter indicating the category of Requirement (=R))
- Ending with the actual requirement number

Example:
FNC_LockArbitrator_R_00004 This is the fourth requirement on function level for the function Lock Arbitrator.

1.7.1.2 Requirements Attributes

The macros provided by “Specification Templates” add attributes to each requirement. This helps to classify requirements. The list of available attributes is given in the RE Wiki.

2 FUNCTION GROUP DESCRIPTION

2.1 Overview

Download Manager is a group of functions in IVSU Feature. The main functionalities of Download manager are as follows.

- Download binary files from URLs and check associated metadata provided by OTA Manager.
- Report progress and Error.
- Pause and resume of binary download.
- HTTPS Post/Head request for OTA manager provided URL and Payload

2.2 Input Requirements

#Hint: List any input requirements here (legal, Trustmark), which need to be taken into account, beyond what is specified on feature level.

2.3 Assumptions & Constraints

3 FUNCTIONAL ARCHITECTURE

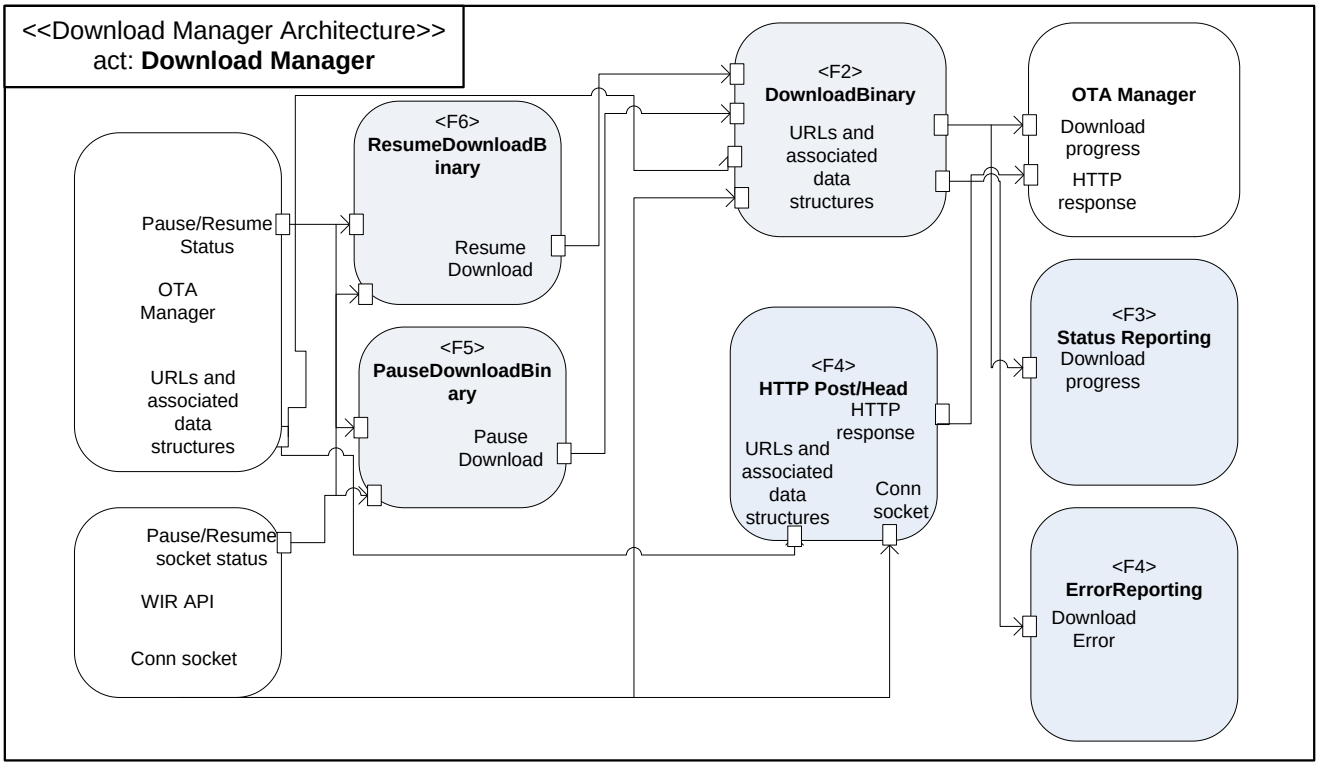

Figure 1: Download Manager – Functional Architecture

3.1 Function List

3.1.1 List of Logical Functions

Table 6: List of Logical Functions

Function ID	Function Name	Function Description
FNC_DownloadManager_R_00001	DownloadBinary	Download binary files from URL provided by OTA Manager with Connectivity Fail over
FNC_DownloadManager_R_00002	ProgressReporting	Status updates during progress of binary files download send to OTA Manager. Based on Policy table/Rules, OTA Manager decide to send to Cloud/HMI at appropriate time.
FNC_DownloadManager_R_00003	ErrorReporting	Error Status updates during progress of binary files download send to OTA Manager. Based on Policy table/Rules,

		OTA Manager decide to send to Cloud/HMI at appropriate time.
FNC_DownloadManager_R_00004	PauseBinaryDownload	Pause binary files download.
FNC_DownloadManager_R_00005	ResumeBinaryDownload	Resume binary files download.
FNC_DownloadManager_R_00006	HTTPSRequest	OTA manager calls this function to post VIL/Status Update, which passes web service URL and associated parameters (Method type, etc) and payload. OTA Manager manager creates payload, packetize(SyncP) and serialize(JSON/Base64).
FNC_DownloadManager_R_00007	*ConnectivityFailOver*	OTA Manager shall provide preferred mediums to be used for DownloadBinary and HTTPSRequest. Preferred medium shall passed to WIR API get socket for session. If None of provided mediums are available and Applink is connected, Applink shall be used.
FNC_DownloadManager_R_00008	CancelBinaryDownload	Cancel binary files download.
FNC_DownloadManager_R_00009	MultipleDownloads	Download Manager shall support multiple (More than one) binary downloads parallel. OTA Manager shall execute based on policy associated with Multiple downloads parallel.
FNC_DownloadManager_R_00010	DataUsage	OTA Manager ingest policy details and User Input on Data Usage limitation. OTA Manger shall Start, reset and get Data usage counter value by providing corresponding input signals. OTA Manager shall Pause and Resume Binary download based on Policy adherence and User Input on Data usage limitation. DM shall invoke Core OS service API to retrieve “Data Usage Since…” data and use it for calculation. If Core OS Service API needs additional parameters to be passed, WIR shall provide along with UID for connections Metered / Non-metered Wi-fi connection type shall be provided through WIR API connection parameters (Say for example, SSID, metered type,etc).

4 LOGICAL FUNCTIONS

4.1 DownloadBinary

4.1.1 Function Description

4.1.2 Function Scope

Refer Figure 3 in section 3

4.1.3 Function Interfaces

4.1.3.1 Logical Inputs

Table 7: List of Logical Inputs

Signal ID	Signal Name	Description
LS_OTAM_TO_DM_00001	URL	Single or multiple URLs.
LS_OTAM_TO_DM_00002	URL_SHA256	Sha256 of each URL contents. Check content integrity Before and After Download. Before start of download HTTP head request check. After Download, calculate Sha256 checksum of content.
LS_OTAM_TO_DM_00003	MEDIUM	Mediums to be used is Connectivity fail over preference provided by OTA Manager from Policy table or Manifest rules.
LS_OTAM_TO_DM_00004	CAMPAIGN_ID	Software Update Campaign ID . Unique identifier identifies a particular software update campaign.

4.1.3.2 Logical Outputs

Table 8: List of Logical Outputs

Signal ID	Signal Name	Description
LS_DM_TO_OTAM_00001	NUM_BYTES_DW	Number of bytes downloaded so far for every URL in request.
LS_DM_TO_OTAM_00002	TOT_SERVER_FILE_SIZE	Total Server file size. Server File size for every URL. Calculated using HTTP head requests.

4.1.3.3 Configuration Parameters

#Hint: Put parameters here, which will be configured using Method 2 or 3

Table 9: Configuration Parameters (Download Binary)

Parameter ID	Parameter Name	Description
CP_DM_00001	MODULE_NAME	If same DM code is reused in different modules (Sync/APIM, ECG, TCU, etc). This Configurable parameter shall be used for Compile time Switch like #If (MODULE_NAME == SYNC). Module name defined in DM header file.

4.1.3.4 Tunable Parameters

Table 10: Tunable Parameters (Download Binary)

Parameter ID	Parameter Name	Description
TP_DM_00001	MAX_NUM_URL_ITEMS	Maximum number of URLs shall downloaded parallel. If OTA Manager request greater than this many parallel URL downloads, DM shall respond with error.

4.1.4 Function Modeling

NA

4.1.5 Function Requirements

4.1.5.1 Functional Requirements

F-REQ-304876/A-###FNC_DownloadManager_R_00001### DownloadBinary

Download binary content through HTTPS web service

- Check for download complete status for each item.
- Check whether downloaded file exists for each item.
- If downloaded file exists, check for server file size and downloaded file size for that item.
- Calculate HTTPS header range based on remaining bytes to be downloaded and chunk size per task rate.
- Create HTTPS session with request time out.
- Send HTTPS GET request for that item's URL.
- If HTTPS response is `HTTP_OK` or `HTTP_PARTIAL_CONTENT`, copy HTTPS response body to item file.
- Update downloaded file size for that item.
- Calculate and update Overall download progress.

F-REQ-304877/A-###FNC_DownloadManager_R_00001### HTTPS only

Download Manager shall only support HTTPS.

4.2 ProgressReporting

4.2.1 Function Description

4.2.2 Function Scope

Refer Figure 3 in section 3

4.2.3 Function Interfaces

4.2.3.1 Logical Inputs

Table 11: Logical Inputs

Signal ID	Signal Name	Description
LS_OTAM_TO_DM_00004	CAMPAIGN_ID	Unique identifier identifies a particular software update campaign.
LS_OTAM_TO_DM_00001	URL	Single or multiple URLs.

4.2.3.2 Logical Outputs

Table 12: Logical Outputs

Signal ID	Signal Name	Description
LS_DM_TO_OTAM_00001	NUM_BYTES_DW	Number of bytes downloaded so far for every URL in request.
LS_DM_TO_OTAM_00002	TOT_SERVER_FILE_SIZE	Server File size for every URL. Calculated using HTTP head requests.

4.2.3.3 Configuration Parameters

#Hint: Put requirements for parameters here, which lead to configuration parameters configured using Method 2 or 3

Table 13: Configuration Parameters (Progress Reporting)

Parameter ID	Parameter Name	Description
NA		

4.2.3.4 Tunable Parameters

Table 14: Tunable Parameters (Progress Reporting)

Parameter ID	Parameter Name	Description
NA		

4.2.4 Function Modeling

NA

4.2.5 Function Requirements

4.2.5.1 Functional Requirements

F-REQ-304878/A-###FNC_DownloadManager_R_00002### ProgressReporting

OTA Manager shall access progress at any rate defined in policy table/Manifest rules/OTA Manager state machine. Based on Policy table/Rules, OTA Manager decide to send to Cloud/HMI at appropriate time.

4.3 ErrorReporting

4.3.1 Function Description

4.3.2 Function Scope

Refer Figure 3 in section 3

4.3.3 Function Interfaces

4.3.3.1 Logical Inputs

Table 15: Logical Inputs (Error Reporting)

Signal ID	Signal Name	Description
NA		

4.3.3.2 Logical Outputs

Table 16: Logical Outputs (Error Reporting)

Signal ID	Signal Name	Description
LS_DM_TO_OTAM_00003	ERROR_TYPE	Error Type . Type of error
LS_DM_TO_OTAM_00004	ERROR_CODE	Error Code . Error code associated (For example HTTP error, 3xx, 4xx & 5xx)
LS_DM_TO_OTAM_00005	ERROR_DESCRIP	Error description . Error message description

4.3.3.3 Configuration Parameters

#Hint: Put parameters here, which will be configured using Method 2 or 3

Table 17: Configuration Parameters (Error Reporting)

Parameter ID	Parameter Name	Description
NA		

4.3.3.4 Tunable Parameters

Table 18: Tunable Parameters (Error Reporting)

Parameter ID	Parameter Name	Description
NA		

4.3.4 Function Modeling

NA

4.3.5 Function Requirements

4.3.5.1 Functional Requirements

F-REQ-304879/B-###FNC_DownloadManager_R_00003### ErrorReporting

DM shall report error to OTA Manager. OTA Manager decide to send to Cloud/HMI at appropriate time.
The following are error conditions

- Item's URL format is incorrect / link is broken.
    - Verify the validity of URL by HTTP Head request.
- HTTP error
- File I/O error
- Buffer/Stream error
- WIR API error

Table 19: List of Error Conditions

Error	Detected	Recoverable	Root Cause	Severity	Report to Cloud
Item's URL format is incorrect / link is broken.	YES	YES	CDN/URL wrong / Manifest contains Incorrect/broken URL	Normal	YES
HTTP error	YES	YES	Connectivity	Normal	YES*
File I/O error	YES	YES	Access change/Memory error/OS Errors	Normal	NO
Buffer/Stream Error	YES	YES	Access change/Memory error/OS Errors	Normal	NO
WIR API Error	YES	YES	Connectivity	Normal	NO
Cache Full	YES	YES	Access change/Memory error/OS Errors	Normal	NO

*only after N number of times consecutive occurrence of exact same certain type errors (For example 3xx, 4xx & 5xx). OTA manager shall decide on this.

4.4 PauseBinaryDownload

4.4.1 Function Description

OTA Manager shall pause binary download. It shall be applicable for one or more binaries download associated Software update Campaign.

4.4.2 Function Scope

Refer Figure 3 in section 3

4.4.3 Function Interfaces

4.4.3.1 Logical Inputs

Table 20: Logical Inputs (Pause Binary Download)

Signal ID	Signal Name	Description
LS_OTAM_TO_DM_00005	PAUSE_BINARY	PauseBinaryDownload
LS_OTAM_TO_DM_00004	CAMPAIGN_ID	Unique identifier identifies a particular software update campaign.
LS_OTAM_TO_DM_00001	URL	Single or multiple URLs.

4.4.3.2 Logical Outputs

Table 21: Logical Outputs (Pause Binary Download)

Signal ID	Signal Name	Description
LS_DM_TO_OTAM_00006	PAUSE_STATUS	Pause Status

4.4.3.3 Configuration Parameters

#Hint: Put parameters here, which will be configured using Method 2 or 3

Table 22: Configuration Parameters (Pause Binary Download)

Parameter ID	Parameter Name	Description
NA		

4.4.3.4 Tunable Parameters

Table 23: Tunable Parameters (Pause Binary Download)

Parameter ID	Parameter Name	Description

NA		

4.4.4 Function Modeling

NA

4.4.5 Function Requirements

4.4.5.1 Functional Requirements

F-REQ-304880/A-###FNC_DownloadManager_R_00004### PauseBinaryDownload

OTA Manager shall pause binary download. It shall be applicable for one or more binaries download associated Software update Campaign.

4.5 ResumeBinaryDownload

4.5.1 Function Description

OTA Manager shall resume binary download. It shall be applicable for one or more binaries download associated Software update Campaign.During resume of binary download, HTTPS offset shall be calculated based on file size, not based on persisted data.

4.5.2 Function Scope

Refer Figure 3 in section 3

4.5.3 Function Interfaces

4.5.3.1 Logical Inputs

Table 24: Logical Inputs (Resume Binary Download)

Signal ID	Signal Name	Description
LS_OTAM_TO_DM_00006	RESUME_BINARY	ResumeBinaryDownload

LS_OTAM_TO_DM_00004	CAMPAIGN_ID	Unique identifier identifies a particular software update campaign.
LS_OTAM_TO_DM_00001	URL	Single or multiple URLs.

4.5.3.2 Logical Outputs

Table 25: Logical Outputs (Resume Binary Download)

Signal ID	Signal Name	Description
LS_DM_TO_OTAM_00007	RESUME_STATUS	Resume Status

4.5.3.3 Configuration Parameters

#Hint: Put parameters here, which will be configured using Method 2 or 3

Table 26: Configuration Parameters (Resume Binary Download)

Parameter ID	Parameter Name	Description
NA		

4.5.3.4 Tunable Parameters

Table 27: Tunable Parameters (Resume Binary Download)

Parameter ID	Parameter Name	Description
NA		

4.5.4 Function Modeling

NA

4.5.5 Function Requirements

4.5.5.1 Functional Requirements

F-REQ-304881/A-###FNC_DownloadManager_R_00005### ResumeBinaryDownload

OTA Manager shall resume binary download. It shall be applicable for one or more binaries download associated Software update Campaign.

4.6 CancelBinaryDownload

4.6.1 Function Description

OTA Manager shall send request to cancel binary download. It shall be applicable for one or more binaries download associated Software update Campaign. Download manager shall delete partially downloaded files.

4.6.2 Function Scope

Refer Figure 3 in section 3

4.6.3 Function Interfaces

4.6.3.1 Logical Inputs

Table 28: Logical Inputs (Cancel Binary Download)

Signal ID	Signal Name	Description
LS_OTAM_TO_DM_00008	CANCEL_BINARY	CancelBinaryDownload
LS_OTAM_TO_DM_00004	CAMPAIGN_ID	Unique identifier identifies a particular software update campaign.
LS_OTAM_TO_DM_00001	URL	Single or multiple URLs.

4.6.3.2 Logical Outputs

Table 29: Logical Output (Cancel Binary Download)

Signal ID	Signal Name	Description
LS_DM_TO_OTAM_00008	CANCEL_STATUS	Cancel Status

4.6.3.3 Configuration Parameters

#Hint: Put parameters here, which will be configured using Method 2 or 3

Table 30: Configuration Parameters (Cancel Binary Download)

Parameter ID	Parameter Name	Description
NA		

4.6.3.4 Tunable Parameters

Table 31: Tunable Parameters (Cancel Binary Download)

Parameter ID	Parameter Name	Description
NA		

4.7 No Title

4.7.1 Function Modeling

NA

4.7.2 Function Requirements

4.7.2.1 Functional Requirements

F-REQ-304884/A-###FNC_DownloadManager_R_00006### CancelBinaryDownload

OTA Manager shall send request to cancel binary download. It shall be applicable for one or more binaries download associated Software update Campaign. Download manager shall delete partially downloaded files.

.

4.8 HTTPSRequest

4.8.1 Function Description

OTA Manager shall use this function for HTTPS HEAD/POST requests. OTA Manager shall pass associated payload for HTTPS requests.

4.8.2 Function Scope

Refer Figure 3 in section 3

4.8.3 Function Interfaces

4.8.3.1 Logical Inputs

Table 32: Logical Inputs (HTTPS Request)

Signal ID	Signal Name	Description
LS_OTAM_TO_DM_00001	URL	Single or multiple URLs.
LS_OTAM_TO_DM_00002	URL_SHA256	Sha256 of each URL contents. Check content integrity Before and After Download. Before start of download HTTP head request check. After Download, calculate Sha256 checksum of content.
LS_OTAM_TO_DM_00003	MEDIUM	Mediums to be used is Connectivity fail over preference provided by OTA Manager from Policy table or Manifest rules.
LS_OTAM_TO_DM_00004	CAMPAIGN_ID	Software Update Campaign ID . Unique identifier identifies a particular software update campaign.
LS_OTAM_TO_DM_00009	HTTPS_METHOD	POST/HEAD. Idempotent HTTPS requests are handled in this function. If HTTPS GET, Binary Download function shall be used.
LS_OTAM_TO_DM_00010	HTTPS_REQ_PAYLOAD	OTA Manager shall provide payload for Idempotent HTTPS POST requests.

4.8.3.2 Logical Outputs

Table 33: Logical Outputs (HTTPS request)

Signal ID	Signal Name	Description
LS_DM_TO_OTAM_00009	HTTPS_STATUS	HTTP Status. HTTP or HTTP error NUM_BYTES_DW
LS_DM_TO_OTAM_00010	NUM_BYTES_RET_PAYLOAD	Number of bytes return payload. Number of bytes returned as payload for HTTP OK.
LS_DM_TO_OTAM_00011	HTTPS_ERROR	HTTP Error.HTTP error code returned
LS_DM_TO_OTAM_00012	PAYLOAD	Returned Payload from cloud for HTTPS POST shall be passed to OTAM

4.8.3.3 Configuration Parameters

#Hint: Put parameters here, which will be configured using Method 2 or 3

Table 34: Configuration Parameters (HTTPS Request)

Parameter ID	Parameter Name	Description
LS_DM_HTTPS_CLIENT_TIMEOUT	HTTPS Client timeout	Default to 120 seconds. Any changes refer to Policy table.

4.8.3.4 Tunable Parameters

Table 35: Tunable Parameters (HTTPS Request)

Parameter ID	Parameter Name	Description
NA		

4.8.4 Function Modeling

NA

4.8.5 Function Requirements

4.8.5.1 Functional Requirements

F-REQ-304885/A-###FNC_DownloadManager_R_00007### HTTPSRequest

OTA Manager shall use this function for HTTPS HEAD/POST requests. OTA Manager shall pass associated payload for HTTPS requests. DM shall only support HTTPS request.

4.9 ConnectivityFailOver

4.9.1 Function Description

4.9.2 Function Scope

Refer Figure 3 in section 3

4.9.3 Function Interfaces

4.9.3.1 Logical Inputs

Table 36: Logical Inputs (Connectivity Failover)

Signal ID	Signal Name	Description
LS_OTAM_TO_DM_00001	URL	Single or multiple URLs.
LS_OTAM_TO_DM_00002	URL_SHA256	Sha256 of each URL contents. Check content integrity Before and After Download. Before start of download HTTP head request check. After Download, calculate Sha256 checksum of content.
LS_OTAM_TO_DM_00003	MEDIUM	Mediums to be used is Connectivity fail over preference provided by OTA Manager from Policy table or Manifest rules.
LS_OTAM_TO_DM_00004	CAMPAIGN_ID	Software Update Campaign ID . Unique identifier identifies a particular software update campaign.

4.9.3.2 Logical Outputs

Table 37: Logical Output (Connectivity Failover)

Signal ID	Signal Name	Description
NA		

4.9.3.3 Configuration Parameters

#Hint: Put parameters here, which will be configured using Method 2 or 3

Table 38: Configuration Parameters (Connectivity Failover)

Parameter ID	Parameter Name	Description
NA		

4.9.3.4 Tunable Parameters

Table 39: Tunable Parameters (Connectivity Failover)

Parameter ID	Parameter Name	Description
NA		

4.9.4 Function Modeling

NA

4.9.5 Function Requirements

4.9.5.1 Functional Requirements

F-REQ-304886/A-###FNC_DownloadManager_R_00008### ConnectivityFailOver

OTA Manager shall provide preferred mediums to be used for DownloadBinary and HTTPSRequest. Preferred medium shall passed to WIR API get socket for session. If None of provided mediums are available and Applink is connected, Applink shall be used.

4.10 MultipleDownloads

4.10.1 Function Description

4.10.2 Function Scope

Refer Figure 3 in section 3

4.10.3 Function Interfaces

4.10.3.1 Logical Inputs

Table 40: Logical Inputs (Multiple Downloads)

Signal ID	Signal Name	Description
NA		

4.10.3.2 Logical Outputs

Table 41: Logical Outputs (Multiple Downloads)

Signal ID	Signal Name	Description
NA		

4.10.3.3 Configuration Parameters

#Hint: Put parameters here, which will be configured using Method 2 or 3

Table 42: Configuration Parameters (Multiple Downloads)

Parameter ID	Parameter Name	Description
NA		

4.10.3.4 Tunable Parameters

Table 43: Tunable Parameters (Multiple Downloads)

Parameter ID	Parameter Name	Description
NA		

4.10.4 Function Modeling

NA

4.10.5 Function Requirements

4.10.5.1 Functional Requirements

F-REQ-304887/A-###FNC_DownloadManager_R_00009### MultipleDownloads

Download shall support multiple (More than one) binary downloads parallel. OTA Manager shall execute based on policy associated with Multiple downloads parallel.

4.11 DataUsage

4.11.1 Function Description

OTA Manager ingest policy details and User Input on Data Usage limitation.
OTA Manger shall Start, reset and get Data usage counter value by providing corresponding input signals.
OTA Manager shall Pause and Resume Binary download based on Policy adherence and User Input on Data usage limitation.
DM shall invoke Core OS service API to retrieve “Data Usage Since…” data and use it for calculation. If Core OS Service API needs additional parameters to be passed, WIR shall provide along with UID for connections
Metered / Non-metered Wi-fi connection type shall be provided through WIR API connection parameters (Say for example, SSID, metered type,etc)

4.11.2 Function Scope

Refer Figure 3 in section 3

4.11.3 Function Interfaces

4.11.3.1 Logical Inputs

Table 44: Logical Inputs (Data Usage)

Signal ID	Signal Name	Description
LS_OTAM_TO_DM_00013	START_DATA_USAGE_COUNTER	To Start Data Usage counters in DM
LS_OTAM_TO_DM_00014	RESET_DATA_USAGE_COUNTER	To reset Data usage counters in DM
LS_OTAM_TO_DM_00015	GET_DATA_USAGE	To retrieve Data usage counters from DM

4.11.3.2 Logical Outputs

Table 45: Logical Outputs (Data Usage)

Signal ID	Signal Name	Description
LS_DM_TO_OTAM_00015	DATA_USAGE	DM shall provide data usage since Last Reset for all uniquely identified connection mediums

4.11.3.3 Configuration Parameters

#Hint: Put parameters here, which will be configured using Method 2 or 3

Table 46: Configuration Parameters (Data Usage)

Parameter ID	Parameter Name	Description
NA		

4.11.3.4 Tunable Parameters

Table 47: Tunable Parameters (Data Usage)

Parameter ID	Parameter Name	Description
NA		

4.11.4 Function Modeling

NA

4.11.5 Function Requirements

4.11.5.1 Functional Requirements

F-REQ-304888/A-###FNC_DownloadManager_R_00010### DataUsage – OTA Manager invoke DM

OTA Manager ingest policy details and User Input on Data Usage limitation.
OTA Manger shall Start, reset and get Data usage counter value by providing corresponding input signals.
OTA Manager shall Pause and Resume Binary download based on Policy adherence and User Input on Data usage limitation.

F-REQ-304889/A-###FNC_DownloadManager_R_01010### DataUsage – DM interface WIR and OS service APIs

DM shall invoke Core OS service API to retrieve “Data Usage Since…” data and use it for calculation. If Core OS Service API needs additional parameters to be passed, WIR shall provide along with UID for connections
Metered / Non-metered Wi-fi connection type shall be provided through WIR API connection parameters (Say for example, SSID, metered type,etc)
.

5 OPEN ISSUES

Table 48: List of Open Issues

ID	Issue Description	e-Tracker / Reference	Responsible	Status	Solution
1					
2					
3					
4					
5					
6					
7					
8					
9					

6 TRACEABILITY MATRIX

Note: The requirements traceability matrix will be generated in the future by VSEM.
For the time being this just lists all requirements specified in this document.

7 REVISION HISTORY

#Hint: A new version number is assigned to a document with a given revision each time it is checked in to Team Center (TCSE). After release of a revision, the document can not be edited and no new versions can be created on that revision. When updating the document after that, a new revision has to be created and new versions on that revision will be created upon checking in.

Table 49: Revision History

Revision	Date *MM/DD/YYYY*	Description	Responsible	Approved by
A		***Initial version***		
B	***3/10/2022***	**As a part of the specification clean-up exercise, removed the following 3 invalid/Not supported requirements.** ***REQ-304883 – Invalid*** ***REQ-304891 – Not supported*** ***REQ-304892 – Not supported*** ***Formatted the list of tables and figures.***	***Snehaja Gogineni***	

8 APPENDIX

8.1 Data Dictionary

#Hint: This gives a list of logical signals. Duplicate sections by copy & paste.