Oracle Application Testing Suite for Oracle Utilities

Reference Guide for Mobile Workforce Management and Oracle Real-Time Scheduler

Release 4.2.0

E58834-01

January 2015

ORACLE®

Oracle Application Testing Suite for Oracle Utilities Reference Guide for Mobile Workforce Management and Oracle Real-Time Scheduler, Release 4.2.0

E58834-01

Contents

Preface

This guide describes the Oracle Utilities Mobile Workforce Management (MWM) v2.2.0.1 and Oracle Real-Time Scheduler (ORS) v2.2.0.1 components and the function libraries used to create those components in Oracle Application Testing Suite for Oracle Utilities (OATSOU) v4.2.0. These components are used to build test flows in Oracle Flow Builder (OFB).

This preface includes the following sections:

- Audience
- Related Documents
- Notational Conventions

Audience

This guide is intended for QA/Test Engineers and Automation Developers to understand the various components and libraries available for them to automate the business test flows for Oracle Utilities Mobile Workforce Management using Oracle Application Testing Suite for Oracle Utilities (OATSOU) for Oracle Utilities Mobile Workforce Management and Oracle Real-Time Scheduler.

Related Documents

For more information, see the following documents:

- *Oracle Application Testing Suite for Oracle Utilities Administration and User's Guide*
- *Oracle Application Testing Suite for Oracle Utilities Release Notes*

See also:

- Oracle Utilities Mobile Workforce Management Documentation Library
- Oracle Real-Time Scheduler Documentation Library

Notational Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Chapter 1

Component Reference

This chapter lists the Oracle Utilities Mobile Workforce Management starter components available to create flows in Oracle Flow Builder for testing the Oracle Utilities Mobile Workforce Management and Oracle Real-Time Scheduler applications.

The chapter includes the following sections:

- Overview

- Components

Overview

Oracle Application Testing Suite for Oracle Utilities for Mobile Workforce Management and Oracle Real-Time Scheduler is a test starter pack built on top of Oracle Application Testing Suite for Oracle Utilities that generates test automation scripts using Oracle Flow Builder.

Oracle Application Testing Suite for Oracle Utilities for Mobile Workforce Management and Oracle Real-Time Scheduler contains out-of-the-box product-specific components used to build new test flows in Oracle Flow Builder to test the Mobile Workforce Management and Oracle Real-Time Scheduler applications. These out-of-the-box components correspond to specific business entities, such as business objects, service scripts, or business services used for interfacing with the application. Users can use these components as available or can extend them. Users can also create new components to be used to create flows. This starter pack also contains a set of function libraries that can be used for creating custom components.

> **Note:** See Chapter 2: Function Library Reference for detailed information about using these function libraries.

Consider this pack to be a starter kit as it does not include executable test flows. It includes few sample flows that are provided as a reference for creating flows.

> **Note:** See the *Oracle Application Testing Suite for Oracle Utilities Administration and User's Guide* for information about creating components and flows.

The components are categorized under the following functional areas:

- Administration Data

- Connected MCP Processing

- Contractor Portal

- Resource Management

- Scheduling

- Service Management

Component Reference 1-1

Oracle Application Testing Suite for Oracle Utilities Reference Guide for Mobile Workforce Management and Oracle Real-Time Scheduler

Components

The following table lists the starter components available in Oracle Utilities Mobile Workforce Management.

Pre-requisites: The XAI inbound service/Web service using the respective business object should be available in the application.

Additional Notes: Failure while creating, reading, or updating the component is logged to the test execution report specifying the reason for the failure to debug/analyze and raise bugs accordingly.

Component	Functional Area	Description
M1-ActivityType	Administration Data	Used to create, read, or update M1 Activity Type via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-ActivityType business object.
M1-AlertType	Administration Data	Used to create, read, or update an M1 Alert Type via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-AlertType business object.
M1-AppointmentBookingGroup	Administration Data	Used to create, read, or update an M1 Appointment Booking Group via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-AppointmentBookingGroup business object.
M1-BreakTaskType	Administration Data	Used to create, read, or update an M1 Break Task Type via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-BreakTaskType business object.
M1-CapacityType	Administration Data	Used to create, read, or update an M1 Capacity Type via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-CapacityType business object.

Component	Functional Area	Description
M1-ComplexActivityType	Administration Data	Used to create, read, or update an M1 Complex Activity Type via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-ComplexActivityType business object.
M1-CrewHierarchy	Administration Data	Used to create, read, or update an M1-CrewHierarchy via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-CrewHierarchy business object.
M1-ContractorType	Administration Data	Used to create, read, or update an M1 Contractor Type via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-ContractorType business object.
M1-CrewShiftType	Administration Data	Used to create, read, or update an M1 Crew Shift Type via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-CrewShiftType business object.
M1-CrewType	Administration Data	Used to create, read, or update an M1 Crew Type via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-CrewType business object.
M1-Deployment	Administration Data	Used to activate or deactivate a deployment via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-Deployment business object.

Component	Functional Area	Description
M1-DeploymentPart	Administration Data	Used to create, read, or update a deployment part via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-DeploymentPart business object.
M1-DeploymentType	Administration Data	Used to create, read, or update a deployment type via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-DeploymentType business object.
M1-Depot	Administration Data	Used to create, read, or update a depot via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-Depot business object.
M1-DepotProfile	Administration Data	Used to create, read, or update a depot profile via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-DepotProfile business object.
M1-DepotTaskType	Administration Data	Used to create, read, or update a depot task type via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-DepotTaskType business object.
M1-DispatchArea	Administration Data	Used to create, read, or update an M1 Dispatch Area via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-DispatchArea business object.

Component	Functional Area	Description
M1-DispatcherType	Administration Data	Used to create, read, or update an M1 Dispatcher Type via a Web service. the CRUD operations can be performed through these components against the M1-DispatcherType business object.
M1-Equipment	Administration Data	Used to create, read, or update an M1 Equipment via a Web service. the CRUD operations can be performed through these components against the M1-Equipment business object.
M1-GeographicArea	Administration Data	Used to create, read, or update an M1 Geographic Area via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-GeographicArea business object.
M1-GlobalConfig	Administration Data	Used to create, read, or update M1 Global Configurations via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-GlobalConfigurations business object.
M1-KPI	Administration Data	Used to create, read, or update an M1 KPI via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-KPI business object.
M1-Location	Administration Data	Used to create, read, or update an M1 Location via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-Location business object.

Component	Functional Area	Description
M1-MobileDeviceTerminal	Administration Data	Used to create, read, or update an M1 Mobile Device Terminal via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-MobileDeviceTerminal business object.
M1-MobileWorkerType	Administration Data	Used to create, read, or update an M1 Mobile Worker Type via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-MobileWorkerType business object.
M1-NPTTaskType	Administration Data	Used to create, read, or update an M1 Non Prd Task Type via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-NonPrdTaskType business object.
M1-POUTaskType	Administration Data	Used to create, read, or update an M1 POU Task Type via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-POUTaskType business object.
M1-POUType	Administration Data	Used to create, read, or update an M1 POU Type via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-POUType business object.
M1-PriorityProfile	Administration Data	Used to create, read, or update an M1 Priority Profile via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-PriorityProfile business object.

Component	Functional Area	Description
M1-ProcedureType	Administration Data	Used to create, read, or update an M1 Procedure Type via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-ProcedureType business object.
M1-RemarkType	Administration Data	Used to create, read, or update an M1 Remark Type via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-RemarkType business object.
M1-Scheduler	Administration Data	Used to create, read, or update an M1 Scheduler via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-Scheduler business object.
M1-SchedulerArea	Administration Data	Used to create, read, or update an M1 Scheduler Area via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-SchedulerArea business object.
M1-SchedulerConfig	Administration Data	Used to create, read, or update an M1 Scheduler Config via a Web service. the CRUD operations can be performed through these components against the M1-SchedulerConfig business object.
M1-ServiceArea	Administration Data	Used to create, read, or update an M1 Service Area via a Web service. the CRUD operations can be performed through these components against the M1-ServiceArea business object.

Component	Functional Area	Description
M1-ServiceAreaHierarchy	Administration Data	Used to create, read, or update an M1 Service Area Hierarchy via a Web service. the CRUD operations can be performed through these components against the M1-ServiceAreaHierarchy business object.
M1-ServiceClass	Administration Data	Used to create, read, or update an M1 Service Class via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-ServiceClass business object.
M1-ShiftCostProfile	Administration Data	Used to create, read, or update an M1 Shift Cost Profile via a Web service. the CRUD operations can be performed through these components against the M1-ShiftCostProfile business object.
M1-Skill	Administration Data	Used to create, read, or update an M1 Skill via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-Skill business object.
M1-StatusReason	Administration Data	Used to create, read, or update an M1 Status Reason via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-StatusReason business object.
M1-TimeSheetType	Administration Data	Used to create, read, or update an M1-TimeSheetType via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-TimeSheetType business object.

Component	Functional Area	Description
M1-VehicleType	Administration Data	Used to create, read, or update an M1 Vehicle Type via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-VehicleType business object.
M1-WorkProfile	Administration Data	Used to create, read, or update an M1 Work Profile via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-WorkProfile business object.
M1-Capacity	Contractor Portal	Used to create, read, or update an M1 Capacity via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-Capacity business object.
M1-CapacityTemplate	Contractor Portal	Used to create, read, or update an M1 Capacity Template via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-CapacityTemplate business object.
M1-CapacityWeeklyTemplate	Contractor Portal	Used to create, read, or update an M1-CapacityWeeklyTemplate via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-CapacityWeeklyTemplate business object.
M1-Company	Contractor Portal	Used to create, read, or update an M1 Company via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-Company business object.

Component	Functional Area	Description
M1-Contractor	Contractor Portal	Used to create, read, or update an M1 Contractor via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-Contractor business object.
M1-ContractorEligibility	Contractor Portal	Used to create, read, or update an M1 Contractor Eligibility via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-ContractorEligibility business object.
M1-Alert	Service Management	Used to create, read, or update an M1 General Alert via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-GeneralAlert business object.
M1-BreakTask	Resource Management	Used to create, read, or update an M1 Break Task via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-BreakTask business object.
M1-CommonShiftWeeklyTemplate	Resource Management	Used to create, read, or update an M1 Common Shift Weekly Template via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-CommonShiftWeeklyTemplate business object.
M1-CrewShftCloseOpenAllocate	Resource Management	Used to close or open a shift using M1-CrShfClAl via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-CrShfClAl service script.

Component	Functional Area	Description
M1-CrewShift	Resource Management	Used to create, read, or update an M1 Crew Shift via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-CrewShift business object.
M1-CrewShiftTemplate	Resource Management	Used to create, read, or update an M1 Crew Shift Template via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-CrewShiftTemplate business object.
M1-DeleteResourceLeavePOU	Resource Management	Used for deleting M1 Resource Leave POU
M1-DepotRelatedShift	Resource Management	Used to create, read, or update an M1 Depot Related Shift via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-DepotRelatedShift business object.
M1-DepotSinglePersonCrew	Resource Management	Used to create, read, or update an M1 Depot Single Person Crew via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-DepotSinglePersonCrew business object.
M1-Dispatcher	Resource Management	Used to create, read, or update an M1 Dispatcher via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-Dispatcher business object.
M1-DispatcherShift	Resource Management	Used to create, read, or update an M1 Dispatcher Shift via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-DispatcherShift business object.

Component	Functional Area	Description
M1-ForceLogOffCrewShift	Resource Management	Used to force logging off an MDT crew shift
M1-GetShiftCapabilities	Resource Management	Used to retrieve the capabilities of a crew shift
M1-HazardArea	Resource Management	Used to create, read, or update an M1 Hazard Area via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-HazardArea business object.
M1-MailingList	Resource Management	Used to create, read, or update an M1 Mailing List via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-MailingList business object.
M1-MainMail	Resource Management	Used to create M1 MainMail
M1-MobileWorker	Resource Management	Used to create, read, or update an M1 Mobile Worker via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-MobileWorker business object.
M1-MultiPersonCrew	Resource Management	Used to create, read, or update a multi person crew via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-CrewType business object.
M1-NonProductiveTask	Resource Management	Used to create, read, or update an M1 NonPrdTask via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-NonPrdTask business object.
M1-POUTask	Resource Management	Used to create, read, or update an M1 POU Task via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-POUTask business object.

Component	Functional Area	Description
M1-RealPOU	Resource Management	Used to create, read, or update an M1 Real POU via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-RealPOU business object.
M1-ResourceTimesheet	Resource Management	Used to create, read, or update an M1 Res Time Sheet via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-ResTimesheet business object.
M1-ResourceleavePOU	Resource Management	Used to create, read, or update an M1 Resource Leave POU via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-ResourceLeavePOU business object.
M1-ShiftWeeklyTemplate	Resource Management	Used to create, read, or update an M1 Shift Weekly Template via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-ShiftWeeklyTemplate business object.
M1-SimpleProcedure	Resource Management	Used to create, read, or update an M1 Simple Procedure via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-SimpleProcedure business object.
M1-SinglePersonCrew	Resource Management	Used to create, read, or update an M1 Single Person Crew via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-SinglePersonCrew business object.

Component	Functional Area	Description
M1-SubscriptionWeeklyTemplate	Resource Management	Used to create, read, or update an M1 Subscription Weekly Template via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-SubscriptionWeeklyTemp business object.
M1-TemplateMail	Resource Management	Used to create, read, or update an M1 Template Mail via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-TemplateMail business object.
M1-TemplatePOU	Resource Management	Used to create, read, or update an M1 Template POU via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-TemplatePOU business object.
M1-Vehicle	Resource Management	Used to create, read, or update an M1 Vehicle via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-Vehicle business object.
M1-VehicleLeaveArea	Resource Management	Used to create, read, or update an M1 Vehicle Leave Area via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-VehicleLeaveArea business object.
M1-getPOUTaskIdForPOUId	Resource Management	Used to retrieve the POU task ID (the task scheduled to a crew shift) for a given real POU ID.
M1-AllocateActivityToAShift	Scheduling	Used to manually allocate an activity to a crew shift.
M1-AppointmentBookingRequest	Scheduling	Used to trigger an M1 Appointment Booking Request.
M1-CheckIfActivityIsScheduled	Scheduling	Verifies if an activity is scheduled to a crew shift.

Component	Functional Area	Description
M1-DailySpeedTemplate	Scheduling	Used to create, read, or update an M1 Daily Speed Template via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-DailySpeedTemplate business object.
M1-FixToCrew	Scheduling	Used to fix an activity to be scheduled to a particular crew
M1-SchedulerRegistryRead	Scheduling	Used to retrieve the scheduler registry details for an entity
M1-SpeedProfileTemplate	Scheduling	Used to create, read, or update an M1 Speed Profile Template via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-SpeedProfileTemplate business object.
M1-TaskScheduleDetails	Scheduling	Used for reading an M1 Task Schedule entity
M1-UnAssignActivity	Scheduling	Used to unassign an activity from the crew shift
M1-Activity	Service Management	Used to create, read, or update an M1 Activity via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-Activity business object.
M1-Assignment	Service Management	Used to create, read, or update an M1 Assignment via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-Assignment business object.
M1-ComplexActivity	Service Management	Used to create, read, or update an M1 Complex Activity via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-ComplexActivity business object.
M1-DeferActivity	Service Management	Used to defer an activity

Component	Functional Area	Description
M1-DepotRelatedActivity	Service Management	Used to create, read, or update an M1 Depot Related Activity via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-DepotRelatedActivity business object.
M1-DepotTask	Service Management	Used to create, read, or update an M1 Depot Task via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-DepotTask business object.
M1-GetAssignmentIdForTaskId	Service Management	Used to retrieve the assignment ID of a dispatched task
M1-GetDepotTaskId	Service Management	Used to retrieve the depot task ID associated to a given depot related or a pre-requisite activity.
M1-OverideTimeWindow	Service Management	Used to create, read, or update an M1 Overide Time Window via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-OverideTimeWindow business object.
M1-Visit	Service Management	Used to create, read, or update an M1 Visit via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-Visit business object.
M2-InstallMeterActivity	M2 Activity	Used to create, read, or update an M2 Install Meter Activity via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-InstallMeterActivity business object.

Component	Functional Area	Description
M2-InstallMeterAssignment	M2 Activity	Used to create, read, or update an M2 Install Meter Assignment via a Web service. After the creation, the CRUD operations can be performed through these components against the M1-InstallMeterAssignment business object.
M2-MaintUtilityActByHost	M2 Activity	Used to create an M2 Activity from an external system
ConnectedMCP_CompleteActivity	Connected MCP Processing	Performs the actions of a field worker for completing an M1 Activity using Connected MCP. **Note:** To be called only after calling the ConnectedMCP_StartActivity component.
ConnectedMCP_CompleteBreak	Connected MCP Processing	Performs the actions of a field worker for completing an M1 Break Task using Connected MCP. **Note:** To be called only after calling the ConnectedMCP_StartBreak component.
ConnectedMCP_CompleteNPT	Connected MCP Processing	Performs the actions of a field worker for completing an M1 Non Productive Task using Connected MCP. **Note:** To be called only after calling the ConnectedMCP_StartNPT component.
ConnectedMCP_CompletePOU	Connected MCP Processing	Performs the actions of a field worker for completing an M1 POU task using Connected MCP. **Note:** To be called only after calling the ConnectedMCP_StartPOU component.

Component	Functional Area	Description
ConnectedMCP_CompleteShift	Connected MCP Processing	Performs the actions of a field worker for completing a shift using Connected MCP. **Note:** To be called only after a shift is in 'start' status.
ConnectedMCP_DispatchedTask	Connected MCP Processing	Performs the actions of a field worker to check if a task has been dispatched using Connected MCP. Then, click it.
ConnectedMCP_EnrouteActivity	Connected MCP Processing	Performs the actions of a field worker to move the status of a task to 'En-route' using Connected MCP. Then, click it. **Note**: To be called only after calling the ConnectedMCP_DispatchedTask component.
ConnectedMCP_EnrouteNPT	Connected MCP Processing	Performs the actions of a field worker to move the status of an NPT task to 'En-route' using Connected MCP. Then, click it. **Note**: To be called only after calling the ConnectedMCP_DispatchedTask component.
ConnectedMCP_EnroutePOU	Connected MCP Processing	Performs the actions of a field worker to move the status of an NPT task to 'En-route' using Connected MCP. Then, click it. **Note**: To be called only after calling the ConnectedMCP_DispatchedTask component.
ConnectedMCP_StartActivity	Connected MCP Processing	Performs the actions of a field worker to move the status of an activity to 'Started' using Connected MCP. Then, click it. **Note**: To be called only after calling the ConnectedMCP_EnrouteActivity component.

Component	Functional Area	Description
ConnectedMCP_StartBreak	Connected MCP Processing	Performs the actions of a field worker to move the status of a break task to 'Started' using Connected MCP. Then, click it. **Note**: To be called only after calling the ConnectedMCP_DispatchedTask component.
ConnectedMCP_StartCrewShift	Connected MCP Processing	Performs the actions of a field worker to start a crew shift using Connected MCP. Then, click it.
ConnectedMCP_StartNPT	Connected MCP Processing	Performs the actions of a field worker to move the status of an activity to 'Started' using Connected MCP. Then, click it. **Note**: To be called only after calling the ConnectedMCP_EnrouteNPT component.
ConnectedMCP_StartPOU	Connected MCP Processing	Performs the actions of a field worker to move the status of a POU to 'Started' using Connected MCP. Then, click it. **Note**: To be called only after calling the ConnectedMCP_EnroutePOU component.

Chapter 2

Function Library Reference

This chapter lists the Oracle Utilities Mobile Workforce Management function libraries and functions available to create components and flows in Oracle Flow Builder for testing the Oracle Utilities Mobile Workforce Management application.

- OUMWMLIB

- OUMWMCONNECTEDMCPLIB

OUMWMLIB

The OUMWMLIB library comprises functions that work on the connected MCP browser UI. These functions mimic the actions of a field worker who updates the task/shift information into the application using the connected MCP.

The library is made up of a collection of actions, such as clicking various buttons, entering text into fields, navigating through the connected MCP UI, etc., used for creating the required functions. These functions are, in turn, used for creating the connected MCP components.

The functions use the object repository "MWM_ConnectedMCP_REPOSITORY.properties" that is in the "etc" directory of the Oracle Application Testing Suite for Oracle Utilities repository.

This section provides a list of functions included in the library, along with their usage details.

getTaskScheduledShiftId

Retrieves the Crew Shift ID to which a given task is scheduled.

This function checks continuously for the given time until it finds the crew shift that the activity is scheduled to. If the activity is not scheduled within the given time, it returns a blank string.

Example:

```
getTaskScheduledShiftId(String <timeInMinutes>, String <activityId>)

Input Parameters: String, String
Return Type: String
```

getPOUTaskIDForPOUId

Retrieves the period of unavailability Task ID for a given real-time period of unavailability ID.

This function checks continuously for the given time to retrieve the POU Task ID. If the POU Task ID is not created within the time, it returns a blank string.

Function Library Reference 2-1

Oracle Application Testing Suite for Oracle Utilities Reference Guide for Mobile Workforce Management and Oracle Real Time Scheduler

Example:

```
getPOUTaskIDForPOUId(String <timeInMinutes>, String <realPOUId>)

Input Parameters: String, String
Return Type: String
```

getAssignmentIDForTaskId

Retrieves the Assignment ID for a given Activity ID. The assignment will be created when the activity is in "Assignment In Progress" state.

This function checks for the Assignment ID for a given Activity ID until the given time. If an assignment is not created within the time, it returns a blank string.

Example:

```
getAssignmentIDForTaskId(String <timeInMinutes>, String <activityId>)

Input Parameters: String, String
Return Type: String
```

getAssignmentCurrentStatus

Retrieves the current status of an assignment.

This function returns a blank string if the assignment does not exist for the given ID.

Example:

```
getAssignmentCurrentStatus (String <assignmentId>)

Input Parameters: String
Return Type: String
```

verifyIfAssignmentIsDispatched

Verifies if an assignment has been dispatched or not.

This function checks until the provided time to assert if the assignment has been dispatched. If the assignment is dispatched, it returns "Y". Else, returns "N".

Example:

```
verifyIfAssignmentIsDispatched (String <timeInMinutes>, String
<assignmentId>)

Input Parameters: String, String
Return Type: String
```

checkActivityScheduledState

Verifies if an activity has been scheduled to any shift.

This function checks for the activity to be scheduled to a shift for a given time. If the activity is scheduled with in the given time, it returns "Y", else returns "N".

Example:

```
checkActivityScehduledState (String <timeInMinutes>, String
<activityId>)

Input Parameters: String, String
Return Type: String
```

getProcedureId

Retrieves the Procedure ID for a given procedure name and Crew Shift ID.

This function checks for a procedure to be created for the procedure name and Crew Shift ID within the given time. If the procedure ID is not found within the given time, it returns a blank string. Else, returns the procedure ID.

Example:

```
getProcedureId(String <procedureName>,String <crewShiftId>,String
<timeInMinutes>)

Input Parameters: String, String, String
Return Type: String
```

getTimesheetIdFromShiftId

Retrieves the Timesheet ID for a given Crew Shift ID.

This function checks for a time sheet to be created for the Crew Shift ID within the given time. If Timesheet ID is not found within the given time, it returns a blank string, else returns the Timesheet ID.

Example:

```
getTimesheetIdFromShiftId(String <crewShiftId>, String
<timeInMinutes>)

Input Parameters: String, String
Return Type: String
```

getVisitId

Retrieves the Visit IDs (comma separated) for a given Complex Activity ID.

If the Visit ID is not found, it returns a blank string. Else, it returns the Visit IDs.

Example:

```
getVisitId(String <complexActivityId>)

Input Parameters: String, String, String
Return Type: String
```

OUMWMCONNECTEDMCPLIB

The OUMWMCONNECTEDMCPLIB library contains functions that have been/can be used for creating Oracle Utilities Mobile Workforce Management components used to create the Oracle Utilities Mobile Workforce Management flows.

The library includes the Oracle Utilities Mobile Workforce Management specific functions that use the underlying framework of core functions. These functions interface with the database and read information, such as scheduling information of tasks, etc.

The library uses the UI element's xpath values specified in the MWM_ConnectedMCP_REPOSITORY.properties file in the **etc** directory of the repository.

> **Note:** The private functions in this library are used to implement the public functions. The private functions cannot be used directly in the Oracle Flow Builder scripts.

invokeApp

Launches the Web browser, navigates to the URL provided, and then checks for the object in the Web page.

If the object is found, the function returns "Y", else returns "N".

Example:

```
loginToMDT(String <URL>,String <objectToCheck>)

Input Parameters: String, String
Return Type: String
```

loginToMDT

Internally calls the invokeApp function with the MCP URL that is connected.

This function inputs the User ID and password, and logs into the MCP application that is connected. It returns "Y" on the successful login, else returns "N".

If the URL parameter is specified as "NA", the connected MCP URL is formulated using the application URL specified in the configuration.properties file by appending "/mobility/Login.jsp" to that URL. If the connected MCP URL is specified to this function, then it is used instead.

Example:

```
loginToMDT(String <username>, String <password>, String
<connectedMCPURL>)

Input Parameters: String, String, String
Return Type: String
```

downloadDeployment

Internally calls the loginToMDT function and waits for the deployment to download.

This function verifies if the **Start Shift** page is displayed within a minute. If the page is displayed, it returns "Y", else it returns "N".

If the URL parameter is specified as "NA", the connected MCP URL is formulated using the application URL specified in the configuration.properties file by appending "/mobility/Login.jsp" to that URL. If the connected MCP URL is specified to this function, then it is used instead.

Example:

```
downloadDeployment(String <username>, String <password>, String
<connectedMCPURL>)

Input Parameters: String, String, String
Return Type: String
```

startShift

Internally calls the downloadDeployment function and waits for the crew shift to start.

This function verifies if the Open Task List page is displayed. If the page is displayed, it returns "Y", else it returns "N". The loginError should be set as "N" for the crew shift to start successfully.

The function also verifies the negative scenarios where an error message is expected to be displayed during the shift start. The error message verification can be triggered by setting the loginError flag as "Y" as the input. If the loginError is set as "Y", the function checks for an error message to be displayed, returns the error message, and then logs off the crew shift. If no error message is retrieved, it then returns a blank string.

If the URL parameter is specified as "NA", the connected MCP URL is formulated using the application URL specified in the configuration.properties file by appending "/mobility/Login.jsp" to that URL. If the connected MCP URL is specified to this function, then it is used instead.

Example:

```
startShift(String <username>, String <password>, String
<connectedMCPURL>, String <loginError>)

Input Parameters: String, String, String, String
Return Type: String
```

verifyTaskDispatched

Verifies if a task has been dispatched for a given Task ID and then clicks on the task to open the UI.

This function returns a "Y" if the task has been dispatched. It checks for a task to be dispatched to the connected MCP within five minutes. If it is not dispatched, it returns a "N".

Example:

```
verifyTaskDispatched(String <tasked>)

Input Parameters: String
Return Type: String
```

enrouteActivity

Performs the actions of a field worker and moves the task to "Enroute" status using the connected MCP.

This function returns "Y" on success, else returns "N". To assign an activity, it has to be called after calling the verifyTaskDispatched function.

Example:

```
enrouteActivity ()

Input Parameters: <none>
Return Type: String
```

startActivityFromEnroute

Performs the actions of a field worker and moves the task to "Started" status using the connected MCP.

This function returns "Y" on success, else returns "N". To assign an activity, it has to be called after calling the enrouteActivity function.

Example:

```
startActivityFromEnroute ()

Input Parameters: <none>
Return Type: String
```

completeActivity

Performs the actions of a field worker and moves the task to "Completed" status using the connected MCP.

This function returns "Y" on success, else returns "N". To assign an activity, it has to be called after calling the startActivityFromEnroute function.

Example:

```
completeActivity ()

Input Parameters: <none>
Return Type: String
```

startBreak

Performs the actions of a field worker and moves the break task status to "Started" using the connected MCP.

This function returns "Y" on success, else returns "N". To assign a break task, it has to be called after calling the verifyTaskDispatched function.

Example:

```
startBreak ()

Input Parameters: <none>
Return Type: String
```

completeBreak

Performs the actions of a field worker and moves the break task status to "Completed" using the connected MCP.

This function returns "Y" on success, else returns "N". To assign a break task, it has to be called after calling the startBreak function.

Example:

```
completedBreak ()

Input Parameters: <none>
Return Type: String
```

enrouteNPT

Performs the actions of a field worker and moves the non productive task (NPT) status to "Enroute" using the connected MCP.

This function returns "Y" on success, else returns "N". To assign a non productive task, it has to be called after calling the verifyTaskDispatched function.

Example:

```
enrouteNPT ()

Input Parameters: <none>
Return Type: String
```

startNPT

Performs the actions of a field worker and moves the non productive task (NPT) status to "Started" using the connected MCP.

This function returns "Y" on success, else returns "N". To assign a non productive task, it has to be called after calling the enrouteNPT function.

Example:

```
startNPT ()

Input Parameters: <none>
Return Type: String
```

enroutePOU

Performs the actions of a field worker and moves the period of unavailability (POU) task status to "Enroute" using the connected MCP.

This function returns "Y" on success, else returns "N". To assign a POU task, it has to be called after calling the verifyTaskDispatched function.

Example:

```
enroutePOU ()

Input Parameters: <none>
Return Type: String
```

startPOU

Performs the actions of a field worker and moves the period of unavailability (POU) task status to "Started" using the connected MCP.

This function returns "Y" on success, else returns "N". To assign a POU task, it has to be called after calling the enroutePOU function.

Example:

```
startPOU ()

Input Parameters: <none>
Return Type: String
```

completePOU

Performs the actions of a field worker and moves the period of unavailability (POU) task status to "Completed" using the connected MCP.

This function returns "Y" on success, else returns "N". To assign a POU task, it has to be called after calling the startPOU function.

Example:

```
completePOU()

Input Parameters: <none>
Return Type: String
```

endShift

Completes/ends the crew shift by performing the actions of a field worker to complete the current day's crew shift using the connected MCP.

This function also closes the connected MCP application. It returns "Y" on success, else returns "N".

Example:

```
endShift()

Input Parameters: <none>
Return Type: String
```

Chapter 3

Sample Work Flows

This chapter describes the Oracle Utilities Mobile Workforce Management sample flows that illustrate common use cases for Oracle Application Testing Suite for Oracle Utilities. It also explains the procedure to execute these sample flows.

The chapter includes the following sections:

- Sample Flows
- Executing Sample Flows

Sample Flows

The three sample flows delivered as part of Oracle Application Testing Suite for Oracle Utilities for Oracle Utilities Mobile Workforce Management demonstrate how flows can be created for Web services based testing and for a combination of Web services and UI based testing using the same framework.

These flows are designed to run using the demo data, there by giving the user, the ability to deploy Oracle Application Testing Suite for Oracle Utilities for Oracle Utilities Mobile Workforce Management and execute the sanity flows immediately. The flows perform a part of the basic sanity testing required to certify that the Oracle Utilities Mobile Workforce Management environment has been setup appropriately.

This section includes the following sample work flows:

- Non-MDT Flow
- MDT Flow
- M2 Non-MDT Flow

Non-MDT Flow

The Non-MDT flow comprises the creation and completion life cycle of an M1 activity - break, non-productive task (NPT), and real period of unavailability (POU), along with the crew shift.

The following table lists the tasks that are created and their respective components in Oracle Utilities Mobile Workforce Management:

Task	MWM Component
Activity	M1-Activity
Break	M1-Break

Task	MWM Component
Non-productive Task	M1-NonProductiveTask
Real Period of Availability	M1-RealPOU

Activity, real POU, break, and non-productive tasks are scheduled to a planned crew shift that has the matching location, required capability, and scheduling time window. The dispatcher logs in the planned crew shift and starts the shift. Once the crew shift starts, activity, non-productive task, real POU, and break are dispatched to the crew shift. The dispatcher starts, completes the activity, real POU, break, and non -productive task.

The M1-Assignment component is used to assign a transition (to enroute --> start --> completing an activity). Similarly, M1-BreakTask, M1-NonProductiveTask, and M1-POUTask components are used to start, complete the break, non-productive tasks, and POU respectively. The dispatcher completes the crew shift after completing the dispatched tasks.

MDT Flow

The MDT flow includes the creation and completion of an activity, break, non-productive task (NPT), and real period of unavailability (POU) using connected MCP.

Task	MWM Component
Activity	M1-Activity
Break	M1-Break
Non-productive Task	M1-NonProductiveTask
Real Period of Availability	M1-RealPOU

Activity, real POU, break, and non-productive tasks are scheduled to a planned crew shift that has the matching location, required capability, and scheduling time window.

The MDT flow is as follows:

1. The dispatcher launches the connected MCP.

2. The crew logs in with the valid credentials and starts the crew shift.

3. The activity, break task, and non -productive tasks are dispatched to the crew shift.

4. Use the following components applicable for each task:

Task	MWM Component
Start the crew shift	ConnectedMCP_StartCrewShift
Verify if the tasks are dispatched	ConnectedMCP_DispatchedTask
Enroute status	ConnectedMCP_EnrouteActivity
Start activity	ConnectedMCP_StartActivity
Complete activity	ConnectedMCP_CompleteActivity

5. The crew shift is completed after completing all the dispatched tasks.

M2 Non-MDT Flow

The M2-Non-MDT flow includes the creation and completion of M-Activity (Install Meter Activity, for example).

Create an Install Meter Activity using the **M2-InstallMeterActivity** component.

The M2-Activity is scheduled to a planned crew shift that has the matching location, required capability, and the scheduling time window.

The M2 Non-MDT flow is as follows:

1. The dispatcher logs in the planned crew shift.

2. The dispatcher starts and completes the M2-Activity.

   The M2-InstallMeterAssignment component is used for M2-Activity assignment transition.

3. The dispatcher completes the crew shift after completing the M2-Activity (Install Meter Activity).

Executing Sample Flows

This section describes the procedure to setup sample flows and execute them.

• Pre-requisites

• Setting Up Sample Flows

Pre-requisites

To execute the sample flow, ensure the following pre-requisites are met:

• Oracle Utilities Mobile Workforce Management v2.2.0.1/Oracle Real-Time Scheduler v2.2.0.1 is up and running with the demo data pack.

• OpenScript is installed in the local machine. See the *Oracle Application Testing Suite for Oracle Utilities Administration and User's Guide* for the version details.

• Oracle Application Testing Suite for Oracle Utilities is installed and repository/directory is setup in the local machine appropriately. See the *Oracle Application Testing Suite for Oracle Utilities Administration and User's Guide* for more details.

Setting Up Sample Flows

To setup a sample flow, follow these steps:

1. Login to Oracle Utilities Mobile Workforce Management/Oracle Real-Time Scheduler.

2. Import the XAI inbound services into the Oracle Utilities Mobile Workforce Management/ Oracle Real-Time Scheduler application where the scenarios need to be executed.

   See the Importing XAI Inbound Services section in Appendix A: XAI Inbound Services for steps to import the XAI inbound services.

3. Navigate to **Admin > X > XAI Command**.

4. Select **Refresh Registry** in the drop-down list.

5. Click **Send Command**.

6. Launch OpenScript in the local machine and perform the following steps:

   a. Navigate to **View > OpenScript Preferences**.

   b. In the left tree, select **OpenScript**. In the sub tree, select **Playback**, and then select **Error Recovery**.

    c. Click **SetAll** and select **Report Error and Continue**.

    d. Click **Apply**, and then click **Close**.

7. Copy **MWM_ConnectedMCP_REPOSITORY.properties** and **configuration.properties** files to the **etc** folder in the Oracle Application Testing Suite repository directory from the installer.

8. Configure the **configuration.properties** file as follows:

    a. Provide the application URL for the parameter:

```
gStrApplicationURL = http\://<%serverName%>\:<%portNumber%>/
ouaf
```

    b. Provide the additional path required for webservice/XAI inbound service URL:

```
gStrApplicationXAIServerPath=/<%AppendThisToAbove
gStrApplicationURL %>/<%AppendThisToAbove gStrApplicationURL
%>/
```

    c. Provide an environment name for display in the results email:

```
gStrEnvironmentName= <%testEnvironmentName%>
```

    d. Provide the application login user ID:

```
 gStrApplicationUserName= <%UserNAme%>
```

    e. Provide the application login password:

```
gStrApplicationUserPassword= <%password%>
```

    f. Provide the SMTP email server and e-mail ID:

```
gStrSMTP_HOST_NAME=<%SMTP ServerName%>
gStrSMTP_PORT=<%PortNumber%>
gStrTO_EMAIL_RECIPIENTS=<%e-mail Id%>
```

    g. Provide the application database details as below:

```
gStrApplicationDBConnectionString =<%jdbc Connectionstring%>
eg: jdbc\:oracle\:thin\:@<%DBserverName%>\:<%port%>\:<%DBSID%>
gStrApplicationDBUsername=<%DBUserID%>
gStrApplicationDBPassword=<%DBPassword%>
```

    h. Provide the full directory path of Oracle Application Testing Suite repository directories in the local machine.

```
gStrOutputFilePath=<%LogFilePath%>
Example: C:\\OATSOUMWM_DEMO\\OUTSP\\Logs\\
gStrXSDFiles=<%XSD Folder path%>
Example: C:\\OATSOUMWM _DEMO\\OUTSP\\Logs\\
```

9. Copy the following function libraries (scripts from the installer) into the **outsp-function-libs** folder in the Oracle Application Testing Suite repository directory.

- OUMWMCONNECTEDMCPLIB

- OUMWMLIB

- OUTSPCORELIB

- WSCOMMONLIB

- WSVALIDATELIB

10. Copy all the .jar files provided in the installer into the **genericJars** folder in the Oracle Application Testing Suite repository directory.

While executing the MasterDrive script to run the sample flows, an error "Child script with alias <%Script Name%> not found" is displayed. This prevents the script from executing further.

To resolve this error, follow these steps:

1. Open the **MasterDrive** script for execution.

2. Navigate to the **Asset** tab.

3. Select **Scenario** and then click **Open**. Close the **Scenario** child script after it is open.

4. Execute the **MasterDrive** script again. This resolves the issue.

Appendix A

XAI Inbound Services

The Oracle Utilities Mobile Workforce Management components are developed using Web services method, and these components require XAI Inbound Services to be defined in the application.

This chapter contains the following sections:

- Creating XAI Inbound Services
- Sample XAI Inbound Services
- Importing XAI Inbound Services
- List of XAI Inbound Services

Creating XAI Inbound Services

To create an XAI Inbound Service, follow these steps:

1. Login to the Oracle Utilities Mobile Workforce Management application.

2. Navigate to **Admin > X > +XAI Inbound Service**.

3. On the **XAI Inbound Service** page, do the following:

    a. Enter the desired **XAI Inbound Service Name**.

    b. Select **BusinessAdaptor** as the **Adapter**.

    c. Select **Business Object** as the **Schema Type**.

    d. Select the appropriate **Schema Name**.

    e. Enter the **Description** and **Detailed Description**.

    f. Select **Add** as the **Transaction Type**.

    g. Click **Save**.

Sample XAI Inbound Services

Below are the details of a sample XAI Inbound Service created in the Oracle Utilities Mobile Workforce Management application.

- **XAI Inbound Service Name**: CM_M1-Vehicle
- **Adapter**: BusinessAdaptor
- **Schema Type**: Business Object
- **Schema Name**: M1-Vehicle

XAI Inbound Services A-1

Oracle Application Testing Suite for Oracle Utilities Reference Guide for Mobile Workforce Management and Oracle Real Time Scheduler

- **Description**: CM_M1-Vehicle

- **Detailed Description**: CM_M1-Vehicle

- **Active**: <Check this option>

- **Transaction Type**: Add

Importing XAI Inbound Services

To import an XAI Inbound Service into the Oracle Utilities Mobile Workforce Management application, follow these steps:

1. Ensure the exported XAI services are available in the local machine.

2. Login to the Oracle Utilities Mobile Workforce Management application.

3. Click **Admin > X > XAI Service Import**.

4. On the **XAI Service Import** page, click **Browse**.

5. Select the location of the file and click **Open**.

6. On the **XAI Service Import** page, click **Read**.

7. Select the **Import?** check box for the respective XAI Inbound Service Name.

8. Click **Save**. The "Imported Successfully" message appears in the **Message text** column.

List of XAI Inbound Services

This section provides a list of XAI Inbound Services that caters the need of required XAI Inbound Services to use the delivered components and flows.

XAI Inbound Service Name	Description
CM_M1-Activity	CM_M1-Activity
CM_M1-Alert	CM_M1-Alert
CM_M1-AllocateActivityToAShift	CM_M1-AllocateActivityToAShift
CM_M1-AppointmentBookingRequest	CM_M1-AppointmentBookingRequest
CM_M1-Assignment	CM_M1-Assignment
CM_M1-BreakTask	CM_M1-BreakTask
CM_M1-CommonShiftWeeklyTemplate	CM_M1-CommonShiftWeeklyTemplate
CM_M1-Company	CM_M1-Company
CM_M1-ComplexActivity	CM_M1-ComplexActivity
CM_M1-Contractor	CM_M1-Contractor
CM_M1-ContractorEligibility	CM_M1-ContractorEligibility
CM_M1-Crew	CM_M1-Crew
CM_M1-CrewShift	CM_M1-CrewShift
CM_M1-CrewShiftCloseOpenAllocate	CM_M1-CrewShiftCloseOpenAllocate
CM_M1-CrewShiftTemplate	CM_M1-CrewShiftTemplate

XAI Inbound Service Name	Description
CM_M1-DailySpeedTemplate	CM_M1-DailySpeedTemplate
CM_M1-DeferActivity	CM_M1-DeferActivity
CM_M1-DepotRelatedActivity	CM_M1-DepotRelatedActivity
CM_M1-DepotRelatedShift	CM_M1-DepotRelatedShift
CM_M1-DepotSinglePersonCrew	CM_M1-DepotSinglePersonCrew
CM_M1-DepotTask	CM_M1-DepotTask
CM_M1-Dispatcher	CM_M1-Dispatcher
CM_M1-DispatcherShift	CM_M1-DispatcherShift
CM_M1-FixToCrew	CM_M1-FixToCrew
CM_M1-ForceLogOffCrewShift	CM_M1-ForceLogOffCrewShift
CM_M1-GetShiftCapabilities	CM_M1-GetShiftCapabilities
CM_M1-HazardArea	CM_M1-HazardArea
CM_M1-MailingList	CM_M1-MailingList
CM_M1-MainMail	CM_M1-MainMail
CM_M1-MobileWorker	CM_M1-MobileWorker
CM_M1-NonPrdTask	CM_M1-NonPrdTask
CM_M1-OverrideTimeWindow	CM_M1-OverrideTimeWindow
CM_M1-POUTask	CM_M1-POUTask
CM_M1-RealPOU	CM_M1-RealPOU
CM_M1-ResTimesheet	Timesheet
CM_M1-ResourceLeave	CM_M1-ResourceLeave
CM_M1-SchedulerRegistryRead	CM_M1-SchedulerRegistryRead
CM_M1-ShiftWeeklyTemplate	CM_M1-ShiftWeeklyTemplate
CM_M1-SimpleProcedure	CM_M1-SimpleProcedure
CM_M1-SinglePersonCrew	CM_M1-SinglePersonCrew
CM_M1-SpeedProfileTemplate	CM_M1-SpeedProfileTemplate
CM_M1-SubscriptionWeeklyTemplate	CM_M1-SubscriptionWeeklyTemplate
CM_M1-TaskScheduleDetails	CM_M1-TaskScheduleDetails
CM_M1-TemplateMail	CM_M1-TemplateMail
CM_M1-TemplatePOU	CM_M1-TemplatePOU
CM_M1-UnassignActivity	CM_M1-UnassignActivity
CM_M1-Vehicle	CM_M1-Vehicle
CM_M1-VehicleLeaveArea	CM_M1-VehicleLeaveArea

XAI Inbound Service Name	Description
CM_M2-InstallMeterActivity	CM_M2-InstallMeterActivity
CM_M2-InstallMeterAssignment	CM_M2-InstallMeterAssignment
CM_M1-ActivityType	CM_M1-ActivityType
CM_M1-AlertType	CM_M1-AlertType
CM_M1-BreakTaskType	CM_M1-BreakTaskType
CM_M1-Capacity	CM_M1-Capacity
CM_M1-CapacityTemplate	CM_M1-CapacityTemplate
CM_M1-CapacityType	CM_M1-CapacityType
CM_M1-CapacityWeeklyTemplate	CM_M1-CapacityWeeklyTemplate
CM_M1-ComplexActivityType	CM_M1-ComplexActivityType
CM_M1-ContractorType	CM_M1-ContractorType
CM_M1-CrewHierarchy	CM_M1-CrewHierarchy
CM_M1-CrewShiftType	CM_M1-CrewShiftType
CM_M1-CrewType	CM_M1-CrewType
CM_M1-Deployment	CM_M1-Deployment
CM_M1-DeploymentPart	CM_M1-DeploymentPart
CM_M1-DeploymentType	CM_M1-DeploymentType
CM_M1-Depot	CM_M1-Depot
CM_M1-DepotProfile	CM_M1-DepotProfile
CM_M1-DepotTaskType	CM_M1-DepotTaskType
CM_M1-DispatchArea	CM_M1-DispatchArea
CM_M1-DispatcherType	CM_M1-DispatcherType
CM_M1-Equipment	CM_M1-Equipment
CM_M1-GeographicArea	CM_M1-GeographicArea
CM_M1-GlobalConfigurations	CM_M1-GlobalConfigurations
CM_M1-KPI	CM_M1-KPI
CM_M1-Location	CM_M1-Location
CM_M1-MobileDeviceTerminal	CM_M1-MobileDeviceTerminal
CM_M1-MobileDeviceTerminalType	CM_M1-MobileDeviceTerminalType
CM_M1-MobileWorkerType	CM_M1-MobileWorkerType
CM_M1-NonPrdTaskType	CM_M1-NonPrdTaskType
CM_M1-POUTaskType	CM_M1-POUTaskType
CM_M1-POUType	CM_M1-POUType

XAI Inbound Service Name	Description
CM_M1-PriorityProfile	CM_M1-PriorityProfile
CM_M1-ProcedureType	CM_M1-ProcedureType
CM_M1-RemarkType	CM_M1-RemarkType
CM_M1-ResTimesheetTyp	CM_M1-ResTimesheetTyp
CM_M1-Scheduler	CM_M1-Scheduler
CM_M1-SchedulerArea	CM_M1-SchedulerArea
CM_M1-SchedulerConfiguration	CM_M1-SchedulerConfiguration
CM_M1-ServiceArea	CM_M1-ServiceArea
CM_M1-ServiceAreaHierarchy	CM_M1-ServiceAreaHierarchy
CM_M1-ServiceClass	CM_M1-ServiceClass
CM_M1-ShiftCostProfile	CM_M1ShiftCostProfile
CM_M1-Skill	CM_M1-Skill
CM_M1-StatusReason	CM_M1-StatusReason
CM_M1-VehicleType	CM_M1-VehicleType
CM_M1-WorkCalendar	CM_M1-WorkCalendar
CM_M1-WorkProfile	CM_M1-WorkProfile
CM_M2-MaintainUtilityActByHost	CM_M2-MaintainUtilityActByHost