SAP Cloud Platform Integration

Technical Specification

Technical Specifications Document

Document Release Note

Document Name:	WA Conflict Resolution - Execute Actions
Version:	1.0.0
Description:	ACTION EXECUTOR: Executes conflict resolution actions in SuccessFactors based on analysis output. Performs three types of operations: (1) Delete work assignments by setting status to CANCELLED, (2) Delete timesheet events (C10/C20), (3) Insert new timesheet events for trimmed times.
Release Date:	2025-11-06

Revision History

Revision	Date	Description	Page	Rationale	Type
1.0	2025-11-06	Initial Draft	All	Initial Version	Add

Document Contact Information

Name:	Abdelrahman Hussein
Role:	Technical Consultant

Table of Contents

1. BUSINESS CONTEXT

1.1 Overview

ACTION EXECUTOR: Executes conflict resolution actions in SuccessFactors based on analysis output. Performs three types of operations: (1) Delete work assignments by setting status to CANCELLED, (2) Delete timesheet events (C10/C20), (3) Insert new timesheet events for trimmed times.

1.2 Development Unit Information

Module	SAP Cloud Platform
Sub Module	Hana Cloud Integration
iFlow Title	WA Conflict Resolution - Execute Actions
Processing Type	Background Online
Execution Frequency	On-Demand (Called by orchestrator)

2. DETAILED DESIGN

2.1 Configuration Details

Package Name: SF-Nadec-WorkAssignment
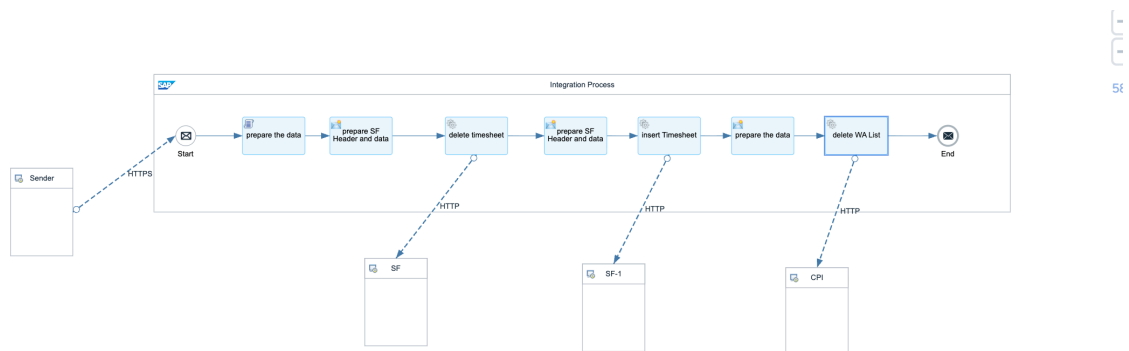
iFlow Name: WA Conflict Resolution - Execute Actions

Technical Name: WA_TS_Conflict_Execute

Endpoint: /resolveWAConflictAction

2.2 SAP CPI iFlow Design

This is a custom-designed SAP CPI Integration flow for work assignment and timesheet conflict resolution.

Detailed Requirements:

1. Receive resolution output from Analyze Logic flow
2. Extract delete lists: timesheetDelete, workAssignmentDelete
3. Extract insert list: timeEventInsert (for trimmed times)
4. Execute work assignment deletions (calls Delete Work Assignment flow)
5. Execute timesheet deletions via SF TimeEvent OData API
6. Create new timesheet events for trimmed times via SF TimeEvent API
7. Handle batch operations for multiple records
8. Return execution results with success/failure counts

Groovy Scripts

Script Name	Description
Parse Resolution Data	Extracts delete and insert arrays from input JSON
Execute WA Deletions	Transforms WA IDs to XML, calls Delete Work Assignment flow
Execute TS Deletions	Formats C10/C20 IDs for SF OData delete API, executes batch delete
Create Trimmed Events	Generates new TimeEvent records with adjusted times (±1 min), posts to SF
Aggregate Results	Counts successful/failed operations, structures response

2.3 Adapter Configuration (Sender & Receiver)

Receiver (SF): Multiple operations - Delete WA: Calls SF_WorkAssignment_Delete flow - Delete TS: SF TimeEvent OData API (DELETE) - Insert TS: SF TimeEvent API (POST) Authentication: Basic + OAuth Bearer

2.4 Error Handling

Standard CPI error handling applies. Errors are logged to message processing log. Failed messages are stored in error queue for manual intervention. Retry mechanism is configured for transient failures (3 attempts with 5-second delay).

3. TESTING

3.1 Test Conditions and Expected Results

Test Condition	Expected Result
10 WA deletions, 5 TS deletions, 3 TS inserts	All operations executed successfully
SF API rejects TS insert (validation error)	Error captured, other operations continue
Duplicate TS event in insert list	SF handles duplicate, returns appropriate error
Network timeout during execution	Retry mechanism handles timeout, resumes from last successful operation

3.2 Test Data Considerations

Test data should include: (1) Typical scenarios with standard work assignments and timesheets, (2) Edge cases with time zone boundaries, (3) Error scenarios with malformed data, (4) Load testing with bulk data volumes.

3.3 Performance Considerations

Expected processing time: <5 seconds for single record, <60 seconds for batch of 100 records. SF API rate limits: 5000 calls/hour. Memory usage: <500MB for typical batch operations.

4. APPENDIX

Additional technical documentation, API specifications, and code samples are available in the project repository: https://github.com/eco-nadec/SF-WA-CPI