

**Draft Quote Submission (eDTK - DAD Upload)**

**Interface Detail Design Document**

**CR 15977**(Process DADs Quotes/Orders through Siebel in the USA, Nordics and Canada)

**eDTK – Draft Quote Submission to Siebel**

IB Interfaces: Draft Quote Submission (**eDTK-EAI-CRM**)

[1) Approvals 3](#_Toc292119733)

[2) Review List 3](#_Toc292119734)

[3) Related Documentation 3](#_Toc292119735)

[4) Glossary 3](#_Toc292119736)

[5) Introduction 4](#_Toc292119737)

[**5.1** **Document Context** 4](#_Toc292119738)

[**5.2** **Document Structure** 4](#_Toc292119739)

[**5.3** **Scope** 4](#_Toc292119740)

[**5.4** **Out of Scope** 5](#_Toc292119741)

[**5.5** **Assumptions** 5](#_Toc292119743)

[6) User Contacts 5](#_Toc292119744)

[**6.1** **Interface Definition** 5](#_Toc292119745)

[6.1.1 Interface Purpose 5](#_Toc292119746)

[6.1.2 Detailed Interface Mapping 6](#_Toc292119747)

[6.1.3 Context 7](#_Toc292119748)

[6.1.4 Functional View 7](#_Toc292119749)

[6.2 Interface Characteristics 9](#_Toc292119750)

[**6.2** **Interface Design** 9](#_Toc292119751)

[6.2.1 High Level Interface Design 9](#_Toc292119752)

[6.2.2 Data Definitions 11](#_Toc292119753)

[**6.3** **Interface Design Components** 11](#_Toc292119754)

[6.3.1 EDTK\_DequeueDraftQuote 11](#_Toc292119755)

[6.3.2 CCRMDAD\_SubmitDraftQuoteMessage 12](#_Toc292119757)

[6.3.3 PL-SQL 14](#_Toc292119760)

[7) Developer Notes 14](#_Toc292119761)

[ **Code management** 14](#_Toc292119762)

[ **Deployment Process:** 14](#_Toc292119763)

[8) Issues/Workarounds 14](#_Toc292119764)

Document History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Change Description | Author | Date Released |
| 1.0 | First draft | Rijoy Purayil | Mar 03,2011 |
| 1.1 | Updated | Rijoy Purayil | Mar 30,2011 |

# Approvals

The individuals listed below will be required to review and approve this document.

|  |  |  |  |
| --- | --- | --- | --- |
| Role | Name | Signature | Date |
|  |  |  |  |
|  |  |  |  |

# Review List

|  |  |
| --- | --- |
| Name | Position |
|  |  |

# Related Documentation

| Ref | Title | Author | Version |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Glossary

| Abbreviation | Definition |
| --- | --- |
| **CCRM** | Commerce and Customer Relationship Management |
| **EAI** | Enterprise Application Integration |
| **eDTK** | electronic Data-feed Tool Kit |
| **DAD** | Data Feed Access Declarations |

# Introduction

##### **Document Context**

This document is one of a set of Interface Definition Documents providing the definition and High Level Design of the interfaces required for the eDTK DAD upload to Siebel via EAI Middleware governed under the Release R5. This document details the interfaces which support the propagation of draft quote message.

This document also covers the high level design documentation for new interfaces which would support the inbound interface that would enable the eDTK to send the draft quote message to EAI that will pass it onto Siebel.

The eDTK is a country based system (aligned to core suite instances) and will be expected to upload the data prior to the working day in the region in which it is based.

eDTK will capture DADs and OMS will trigger eDTK to create a draft quote in Siebel.

##### **Document Structure**

This document describes about the requirements of an interface in terms of scope, functionality and characteristics. This document focus on what the interface does from an end-to-end perspective. Also covers the design of interface and the technical integration implementation.

##### **Scope**

* This document covers the interface design that will support the Enterprise Application Integration (EAI) solution that would provide the ability to the external system eDTK to upload the DAD differences to Siebel CRM as draft Quote Message.
* The process flow described in the following sections depicts the processing of DAD xml payload to create a generic quote message and its hat can be accepted by Siebel CRM and their subsequent propagation to Siebel inbound queue.
* This document also emphasis on strategic business decision where in the dad differences for the EDTK implemented countries would be propagated to Siebel to create draft Quote Messages.
* As per the agreed design, EDTK does not receive any Acknowledgement from EAI. eDTK accepts successful message delivery once the process to put reply in MQ is complete at EDTK side.
* Similarly once the Draft Quote Message has been en-queued to Siebel inbound Queues, EAI would not receive any acknowledgement from Siebel for Successful Quote creation.

##### **Out of Scope**

##### -

##### **Assumptions**

* The Integration assumes that the incoming DAD Xml Payloads are validated by the source system i.e; e-DTK and EAI would not perform any additional xml validation for the incoming request messages.
* The integration expects a reliable message exchange via MQ connectivity between eDTK and EAI for inbound messages from EDTK and a similar connectivity between EAI and Siebel CRMing for propagating the same as generic draft quote message.

# User Contacts

##### **Interface Definition**

* 1. Interface Purpose

eDTK will capture DADs and OMS will trigger eDTK to create a draft quote in Siebel. A generic draft quote interface will be developed on Siebel front ,so that it can be reused for other systems to create quote requirements. Here in this case eDTK will send the draft quote message to EAI ,which would pass it onto Siebel.

The purpose of this interface design Document is to allow eDTK application to Upload the DAD Difference to Siebel to create Draft Quotes in Siebel.

Siebel won’t be sending any kind of response acknowledgements for the eDTK-DAD uploads propagated via EAI.

eDTK Application wouldn’t expect any kind of response for the successfully uploaded DAD.

The eDTK DAD Difference Xml payload structure consist of DAD envelope that encloses a DAD header and multiple Quote Line Items associated to it. Each DAD payload is identified by a unique DAD Id which would be part of DAD Header section.

The Status field associated to each quote item in DAD xml payload determines the Action code and Operation associate to each Quote line Item in generic Quote interface.

|  |  |  |
| --- | --- | --- |
| Status(eDTK) | Action Code(Siebel CRM) | Operation – (Siebel CRM) |
| I | Add | New |
| R | Delete | Cancellation |

* 1. Context

The B2CC wave 2 release will rollout SAP billing into the USA and Nordics and Customer Administration require the ability to invoice customers for orders raised as a result of Data Feed Access declarations (DADs) being returned to Thomson Reuters.

Currently the DADS orders raised in the USA and Nordics are processed through core suite and the resultant assets are RCIP’d to Siebel, therefore no orders are raised through Siebel for DADs activities.  
  
In Canada (B2CC wave 1) there is already a process allowing DADs orders to generate invoices from SAP. The process is not sustainable for customer administration when SAP is rolled out to the USA due to the significant increase in manual work as compared with the previous core suite based ordering process.

* 1. Detailed Interface Mapping

The below tabular structure shows the Mapping details of EAI Integrationbetween eDTK and Siebel CRM along with the defaulted values and conditions to be checked during xslt transformation..

Tabular structure shows, eDTK fields expected in DAD XML payload, the corresponding Siebel generic schema fields, Field those are defaulted in Siebel CRM , whether Mapping needs to be done in EAI or not, and which all values need to be defaulted in EAI during DAD to Draft quote message Transformation. The highlighted fields are received in EAI and Transformed as per the detailed Master Mapping sheet.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **eDTK Fields** | **Siebel Fields** | **Default in Siebel** | **Mapping in EAI** | **Default in EAI** |
| STATUS | ActionCode |  | **Y** | if the value is "I" EAI will default as "Add" and if its "R" EAI will default as "Delete" |
| DAD Declared Date | BillingStartDate |  | **Y** |  |
| DAD Declared Date | BillingStopDate |  | **Y** |  |
| PLI | GlobalPLI |  | **Y** |  |
| QUANTITY | InstalledQuantity |  | **Y** |  |
| PLI\_COUNT | LineStatus |  | **Y** | If the PDP has more than one PLI count EAI will send the status as "Pending OMS Review" else "Ready" |
| STATUS | Operation |  | **Y** | if the value is "I" EAI will default as "New" and if its "R" EAI will default as "Cancellation" |
| PDP\_CODE | PDPCode |  | **Y** |  |
| DIFFERENCE | QuantityRequested |  | **Y** |  |
| COST\_CENTRE | SubAccountNumber |  | **Y** |  |
| CLIENT\_NO | SubscriberNumber |  | **Y** |  |
| DAD ID | TransactionId | Siebel will concatenate the DAD ID to Quote Id and show in Quote Name | **Y** |  |

* + 1. Functional View

|  |
| --- |
|  |

* **Sequence Diagram:**

The Below Sequence Diagram shows the various EAI components involved in the DAD xml Upload from eDTK to Siebel CRM.

It also shows the MQ Components at the source and target end for ,



* 1. Interface Characteristics
  + The Generic Quote Interface for uploading the Draft Quote Message is offered by EAI to eDTK for DAD Difference Upload to Siebel. Real time interface from eDTK to draft Quote Message in Siebel would be built in such a way where in Siebel would apply the duplicate resolution logic to identify whether it needs to create a Contact or update an existing contact.
* Siebel to send the acknowledgements to EAI for the contact created/updated from TOM.

TOM to store the error messages if any.

| eDTKEAISIEBEL | | | | |
| --- | --- | --- | --- | --- |
| Process | Source | Target | Mechanism | Description |
|  |  |  |  |  |
|  |  |  |  |  |
| DAD xml Upload between eDTK and EAI | eDTK | **EAI Fusion Middleware** | MQ Service | eDTK to en-queue the DAD xml to Region Based Remote Queues managed by eDTK Queue Manger |
| Draft Quote Message Upload between EAI and Siebel | **EAI Fusion Middleware** | Siebel CRM | MQ Service | EAI to en-queue the transformed DAD xml to send a Draft Quote message to Siebel. |

##### **Interface Design**

* 1. High Level Interface Design



* 1. Data Definitions

The message specifications used in the interfaces between EAI and the external systems are defined in the API specification [6]. Latest versions of the XML schemas for the messages used in the interface are stored in the Reuters Developer Network Subversion repository (https://int.thomsonreuters.com) project called sami (https://sami.cdt.int.thomsonreuters.com). The definitions for this interface can be found in subversion at:

https://sami.cdt.int.thomsonreuters.com/svn/eai\_eai/branches/6.1/design/EAI1.0/Interfaces

##### **Interface Design Components**

|  |  |  |
| --- | --- | --- |
| **Module Name/Method** | **Module Type** | **Existing/New** |
| **EDTK\_DequeueDraftQuote** | BPEL Process | New BPEL Component |
| **CCRM\_SubmitDraftQuoteMessage** | BPEL Process | New BPEL Component |
|  |  |  |

* + 1. EDTK\_DequeueDraftQuote
* **Purpose**

This module provides the De-queue component that would get the DAD Difference xml payload from the Local Message Queue. These Queues would be populated by the external system eDTK as part of their XML upload to their remote Queues

* **Program Logic**

This EAI service receives the DAD xml payload of the schema structure EAI.MSG.eDTK.QM.307\_DraftQuoteMessage.xsd from Local Message Queue and Transform to EAI’s internal canonical structure for Draft Quote Message in the form of EAI.MSG.CCRM.DAD.309\_DraftQuotePublishRequest.xsd. The request message will be transformed for the changes expected by the target system.

This service invokes the CCRM\_SubmitDraftQuoteMessage BPEL Service using SOAP over HTTPS. The transformed canonical structure will the passed on to this component for propagation to end system.

* **DVM**

-NONE-

* **Data Mapping (Transformation)**

The EAI service receives the DAD xml payload in the form of EAI.MSG.eDTK.QM.307\_DraftQuoteMessage.xsd and the same is transoformed the EAI’s internal canonical structure for draft quote (EAI.MSG.CCRM.DAD.309\_DraftQuotePublishRequest.xsd )using the xslt transformation.

* + - XformDADReq\_To\_DraftQuoteNew.xsl
* **Scope:**

ValidateXML

* **Exception Handling:**

For all the scopes in EDTK\_DequeueDraftQuote catch all types of exceptions and throwing the fault message.

* + *Selection Failure****:***

For any Selection Failure fault exception while transformation catches the exception and throwing remote fault exception and appending the fault message to the title.

* + *remoteFault:*

For any Remote fault exception while transformation catches the exception and throwing remote fault exception and appending the fault message to the title.

* + *runtimeFault:*

For any run time fault exception while transformation catch the exception and throwing run time fault exception and appending the fault message to the title.

* + *bindFault:*

#### For any data binding or data validations exceptions while transformation catch the exception and throwing remote fault exception and appending the fault message to the title.

NOTE: After Fault the BPEL instance is ready for Re-initiate, Manually user/Ops team will be re-initiating after all validations done.

* **Input/output Schemas:**
* EAI.MSG.eDTK.QM.307\_DraftQuoteMessage
* EAI.MSG.CCRM.DAD.309\_DraftQuotePublishRequest.xsd
* **Partner links**
* [DequeueDADService.wsdl](http://eaiblue.ime.reuters.com:7777/orabpel/ccrmasync/EDTKDAD_DequeuedraftQuote/R5.0_15.1/DequeueDADService.wsdl)
* CCRMDAD\_SubmitDraftQuoteMessage
  + 1. CCRMDAD\_SubmitDraftQuoteMessage
* **Purpose**

This process receive the canonical message structure for the draft quote message and does the necessary payload transformation before propagating the same to Siebel CRM via MQ.

* **Program Logic**

This EAI service receives the canonical message structure in the form of as input structure EAI.MSG.CCRM.DAD.309\_DraftQuotePublishRequest.xsd and is transformed to Siebel’s Generic quote schema structure prior to invoking the MQ for submitting the Draft quote message.

This transformation is achieved through xsl transformation.

* **DVM**

-NONE-

* **Data Mapping (Transformation)**

The necessary message transformations are performed by the following xsl:

TransDraftReqMsgToEnqueueDraftQuote.xsl.

* **Exception Handling:**

For all the scopes in CCRMDAD\_SubmitDraftQuoteMessage catch all types of exceptions and throwing the fault message.

* + *Selection Failure****:***

For any Selection Failure fault exception while transformation catches the exception and throwing remote fault exception and appending the fault message to the title.

* + *remoteFault:*

For any Remote fault exception while transformation catches the exception and throwing remote fault exception and appending the fault message to the title.

* + *runtimeFault:*

For any run time fault exception while transformation catch the exception and throwing run time fault exception and appending the fault message to the title.

* + *bindFault:*

#### For any data binding or data validations exceptions while transformation catch the exception and throwing remote fault exception and appending the fault message to the title.

#### NOTE: After Fault the BPEL instance is ready for Re-initiate, Manually user/Ops team will be re-initiating after all validations done

* **Input/output Schemas:**
* EAI.MSG.CCRM.DAD.309\_DraftQuotePublishRequest.xsd
* EAI.MSG.CCRM.DAD.308\_DraftQuotePublishRequest.xsd
* **Partner links**
* [CCRMDAD\_SubmitDraftQuoteMessage.wsdl](http://eaiblue.ime.reuters.com:7777/orabpel/ccrmasync/CCRMDAD_SubmitDraftQuoteMessage/R5.0_15.1/CCRMDAD_SubmitDraftQuoteMessage.wsdl)
* [Enqueue\_DrafttQuoteService.wsdl](http://eaiblue.ime.reuters.com:7777/orabpel/ccrmasync/CCRMDAD_SubmitDraftQuoteMessage/R5.0_15.1/Enqueue_DrafttQuoteService.wsdl)
  + 1. PL-SQL

**-**

# Developer Notes

##### **Code management**

For Code management and usage of subversion refer the document.

##### **Deployment Process:**

The deployment process for BPEL, ESB or web services components and as well as for any other deployments like DB Scripts, Business rules configurations: refer EAI- Build and Deployment Process.doc in SVN folder:

<https://sami-crm6-eai.reutersdev.net/svn/sami-crm6-eai/trunk/dev/EAI-CRM61/R2releases/BuildDocs>

# Issues/Workarounds

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.No | Summary | Status | Assigned to | Solution |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |