

Version

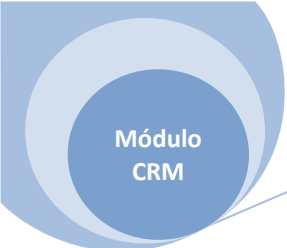
This article was created by SAP C4C (1608) Version

Brief Description

The objective of this article is describe in details how define a delta Master Data between SAP on Premise (ECC) and SAP C4C. On this Article I will describe the delta load of Customers, replicating data customer from ECC to C4C for every bidirectional changes.

Author

Cláudio Goulart is CRM Consultant since 1999, expert on plenty of CRM Suites as Vantive, PeopleSoft, Siebel, SAP CRM and SAP C4C.



Article - Blog Módulo CRM

Content

- 1. Customization 3
 - 1.1. Description 3
 - 1.2. Activating Change Pointers on SAP On Premise (ECC) 3
 - 1.3. Defining Display Distribution Model..... 3
 - 1.4. Setting Message Types Inbound & Outbound properties 4
 - 1.5. Setting Additional Data for Message Type 5
 - 1.6. Executing JOBS do Replicating Data 5

1. Customization

1.1. Description

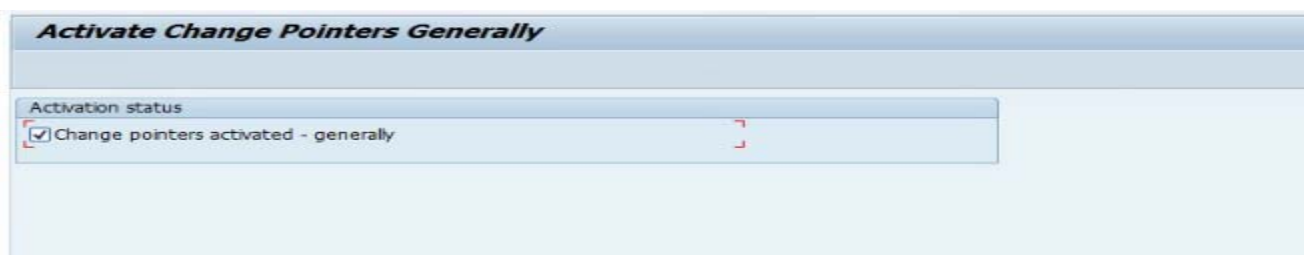
As I mentioned in article "Initial LOAD to SAP On Premise (ECC)", I detailed steps to Initial Load Data Master to SAP C4C. On this article, I will describe how to define the Delta Load between SAP ECC and SAP C4C.

Initial LOAD to SAP On Premise (ECC)

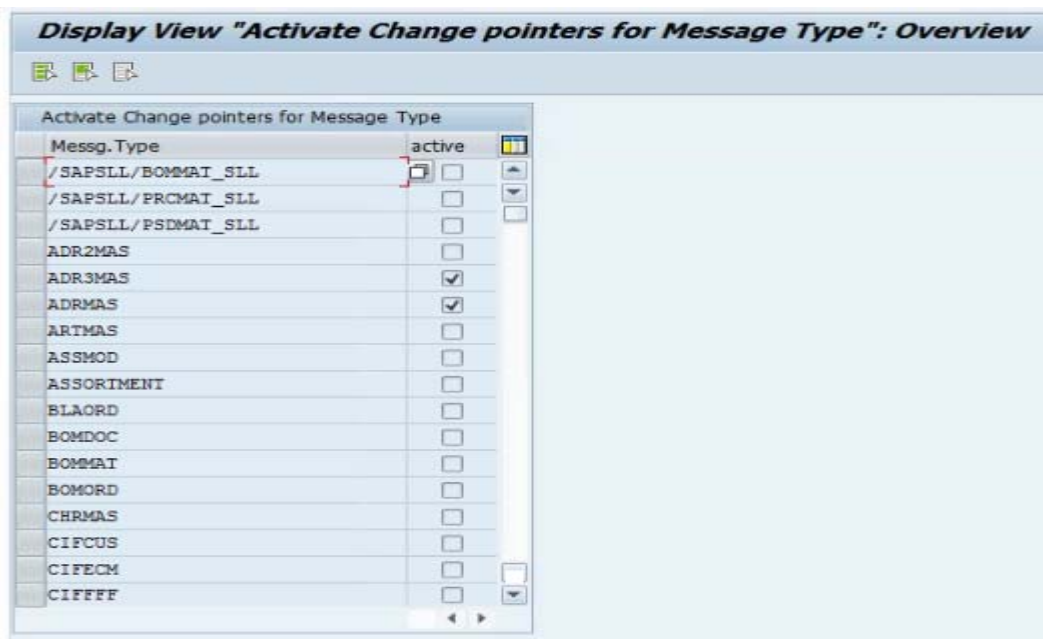
Article : <http://www.modulocrm.com.br/?p=92>

1.2. Activating Change Pointers on SAP On Premise (ECC)

IMG : SAP Customizing Implementation Guide -> SAP NetWeaver -> Application Server -> IDoc Interface / Application Link Enabling (ALE) -> Modelling and Implementing Business Processes -> Master Data Distribution -> Replication of Modified Data -> Activate Change Pointers - Generally



IMG : SAP Customizing Implementation Guide -> SAP NetWeaver -> Application Server -> IDoc Interface / Application Link Enabling (ALE) -> Modelling and Implementing Business Processes -> Master Data Distribution -> Replication of Modified Data -> Activate Change Pointers for Message Types



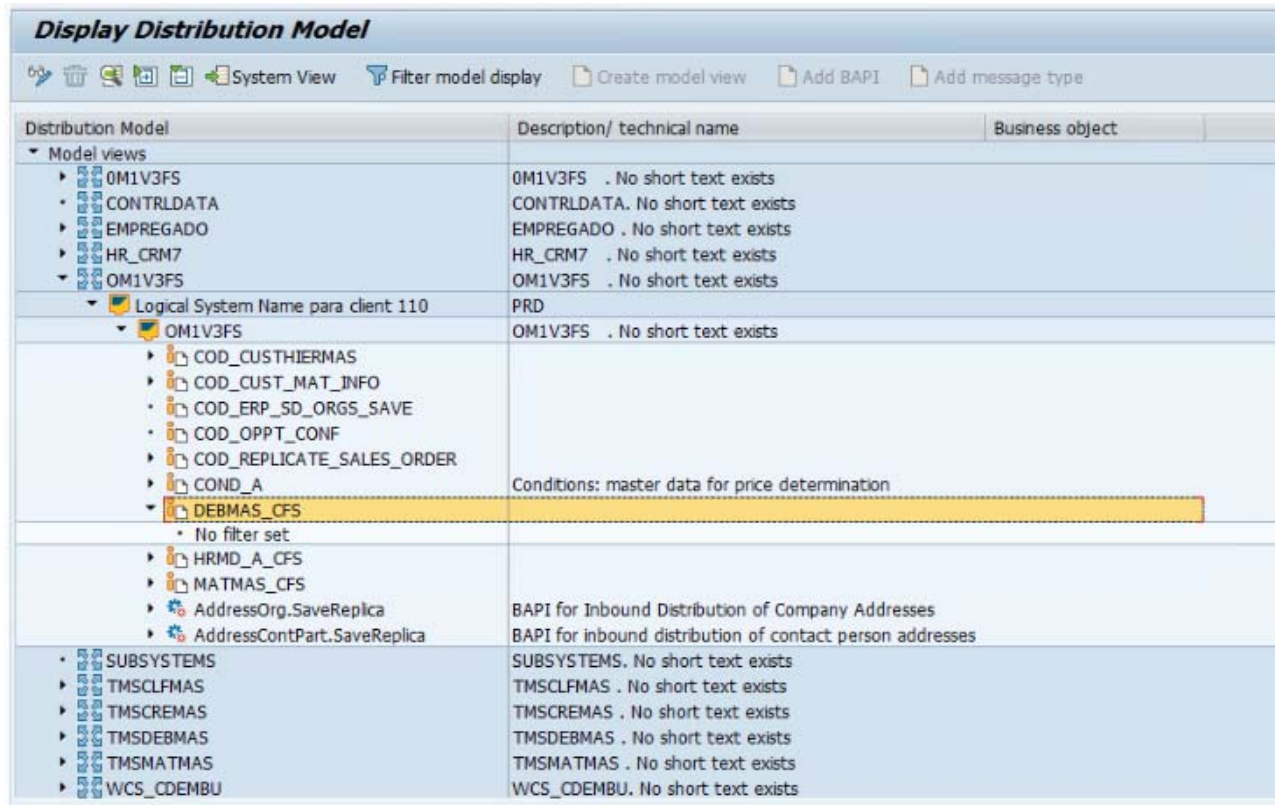
- DEBMAS_CFS (Customer Header)
- ADR3MAS (Address)
- ADRMAS (Contacts)

1.3. Defining Display Distribution Model

On Transaction BD64 is defining the distribution model, which define what record will be distributed to C4C.

Transaction : BD64

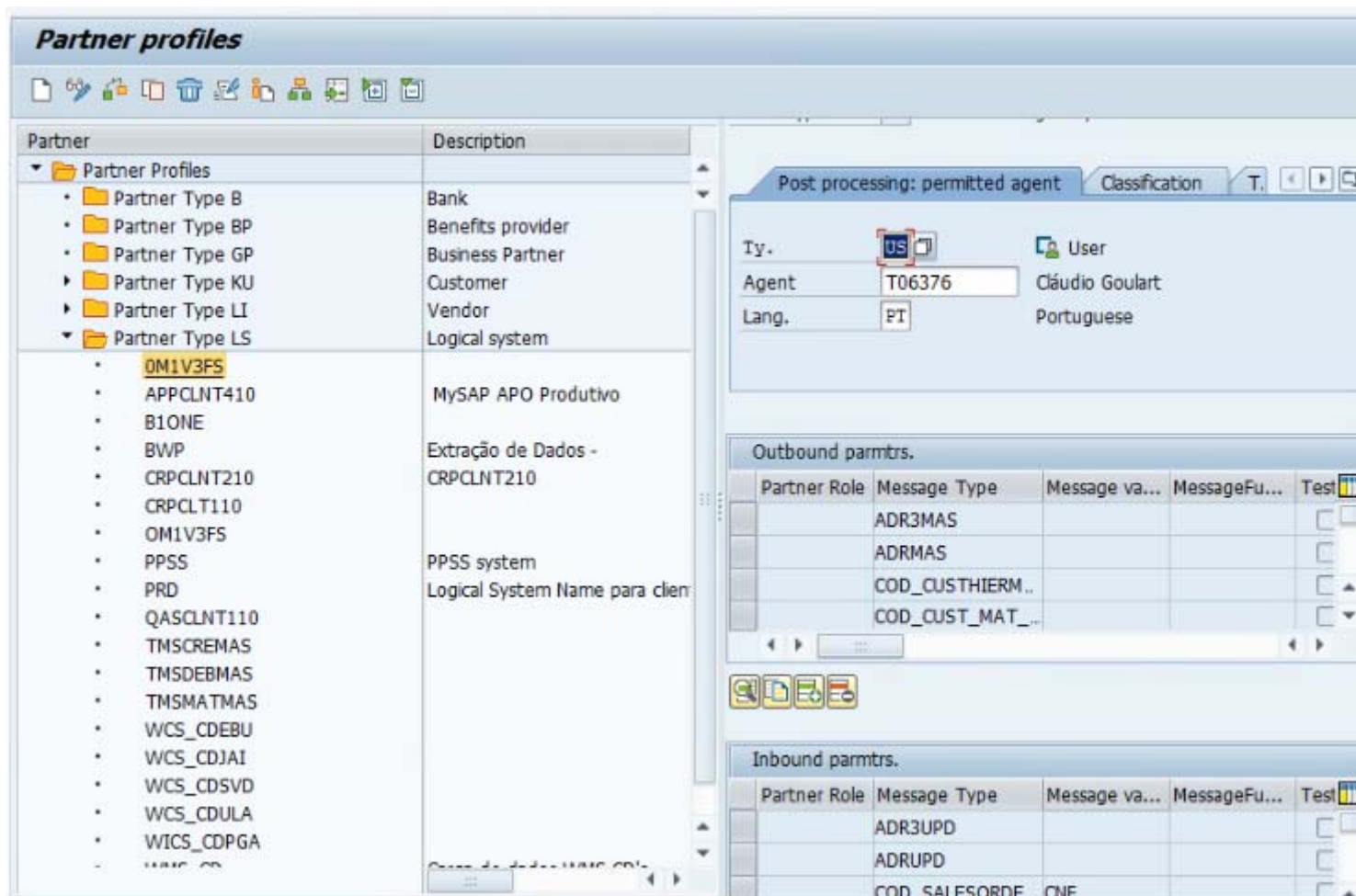
Selecting your message type and selecting new filter. That filter could be by Organization, Office and much more.

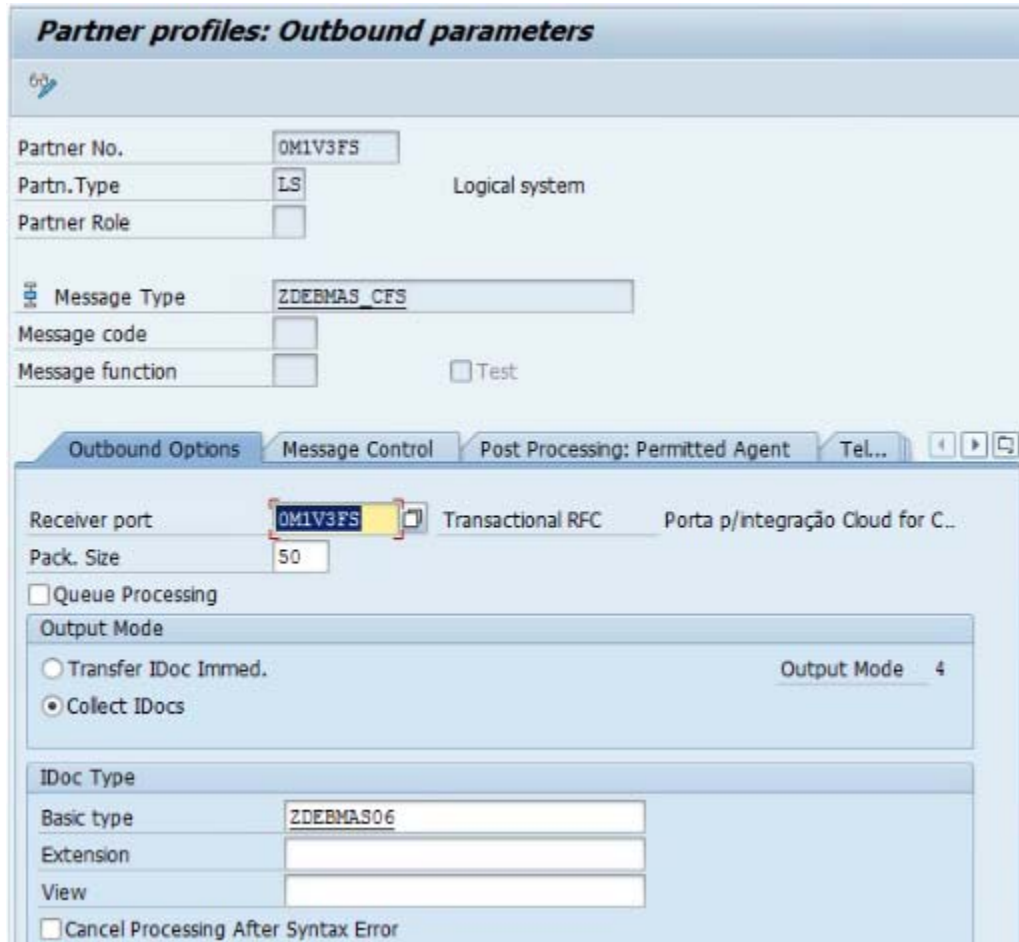


1.4. Setting Message Types Inbound & Outbound properties

This configuration define which message type Outbound and Inbound can be distributed.

Transaction : WE20





1.5. Setting Additional Data for Message Type

On view below is responsible to specified a mapping of message type and a function module to additional data manipulation. As you can see I created an Z function module to accomplish my objectives.

Transaction : SM30

View : V_TBDME

Messg.Type	Ref.msg.	Funct.Mod.	Table
DEBMAS_CFS	DEBMAS	ZCOD_CUST_GROSS_REPL_WITH_CP	KNA1
DEBMAS_SOD	DEBMAS	MASTERIDOC_CREATE_SMD_DEBMAS	KNA1
DEPNET	DEPNET	MASTERIDOC_CREATE_SMD_DEPNET	
DOCMAS	DOCMAS	MASTERIDOC_CREATE_SMD_DOCMAS	DRAW
DOLMAS	DOLMAS	MASTERIDOC_CREATE_SMD_DOLMAS	
ECMMAS	ECMMAS	MASTER_IDOC_CREATE_SMD_ECMMAS	
ECMREV	ECMREV	MASTER_IDOC_CREATE_SMD_ECMREV	
FRE_ART_HIER	FRE_ART_HIER		
FRE_ART_SITE	FRE_ART_SITE		
FRE_DIF_NO_SITES	FRE_DIF_NO_SITES		
FRE_LAYMOD			
FRE_LOC_ADDRESS	FRE_LOC_ADDRESS		
FRE_LOC_SITE	FRE_LOC_SITE		
FRE_LOC_VENDOR	FRE_LOC_VENDOR		
FRE_PLIFZ	FRE_PLIFZ		
FRE_REF_SITE	FRE_REF_SITE		
FRE_REL_PRO	FRE_REL_PRO		

1.6. Executing JOBS do Replicating Data

It's necessary to schedule a JOB on SAP On Premise to replicate data between SAP ECC and C4C. Below I defined a JOB strategy to data replicate. Of course, it can change which business process needs.

JOB 1

JOB	Sequence	Report	Message Type	Interval
Customer Transfer (ECC -> C4C)	1	RBDMIDOC	DEBMAS_CFS	5 minutes
	2	RBDMIDOC	ADR3MAS	Dependent step before
	3	RBDMIDOC	ADRMAS	Dependent step before
	4	RSEOUT00	DEBMAS_CFS	Dependent step before
	5	RSEOUT00	ADR3MAS	Dependent step before
	6	RSEOUT00	ADRMAS	Dependent step before

JOB 2

JOB	Sequence	Report	Variant	Interval
Customer Transfer (C4C -> ECC)	1	RBDAPP01	DEBMAS	5 minutes
	2	RBDAPP01	ADR3UPD	Dependent step before
	3	RBDAPP01	ADRUPD	Dependent step before