

S/4 HANA migration cockpit comes with preconfigured content and mapping for each migration object. For all major functional areas like product, customer, bank, profit centre, cost centre, plant, GL, etc. migration objects are predefined and their sequence and dependency is also defined. So, what is the process of migrating custom data into SAP S/4HANA? This blog addresses the issue of migrating data in custom tables. A new migration object has to be created or changes to an already existing migration object need to be made. Migration object modeler is the application that is used when custom data has to be migrated into SAP S/4 HANA system.

Note: SAP S/4HANA Migration Object Modeler is only relevant for the on-premise edition of SAP S/4HANA.

Transaction LTMOM can be used to access SAP S/4HANA Migration Object Modeler (MOM).

There are 2 ways of migrating custom data using MOM:

- 1. Creating a new migration object
- 2. Changing already existing migration object
- This blog covers Option 1 only.
- Steps of creating new migration object

1. Create Project: Creating a project in migration cockpit is 1st step for migrating data in S/4HANA system. Run transaction LTMC and create one project.

2. Create a new migration object: Run transaction LTMOM for creating new migration object. Create a new migration object in LTMOM.

3. Create a new function module: Go to transaction SE80 and create one custom function module. Define import and export parameters for FM. This FM would decide template for the new migration object.

4. Define source structure: In this step, define the new structure and its fields as per custom data requirement. This structure would work as source structure in the migration process.

5. Define Target structure: After defining source structure define the new target structure and its fields.

6. Structure mapping: In this step, map source structure to the new custom function module and then map source structure to target structure.

7. Field mapping: After mapping between source and target structure, define mapping between every field of source structure to a field of the target structure.

8. Define Rules: In this step, define rules for every field like constant value, condition on which value should be filled, etc.

9. Generate Runtime object: After completing all the above steps, generate runtime object of new migration object.

After completing the above steps, a new migration object can be seen in the LTMC transaction under the project that was created in Step 1.

## Example: Migrate data from a custom table to S/4 HANA system

The custom table 'ZCUSTINS' below that stores customer number and their policy id needs to be migrated to SAP S/4 HANA. To do this, a new migration object is created in MOM.

Following is the structure of a custom table.

ransparent Table	2CUS:	TINS	D Active	r.			
nort Description	test o	uston	ner policy				
Attributes D	elivery a	nd Ma	intenance Fields	Input He	lp/Check	Curr	rency/Quantity Field
					_		
		-		Srch Help	Buil	t-In Typ	e
Field	Key	<b>V</b> Ini	Data element	Srch Help Data Type	Length	t-In Type Deci	e Short Description
Field	Key	Ini	Data element	Srch Help Data Type CLNT	Length 3	t-In Type Deci	Short Description
Field <u>MANDT</u> <u>CUSTOMER</u>	Key V	Ini	Data element MANDT KUNNR	Srch Help Data Type CLNT CHAR	Built Length 3 10	t-In Type Deci C	Short Description Client Customer Number

## 1.Create one project 'ZNS4\_DATA\_MOM' in LTMC.

~	✓ Migration Project Details						
		Name: zns4_Data_mom			Status:		Starter
	Data	Source: File		Ma	ass Transfer ID:	007	
	Defa	ult View: On-premise - Ente	erprise Management scope	$\sim$			
	Migration	Objects Notifications	Settings				
Filte	er: All	✓ Open Activate Dea					
Ē	Status	Status Text	Name		Documentatio	on	Dependent N
	$\diamond$	Inactive	Accounts payable (Vendor) open item		Show		Cost center
	\$	Inactive	Accounts receivable (Customer) open item		Show		Customer

2. Create one function module 'Z\_FM\_CUST\_INS' with the following import and export parameters:

Function module	Z_FM_C	UST_INS		Active				
Attributes Import	Ð	kport Changi	ng Tables	Exce	ptions	5	Source code	
Parameter Name	Турі	Associated Type	Default val	ue	Op	Pa	Short text	Lo
IV_DOCNO	TYPE	SWO_COMMIT				$\checkmark$	Call to COMMIT WORK	
Function module	Z_F	CUST_INS		Ac	tive			
Attributes Imp	ort 🍸	Export Cha	anging Ta	bles	Exce	eption	ns Source code	
Parameter Name	Тур	ing Associ	ated Type	0	ption	al S	Short text	
IT_CUSTINS	LIK	E ZCUST	INS			t	est customer policy	
IT_RETURN	LIK	E BAPIR	ET2			R	leturn Parameter	

IV\_DOCNO is a parameter which will ensure that FM would call when simulation will start in LTMC transaction.

IT\_CUSTINS is a table in which data will come into the function module. IT\_RETURN table would be used to pass error, success or warning messages.

3. Run transaction LTMOM and select our project ZNS4\_DATA\_MOM

Migration Object Modeler	
<b>i</b>	
Object Browser	
Project 🗸	
zns4_Data_mom  × × × 632	
,	

4. Create a new migration object. Use the below path:

Go to menu ->migration object->create migration object->user defined.

The following pop up window will come. Give a name and description of a new migration object:

Project	📴 User-Defined	×
zns4_Data_mom	Select a project a specify object	
	Project Name	zns4_Data_mom
<u>8</u> 20		
Name	Migration Object ID	z_cust_ins _007
ZSIN_MIG_00	Description	Customer Insurance
<ul> <li>Subprojec</li> <li>ZSIN I</li> </ul>		
T 🗁 Mig	Define Source Structure from	Fie
• 🗖	Define Target Structure from	Function Module
• 🖻		
1		
• Foo		
• 🗖 Tra		
🕨 🖿 Va		
		Steven Next 🗶

5. After clicking on the next button it will ask for function module information:

🖻 User-Defined	
Project Name	zns4_Data_mom
Define Target Structure	
Name of Function Module	z_Fm_cust_ins
Options for Processing Business Object Instances	
Process One Instance at a Time	۲
Process Multiple Instances at Once	0
Return Parameter Settings	
Parameter that Returns Result	it_return
Include for Handling the Result Parameter	
Simulation Settings	
Parameter for Simulation	iv_docno
Value for Simulation	
Value for Writing Data	x
	😽 Back 🗹 🗶

Give FM name, success and error message would be passed

IT\_RETURN table, IV\_DOCNO is used as a simulation parameter.

6. New migration object Z\_CUST\_INS\_007 has created successfully.

Project 💙				
zns4_Data_mom	× 🔻 65°			
<u>sa</u>				
Name	Description			
ZSIN_MIG_007	zns4_Data_mom			
<ul> <li>Bubprojects</li> </ul>				
ZSIN_MIG_007	zns4_Data_mom			
<ul> <li>Migration Objects</li> </ul>				
Z_CUSTOMER_007	Customer			
Z_Z_CUST_INS_007	Customer Insurance			
<ul> <li>Global Data</li> </ul>				
<ul> <li>Source Structures</li> </ul>				
<ul> <li>Target Structures</li> </ul>				
<ul> <li>Structure Mapping</li> </ul>				
<ul> <li>Field Mapping</li> </ul>				
<ul> <li>Exed Values</li> </ul>				
<ul> <li>Rules</li> </ul>				
<ul> <li>Translation Objects</li> </ul>				
<ul> <li>Variables</li> </ul>				

7. Double click on source structure, it will open a new window on left. Define new structure and its fields as follows.

			Field	s of custom	er insurance					
Project 🕑		<ul> <li>Source Structures</li> </ul>	So	urce Structu	re ZCUSTIN	IS [				
zns4_Data_mom	× 🔻 65°	<ul> <li>Customer insurance</li> </ul>	De	scription	custome	r insurance				
Name	Description			(PIC)	1 El					
ZSIN_MIG_007	zns4_Data_mom		-							
<ul> <li>Emprojects</li> </ul>				Key field	Name	Data Type	Length	Decimal Places	Amount Field	Column Header
ZSIN_MIG_007	zns4_Data_mom			•	CUSTOMER	CHAR	10			Customer
Migration Objects					POLICYID	CHAR	10			policyid
Z_CUSTOMER_007	Customer									
Z_Z_CUST_INS_007	Customer Insurance									
<ul> <li>Global Data</li> </ul>										
<ul> <li>Source Structures</li> </ul>										
<ul> <li>Target Structures</li> </ul>										
<ul> <li>Structure Mapping</li> </ul>										

8. Above structure would create a template for migration cockpit. Some fields can be made mandatory or disabled by making some changes in customer view of this structure. Right click on structure name and select 'display view'. Below view is Customer view.

Object Browser		Migration Object: Customer Insurance					Generated
Broject		ت ا الله الله الله الله الله	Vie	ws on cust	omer insurance		
znst Data mom	X 👻 Co	Source Structures					
[enor_bace_more		• ••• cuscomer insurance	E	Туре	Name	Customer	View
				Structure	customer insurance	Requireu	
				Field	Customer	Visible	<b>a</b>
Name	Description			Field	policyid	Visible	<b>a</b>
ZSIN_MIG_007	zns4_Data_mom						
<ul> <li>Subprojects</li> </ul>							
ZSIN_MIG_007	zns4_Data_mom						
Migration Objects							

All the fields for this structure need to be filled in the source template.

9. Double click on target structure. Either we can define fields for target structure one by one or we can get all fields of target structure from FM 'Z\_FM\_CUST\_INS' (which we defined earlier for custom migration object) by synchronization. For synchronization use below the path

Go to -> menu->migration object->synchronize structure.

10. Now go to structure mapping. Assign source structure to function module first and then assign source structure to target structure. Mapping can be done by dragging from left to right.

S< M <	
<ul> <li>Target Structures</li> </ul>	<ul> <li>Bource Structures</li> </ul>
• 🔄 Start of Processing ( )	🔻 💐 customer insurance
• 🛃 Start of Block ( )	
▼ 🖉 Z_FM_CUST_INS	
<ul> <li>Call to COMMIT WORK</li> </ul>	
test customer policy < <customer insurance<="" p=""></customer>	
<ul> <li>Return Parameter</li> </ul>	
• 🦉 End of Block ( )	
• 🦉 End of Processing ( )	

11. Now go to field mapping and assign fields of source structure to target structure by dragging and dropping from left to right.

Target Structures		<ul> <li>E Source Structures</li> </ul>
Z_FM_CUST_INS	۳	<ul> <li>customer insurance</li> </ul>
Call to COMMIT WORK		• 🔗 Customer
<ul> <li>A Call to COMMIT WORK</li> </ul>		• 🖉 policyid
test customer policy < <customer insurance<="" p=""></customer>		
<ul> <li>Description</li> <li>End Start of Loop(customer insurance) ( )</li> <li>End Start of Record(test customer policy) ( )</li> </ul>		
• 🧬 Client		
• 🔗 Customer Number MOVE (customer insurance-Customer	2	
<ul> <li>Policy id MOVE (customer insurance-policyid)</li> </ul>		
<ul> <li>End Of Record(test customer policy) ( )</li> </ul>		
<ul> <li>After Loop(customer insurance) ( )</li> </ul>		
<ul> <li>Return Parameter</li> </ul>	12	
<ul> <li>Message type: S Success, E Error, W Warning, I Info, A Ab</li> <li>Message Class</li> </ul>	¢	

Assign customer from source structure to customer number and policy id to policy ID.

12. Create a run time object by clicking on the following highlighted button:

Change Migration Object Customer Insurance				
🂖 🕙 🖆 💿 🗆 🖬				
Object Browser		Migration Object: Customer Insu		
Design at the local		📚 🚖   🛗 👘   🍞 , 🎽 Hid		
Project		<ul> <li>Target Structures</li> </ul>		
zns4_Data_mom	× 🔻 68	Z_FM_CUST_INS		
		<ul> <li>Call to COMMIT WOR</li> </ul>		
		<ul> <li>A Call to COMMIT W</li> </ul>		
Namo	Description	<ul> <li>test customer policy</li> </ul>		
	Description	<ul> <li>E</li> <li>Start of Loop(cus</li> </ul>		
* 📂 ZSIN_MIG_00/	zns4_Data_mom	<ul> <li>Estate of Pacard(t)</li> </ul>		

13. Run transaction LTMC and select project ZNS4\_DATA\_MOM. You can see the new custom migration project below.

Name:	zns4_Data_mom	s	tatus:				
Data Source:	File	Mass Transf	fer ID:				
Default View:	On-premise - Enterprise I	Nanagement scope	~				
ition Objects Notifications Settings							
✓ Open Activate Deactivate							
Status T	iext	Name	Docur	mentat			
Inactive		Characteristic	🗐 SI	how			
Inactive		Class	📑 Si	how			
Inactive		Cost center	📑 SI	how			
Started		Customer	🗐 SI	how			
Inactive		Customer - extend existing record by new org levels	🗐 SI	how			
Started		Customer Insurance					
Inactive		Equipment	📄 Si	how			

Above is the default view 'On-premise- Enterprise Management scope'. The newly created custom migration object needs to be edited. Click on the edit button and select the custom view option.

~	Migration Project De		
	Name:	zns4_Data_mom	
ĺ	Data Source:	File	٦
	Default View:	Customer View 🗸	
		0	Į

14. Now select custom migration object 'customer insurance' and download its template for filling customer insurance data into the input file.

The Downloaded template has below 2 fields (Which were defined in source structure of custom migration object)

- Customer
- Policy

Fill customer number and policy id into the downloaded file. Below is a screenshot of the downloaded file after filling data into it. 

CUSTOMER	POLICY
000000014	000001001

15. Upload file and start transfer.

16. Go to table ZCUSTINS and check entry created.

	Data Browser: Table ZCUSTINS Select Entries 2						
🗅 🖉 🎸 🥞 🚢 🐺 🗊 🔂 🗓							
Table: ZCUSTINS Displayed Fields: 3 of 3 Fixed Columns: Client Customer policy id 100 000000014 0000001001					[2]	List Width 0250	
	100	000001001	0000000011				

We have uploaded custom data into SAP S/4HANA system successfully with the help of MOM.

Migration object modeler is an application to migrate custom data into s/4HANA system by creating or changing migration object as per custom requirement. It is an assistant application for LTMC. Data uploading can only be done via migration cockpit.

In the next blog, we will discuss how to change existing migration object with the help of MOM.

For all enquiries please contact at : corp@acnsol.com , Tel : +1(877)-849-5838 Visit us at : www.acnsol.com

USA Head Office 3350 Scott Blvd, Bldg 34 Santa Clara, CA 95054

South Africa 609 Lanseria Corporate Estate, Galvarino Gallardo 1638, Falcon Lane, Lanseria, Gauteng

Chile Providencia, Santiago

India **Development Centre** 102A, HARTRON, Electronics City, Gurgaon



Copyright © Accrete Business Solutions 2019. All rights reserved.