



Level: Intermediate

Works with: Lotus Enterprise Integrator

Updated: 02-Dec-2002

Creating SAP requisitions from a Domino application

by
Scott
Morris

SAP R/3, also known as mySAP.com, is the world's leading Enterprise Resource Planning (ERP) software. It is both powerful and complex, providing structure and rules that help companies deal with their business tasks. These tasks can range from general areas such as human resources, inventory control, and finance, to more specialized services for industries like chemical manufacturing or beverage distribution.

Due to its complexity, you can't simply install SAP "out of the box" and expect users to figure out how it works. You must first customize SAP to meet your needs, then your users must be carefully trained to use it. This training can add significant cost to your overall SAP deployment budget.

Domino can help you reduce your training (as well as licensing) costs for your SAP software by providing front-end SAP functionality through Domino applications. This combination of SAP working together with Domino also gives you features beyond what either product can provide on its own.

In this article, we build a simple Domino application as a front-end for the Purchase Requisition System of SAP. Our application will feel very familiar to Notes and Web users, avoiding the need for SAP training. Perhaps equally important, our application will significantly reduce the number of SAP user licenses needed at your site, since users will be using Domino (instead of SAP itself) to access this system.

This article is intended for Domino application developers familiar with Lotus Enterprise Integrator (LEI). While experience with SAP is not required for this article, it may help you better understand some of the SAP concepts we mention.

How the Domino application works with SAP

The integration application works as follows:

1. You download the SAP material inventory into a Domino database so that it can be used by the forms of the Domino application.
2. In this application, users can order something from the inventory that is available in the Domino database.
3. Users then save the Domino document containing all the information needed to create a Purchase Requisition in SAP.
4. You send this requisition to SAP, where SAP's application logic either accepts or rejects it.
5. If rejected, the user receives email with SAP's message about what is wrong.
6. If accepted, the user receives an email message with the Purchase Requisition Number from SAP, which can be used for additional tracking, if needed.

To read data from or to write data to SAP from Lotus Domino, we use the Lotus Enterprise Integrator (LEI) together with the Lotus Connector for SAP R/3. LEI is a great tool that moves data from a source to a target or synchronizes data between two sides without writing any code. You just fill out the forms of the LEI Administration database that defines how to access our data sources or targets, and then you define activities to control the data

flow. The Lotus Connector for SAP, which we refer to as SAP Connector, reads and writes data to SAP's Remote Function Calls (RFCs) and Business Application Programmers Interface (BAPIs), which are really just RFCs. The SAP Connector can also call transactions via batch input mode.

Think of the RFCs and BAPIs as similar to standard C, C++, or even LotusScript functions. They have input parameters and output parameters and some parameters that are both input and output. Some of these parameters are a special type known as tables, which are groups of parameters. Transactions are recordings of forms, one after the other, as they are submitted. By filling out the fields in each of the forms and sending them in a stream to SAP, you can replay the transaction for SAP.

Typically, the out-of-the-box Domino application developer knows very little about SAP, particularly RFCs and BAPIs, so the developer needs to work closely with the SAP application developers—the SAP ABAP developers—who know the details of how SAP runs inside. The SAP ABAP developers can tell you which RFCs or BAPIs provide the services that you are looking for and how to access those services.

Building the Domino application that connects to SAP

Let's start building the application. First, you should download SAP material inventory information into a Domino database. Why not just access the data from Domino live? The answer is that you can save on SAP user licenses by making a copy of the data. It takes one SAP license to download the data into Domino. Now it becomes available to any Notes/Domino user without additional cost. There are other reasons for downloading the data. What if SAP is only available at the Chicago location, but you have Domino users all over the country? Even if you have a connected corporate network, bandwidth issues make the prospect of working from local Domino servers much more efficient. Or, if you rely on Domino to periodically connect users over telephone lines, working locally may be the only option. The application would in this case extend the services of SAP to those who can't access them directly, helping you to get more out of your SAP investment and preventing the sin of all sins—a paper-based process.

How do you download information from SAP into Domino? You ask your SAP ABAP developers, and they tell you about two RFCs available to help with your task:

- `MM_SELECT_MATERIAL_GROUPS` provides the complete list of categories for items in your SAP system
- `MM_MATERIALS_FOR_GROUP` provides all of the materials or items available for a given group

There are two ways to attack this problem. You could write a script using the Lotus Connectors Lotus Script Extension (LCLSX) that, using the SAP Connector, calls `MM_SELECT_MATERIAL_GROUPS` and for each entry found, makes a second call to `MM_MATERIALS_FOR_GROUP`. If you had 10,000 material groups, you might consider this path of action. But for this example, you only have a handful of material groups, so you can use LEI to periodically call `MM_MATERIALS_FOR_GROUP` for each of the groups that you want. These groups are maintained in the SAP system. By doing this, you avoid having to write a lot of LotusScript. You end up having a little script to help properly fill the forms and to recalculate them at the correct time.

How do you do this? You define the connection to your SAP application server, and then call `MM_MATERIALS_FOR_GROUP`.

Connecting to the SAP application server

To retrieve information from the SAP application server, you need two connection documents. The first connection document is an SAP Connection document that connects to the SAP application server. The second document tells the server where to send the data. You create the first connection document in the Lotus Connector for SAP R/3 database. Here is what the connection document looks like in our example:

SAP Connection	
Connection Properties	
Name:	MM_MATERIALS_FOR_GROUP
Hostname:	eitech
Destination:	E11
System Number:	0
Client:	800
User Name:	muster
Password:	XXXXXXXXXX
Language:	E
Select RFC, BAPI or Transaction:	<input checked="" type="radio"/> RFC <input type="radio"/> BAPI <input type="radio"/> Transaction
RFC or BAPI:	MM_MATERIALS_FOR_GROUP
Debug Level:	0
RFC User:	0
Password NOT Encrypted	

Eitech is the hostname of the SAP R/3 4.6C system running on IBM's Universal Database (UDB) 7.1. The other entries in this document include standard SAP logon information provided by the SAP developers. Under RFC or BAPI, you tell this connection to the SAP application server to call MM_MATERIALS_FOR_GROUP. Now you have more than one group to get the materials for, so you use this connection document in more than one data movement activity.

Next, you create a Notes Connection document in the LEI Administrator's database to define where to send the SAP data. Here is a sample Notes Connection document:

Notes Connection	
Connection Properties	
Name:	Buy It Database
Domino Server:	eitech
Notes Database:	materials.nsf

The example Domino server is also on eitech. It is not common to have Domino and SAP running on a single host, but it is possible (and it lets us perform live demos wherever we are!). The Notes database where you send the SAP data is titled Buy It Database with the file name materials.nsf. You can create this database before you set up the connection documents, or you can configure the Notes Connector to create it for you. The database created by the Notes Connector won't be pretty, but any help that moves you along faster is welcome. It is more efficient to let the Notes Connector create the database now and to clean up the user interface later so that you don't have to worry about the data types and field mapping, which LEI and the connectors can take care of for you.

Tip If you want to call an RFC or BAPI in SAP to return many fields of data, first create a Direct Transfer Activity document in the LEI Administrator. Choose the Create Target Metadata option and let this activity create the Notes form for you with all of the field names defined with the correct data types. Do this even if you want the services of a Replication Activity. Run the Direct Transfer Activity once and let it jumpstart the rapid application development process.

The MM_MATERIALS_FOR_GROUP only returns three fields, so you may want to create this form manually, but just to show you, here is the Direct Transfer Activity document sample:

Direct Transfer Activity		Author: S
Activity Name:	MM_MATERIALS_FOR_GROUP	
Current Status:	Disabled	
Last Completed Run:	Time: 01/29/2002 05:20:36 PM Result: Activity completed successfully	
Source		
Connection Name:	MM_MATERIALS_FOR_GROUP (SAP)	
Table Name:	T_MATERIALS	
Select Statement:	I_MATERIAL_GROUP="002" I_LANGUAGE="E" I_PLANT="1000"	
Target		
Connection Name:	Buy It Database (Notes)	
Form Name:	Material	
Field Mapping <input type="checkbox"/> Automatic		
Source Field List:	Target Field List:	
T_MATERIALSMATX T_MATERIALSMATNR T_MATERIALSMEINS	MAKTX MATNR MEINS	

Under Direct Transfer Options (which is not shown in the previous screen), select the Create Target Metadata option to create the form Material in the Buy It Database with the fields as described in the field mapping section.

You can also select automatic field mapping option to name the fields as they are named in the source or SAP connector side of the field mapping.

In the previous screen, note the Select Statement in the Direct Transfer Activity document. (The parameters of this Select Statement are listed in the following table.) You must tell the RFC MM_MATERIAL_FOR_GROUPS what you want from it. The Select Statement field shows the IMPORT parameters of this call. This call is not an SQL or a Notes selection formula. It is specific to the SAP connector. You have three IMPORT parameters, which are described in the following table:

Parameter	Description
I_MATERIAL_GROUP="002"	Indicates which group you want the materials for. You can get only one group at a time from an activity.
I_LANGUAGE="E"	Indicates which language you want the results in.
I_PLANT="1000"	Indicates which plant you want the information for. In the example, there is only one plant where everything is located.

In the interface of MM_MATERIALS_FOR_GROUP in SAP's ABAP workbench, you can see how these values fit into the call. These three IMPORTS go to the IMPORTING section, and you get back a table, T_MATERIALS, containing the data that you are looking for. The following SAP code tells you which tables the data comes from:

```

FUNCTION MM_MATERIALS_FOR_GROUP.
**-----
** Local Interface:
** IMPORTING
**   VALUE(I_MATERIAL_GROUP) LIKE T023-MATKL
**   VALUE(I_LANGUAGE) LIKE SY-LANGU DEFAULT SY-LANGU
**   VALUE(I_PLANT) LIKE T001W-WERKS
** TABLES
**   T_MATERIALS STRUCTURE MATM_WWW
**-----
CLEAR T_MATERIALS.
REFRESH T_MATERIALS.
IF I_PLANT IS INITIAL.
  GET PARAMETER ID 'WRK' FIELD H_WERKS.
ELSE.
  H_WERKS = I_PLANT.
ENDIF.

* read all stock of material to the categories of commodities
SELECT * FROM MARA WHERE MATKL EQ I_MATERIAL_GROUP.

* also read short text of the materials
SELECT SINGLE * FROM MAKT WHERE SPRAS EQ I_LANGUAGE

```

AND MATNR EQ MARA-MATNR.

```
* must be maintained for Reservations and/or Banf
SELECT SINGLE * FROM MARC WHERE MATNR EQ MARA-MATNR
AND WERKS EQ H_WERKS.
CHECK SY-SUBRC EQ 0.
T_MATERIALS-MATNR = MARA-MATNR.
T_MATERIALS-MAKTX = MAKT-MAKTX.
T_MATERIALS-MEINS = MARA-MEINS.
APPEND T_MATERIALS.
ENDSELECT.
SORT T_MATERIALS BY MAKTX.

ENDFUNCTION.
```

If you know how to read SAP ABAP code, you know that the data comes from table MARA that contains all of the material inventory information and table MAKT that contains language-specific descriptive text.

Replication Activity

But what you really want is to have the downloaded SAP material information regularly synchronized. The best way to do this is with a Replication Activity document in the LEI Administration database. You can set up a one way replication that pulls from MM_MATERIALS_FOR_GROUP into the Buy It Database.

This next part may be a bit tricky to understand. You want to get the materials for groups 001, 002, and 003, so you need three replication activities. The first activity calls MM_MATERIALS_FOR_GROUP for the I_MATERIAL_GROUP="001." The second Replication Activity makes a call for I_MATERIAL_GROUP="002," and the third makes a call for I_MATERIAL_GROUP="003."

After replication, the target needs to be a copy of the source. How can you do this with three sources all writing to the same Notes form? There are about ten answers to this question, but the simple answer is that there is not one Notes Connection document, as pictured earlier, but three of them. In these connection documents, you use the Data Transformation feature. As mentioned earlier, MM_MATERIALS_FOR_GROUP returns the material information for a single group in SAP. It does not, however, return the group itself because that was an IMPORT parameter: I_MATERIAL_GROUP. In the Replication Activity document, you want to replicate against only one I_MATERIAL_GROUP's data at one time, so the group also needs to be stored in the Notes Connection document, even though it is not available to you in field mapping. The Notes Connection document's Data Transformation lets you do this.

As the following screen shows, the Notes Connection document adds a field Group set to 002 in each of the documents to be created in Notes during the run of the activity.

▼ Connection Options	
▶ General Options	
▶ Document Selection	
▶ Field Selection	
▼	
Data Transformation	
Formula to Execute During Select:	
Formula to Execute Prior to Insert:	select @all; FIELD Group:= "002";
Formula to Execute Prior to Update:	

It makes sense to change the name of the activity to reflect the specific nature of this Data Transformation. In the example, we change Buy It Database to Buy It Database - Group 002. You need one of these documents for each of the three groups mentioned before.

Now you are ready to create the Replication Activity document in the LEI Administrator database. The following screen shows the Replication Activity document for our Buy It Database.

Replication Activity		Author: Scott Morris	
Activity Name: Get Materials for Group 002		Current Status: Disabled	
Connection A		Connection B	
Connection Name: MM_MATERIALS_FOR_GROUP (SAP)	Table Name: T_MATERIALS	Connection Name: Buy It Database - Group 002 (Notes)	Form Name: Material
<input checked="" type="radio"/> Connection A is Master <input type="radio"/> Connection B is Master <input type="radio"/> Skip Conflicts (no Master)			
Connection A Restrictions: <input checked="" type="checkbox"/> Skip Insertions <input checked="" type="checkbox"/> Skip Updates <input checked="" type="checkbox"/> Skip Deletions		Connection B Restrictions: <input type="checkbox"/> Skip Insertions <input type="checkbox"/> Skip Updates <input type="checkbox"/> Skip Deletions	
Field Mapping <input type="checkbox"/> Automatic			
Key Field(s) A: T_MATERIALSMATNR		Key Field(s) B: MATNR	
Source Field List: T_MATERIALSMATNR T_MATERIALSMAKTX T_MATERIALSMEINS		Target Field List: MAKTX MATNR MEINS	

First of all, you don't have a source and a target as you do in the Direct Transfer Activity document. Instead, you have Connections A and B. A full replication synchronizes A with B and vice versa, but the nature of SAP prevents this. A full replication is made up of four events—Create, Open, Update and Delete. No one RFC in SAP performs all of these tasks, and LEI doesn't let you call four RFCs in a single Replication Activity. MM_MATERIALS_FOR_GROUP is the Open event—it only reads. This limits you to a one-way replication out of SAP. Fortunately, this is still very useful.

You can select Connection A as the SAP Connection and Connection B as the Buy It Database - Group 002. Then click the One Way replication button available when the document is in edit mode, which selects Skip Insertions, Updates, and Deletions from the SAP side of the form. This is now a one-way out of SAP into Domino Replication Activity. Another difference between the Replication and Direct Transfer Activities is that with replication, you need to select a Key Field. This is how you know that two documents are a match.

Tip Replication Activities are order dependent, which means that the data returned from SAP Connection A must appear in the same sort order that the Notes Connection B will return it in. Should you find that you have many updates or new document creations after a Replication Activity run, even when nothing in SAP changed, you should examine the services of the Order Metaconnection document in the LEI Administrator database. It can reorder a connector's result set to make things like replication work as you would expect. For more information about the Order MetaConnection document, see the [LEI Lotus Connectors and Connectivity Guide](#).

Here is what the example Order Metaconnector looks like:

Order MetaConnection		Author: Scott Morris	
MetaConnection Properties			
Name:	MM_MATERIAL_FOR_GROUP: Ordered		
Connection to Use:	MM_MATERIALS_FOR_GROUP (SAP)		
Text Trimming:	<input type="checkbox"/> Disable Trimming of Text Trailing Spaces		
Sorting			
Sort Character Set (optional):			
Sorting Fields:	T_MATERIALSMATNR <small>*If not provided in Activity</small>		
Sorting Options:	<input checked="" type="radio"/> Sort Ascending <input type="radio"/> Sort Descending <input checked="" type="radio"/> Sort Text as Binary <input type="radio"/> Sort Text with Case Sensitivity <input type="radio"/> Sort Text without Case Sensitivity		

Connection A references this connection name instead of MM_MATERIALS_FOR_GROUP. You could create another Order MetaConnection document for the Notes side, but it is more efficient and faster to create a hidden view for LEI that is already sorted in the correct order. The Notes Connector can create the view for us if we select the Allow View Creation/Overwrite option on the Notes Connection document. This view is used for replication of materials from the SAP system to the Domino application.

Connection Options		
<div> <div>General Options</div> <div> <div>Document Selection</div> <div> <input type="checkbox"/> Include View Responses <input type="checkbox"/> Copy Response Hierarchies (Notes to Notes only) <input type="checkbox"/> Include Documents of Other Forms <input type="checkbox"/> Delete Result Set <input type="checkbox"/> Case-Insensitive View Searches </div> <div> <div>View to Use for Selection:</div> <div>Agent to Execute:</div> <div>Full Text Search Query to Execute:</div> </div> <div> <div>(All Materials)</div> <div><input checked="" type="checkbox"/> Allow View Creation/Overwrite</div> </div> </div> </div>		

Now, back to the Replication Options of the Replication Activity as shown in this screen:

Replication Options	
<div> <div>Metadata Creation</div> <div><input type="checkbox"/> Create Connection B Metadata</div> </div>	
<div> <div>Logging</div> <div> <div>Error Logging:</div> <div><input checked="" type="checkbox"/> Log Key Values On Error</div> </div> </div>	
<div> <div>Conflict Handling</div> <div> <div>Data</div> <div> <div>Data</div> <div> <div>Data Comparison and Trimming:</div> <div> <input type="checkbox"/> Reduced-Precision Comparison <input type="checkbox"/> Case-Insensitive String Comparison <input type="checkbox"/> Disable Trimming of Text Trailing Spaces </div> </div> </div> </div> </div>	
<div> <div>Conditional Clause for A:</div> <div>I_MATERIAL_GROUP="002"J_LANGU AGE="E"J_PLANT="1000"</div> </div>	<div> <div>Conditional Clause for B:</div> <div>Group="002"</div> </div>

The Conditional Clauses, also known as Select Statements in Direct Transfer Activities, appear under Data. The SAP side is unaffected. On the Notes side, you want only Material documents with Group = "002" (field and value created by Data Transformation of the Notes Connection document).

This Replication Activity is now ready to roll. You can schedule it (under Scheduling) to run once a day. After it runs, and the other two activities for I_MATERIAL_GROUPS 001 and 003 also run, the downloaded SAP inventory will be in the Domino database and be kept up-to-date as well.

The Materials view

To allow people to browse all possible materials, you can create a view called Materials in the Buy It Database. (You can also use this view in the Requisition form.) The Materials view has two columns:

- Group, which references field Group in our Materials form
- Material Description, which references field MAKTX in our Materials form (The MAKTX field provides a description of a given item.)

Sort the Group column in ascending order and categorize the field values.

The Requisition form

At this point, the Buy It Database has a hidden view (Materials) created automatically by the Notes Connector. What you need now is a form for creating Purchase Requisitions and a view to use it in. You can create the form first to make the fields available when you build the view.

The SAP ABAP developers tell you that BAPI_REQUISITION_CREATE is the BAPI that lets you create new purchase requisitions in SAP. This is a very complicated BAPI to use. Only the SAP ABAP developers can help you here. They say that you need to complete a bunch of fields each time you call this BAPI. Here is where the power of the Domino application environment can make your end users' lives better and easier. How? With hidden fields. Of all of the fields that are required each time you call BAPI_REQUISITION_CREATE, only six of them change from requisition to requisition. Furthermore, two of these six can be completed with Notes formulas. So of the 22 total fields that you need to send to SAP, your end users must deal with only four of them. You can hide the rest of the fields, and you can make two of the remaining four fields very easy to use.

First, you need to create the following fields in the form. The following two tables show the 17 hidden fields. (Please note that the following default values work with SAP's IDES 4.6C release. These values are different on your production SAP system!)

Field name	Field type	Default value
COSTCENTER	Text	0000001000
G_L_ACCT	Text	0000400000
APREQ_ITEM	Number	1
PREQ_QTY	Number	1
ACCTASSCAT	Text	K
CURRENCY	Text	USD
C_AMT_BAPI	Number	75
DOC_TYPE	Text	NB
GR_IND	Text	X
IR_IND	Text	X
ITEM_CAT	Text	0
PLANT	Text	1000
PREQ_ITEM	Number	1
PUR_GROUP	Text	000
UNIT	Text	EA

The following two fields are dynamically computed at runtime by formulas:

Field name	Field type	Formula
CREATED_BY	Text	@Name([CN];@UserName)
PREQ_NAME	Text	@Name([CN];@UserName)

The four fields that users complete are:

- ITEM_NAME
- SHORT_TEXT
- QUANTITY
- DELIV_DATE

The following table describes the last three fields in the list:

Field name	Field type	Description
SHORT_TEXT	Text	Reason for Order
QUANTITY	Number	How many of the ITEM_NAMES they want
DELIV_DATE	Date/Time	This field is of Style Calendar/Time control and shows only month, day, and year. The user must pick a date when the item should be delivered by. You may choose to hard code a date one or two weeks from today.

The Field properties for the ITEM_NAME field include the following:

- Show field delimiters
- Use View dialog for choices using the current database and column 1 of the Materials view
- Display entry helper button
- Refresh fields on keyword change

You also need to add some script to the Exiting Event of the ITEM_NAME field to trigger a document recalculation:

```
Sub Exiting(Source As Field)
    Dim ws As New NotesUiWorkspace
    Dim uidoc As NotesUiDocument
    Set uidoc = ws.CurrentDocument
    Call uidoc.refresh
End Sub
```

In addition to the 22 fields that SAP requires, add four more fields to the form:

- MATERIAL
- MAT_GRP
- EXPORTSNUMBER
- RETURNMESSAGE

MATERIAL and MAT_GRP are text fields and are completed by the Postrecalc event of the new form. Here is the code for that event:

```
Sub Postrecalc(Source As Notesuidocument)
    Source.Autoreload = False
    If Source.Editmode = True Then
        Dim session As New notessession
        Dim collection As NotesDocumentCollection
        Dim dateTime As New NotesDateTime("12/01/94")
        Dim doc, materialdoc As notesdocument
        Dim db As notesdatabase
        Set db = session.currentdatabase
        Set doc = Source.document
        Set collection = db.Search(("MAKTX="|doc.ITEM_NAME(0)+"|",dateTime,0)
        Set materialdoc = collection.getFirstdocument
        doc.MATERIAL = materialdoc.MATNR
        doc.MAT_GRP = materialdoc.GROUP
    End If
    Call Source.Reload
End Sub
```

The EXPORTSNUMBER AND RETURNMESSAGE fields hold the results of the call to BAPI_REQUISITION_CREATE. The following table describes these two fields:

Field name	Field type	Description
EXPORTSNUMBER	Text	Will hold the Requisition Number if the call was successful. You can hide this field and any descriptive text just before it if the field is empty.
RETURNMESSAGE	Text	Will hold any error information if not successful. You can hide this field and any descriptive text just before it if the field is empty.

As to the design of the form, the world is our oyster. Here are two different designs: the first shows you all of the fields, and the second shows only what the user needs to see.

Requisition Form

REQUISITION_ACCOUNT_ASSIGNMENTCOSTCENTER: COSTCENTER T

REQUISITION_ACCOUNT_ASSIGNMENTG_L_AOCT: G_L_AOCT T

REQUISITION_ACCOUNT_ASSIGNMENTPREQ_ITEM: APREQ_ITEM #

REQUISITION_ACCOUNT_ASSIGNMENTPREQ_QTY: PREQ_QTY #

REQUISITION_ITEMS

ACCTASSCAT	C_AMT_BAPI	DOC_TYPE	ITEM_CAT	PLANT	PREQ_ITEM	SHORT_TEXT
ACCTASSCAT	C_AMT_BAPI	DOC_TYPE	ITEM_CAT	PLANT	PREQ_ITEM	SHORT_TEXT

CURRENCY	UNIT	MAT_GRP	PUR_GROUP	CREATED_BY	PREQ_NAME	IR_IND	GR_IND
CURRENCY	UNIT	MAT_GRP	PUR_GROUP	CREATED_BY	PREQ_NAME	IR_IND	GR_IND

Item Name: ITEM_NAME # MATERIAL T

Quantity: QUANTITY #

Delivery Date: DELV_DATE [C]

Error Message: RETURNMESSAGE T

Requisition Number: EXPORTSNUMBER Y

This form holds everything BAPI_REQUISITION_CREATE needs. Now let's look at what this form displays to end users:

SRM Enterprises Requisition Form

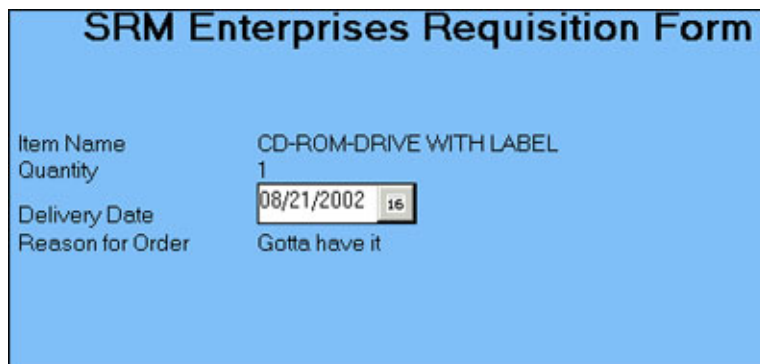
Item Name: [Pop-up list]

Quantity: [Text field]

Delivery Date: [Calendar control]

Reason for Order: [Text field]

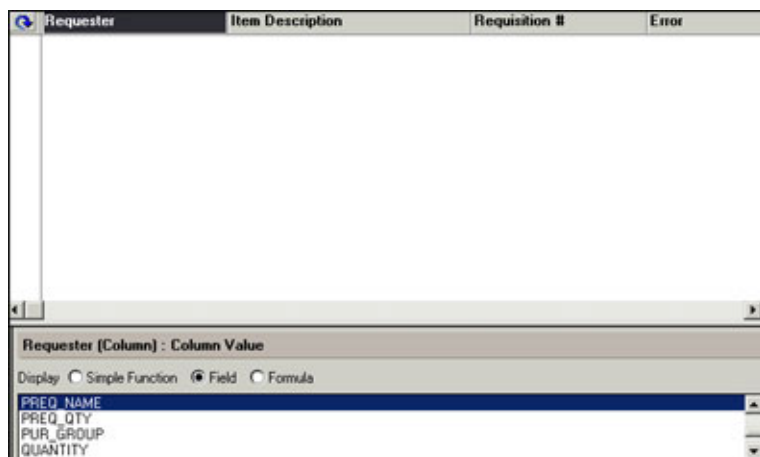
How does it work? You select the Item Name from the pop-up list. Any time you select a value, instead of typing one, the error count decreases. Enter how many items you want. You can build in logic to limit this, if you need to. Pick a date from the Calendar Control, and finally, enter a reason for your order. Save this document. If there are no errors, you are done.



If you build in workflow, this document may be sent to a database that serves as the collection point to send requests to SAP. Maybe that database is in another city or country. Adding Domino workflow functionality can enhance what SAP can do. In this example, we read directly from the Buy It Database and send the data to SAP.

The Requisitions view

The Requisitions view is the default view that opens when the database is opened. It displays all requisitions. The view contains four columns that map to fields Requestor, Item Description, Requisition #, and Error as shown in the following screen.



The following table lists the view columns and fields that each column maps to.

Column	Field
Requester	PREQ_NAME
Item Description	ITEM_NAME
Requisition #	EXPORTSNUMBER
Error	RETURNMESSAGE

Calling BAPI_REQUISITION_CREATE

You can set up a periodic Direct Transfer Activity in LEI to send requests to SAP. There are three things that you need:

- A data source (Buy It Database)
- A data target (SAP Connector and BAPI_REQUISITION_CREATE)
- A Direct Transfer Activity to move the data along

You need a new Notes Connection document for the Buy It Database because the view to use for the selection value is not what you want for this new activity. Create a new Notes Connection document that reflects the same values as the previous Notes Connection document before you edited it to work with Replication Activities.

Notes Connection	
Connection Properties	
Name:	Buy It Database
Domino Server:	eitech
Notes Database:	materials.nsf

Next, create another SAP connection document. In it, list BAPI_REQUISITION_CREATE in the Name and RFC or BAPI fields.

SAP Connection		Author: Scott Morris/EITECH
Connection Properties		
Name:	BAPI_REQUISITION_CREATE	
Hostname:	eitech	
Destination:	E11	
System Number:	0	
Client:	000	
User Name:	muster	
Password:	XXXXXXXX	
Language:	E	
Select RFC, BAPI or Transaction:	<input type="radio"/> RFC <input checked="" type="radio"/> BAPI <input type="radio"/> Transaction	
RFC or BAPI:	BAPI_REQUISITION_CREATE	
Debug Level:	0	
RFC User:	0	
Password NOT Encrypted		

Next, you need a Direct Transfer Activity. The fields in the first half of the Direct Transfer Activity document should reflect the field values in the following table:

Field name	Value
Activity Name	BAPI_REQUISITION_CREATE
Source fields	
Connection Name	Buy It Database
Form Name	Requisition
Select Statement	EXPORTSNUMBER=""&RETURNMESSAGE=""
Target fields	
Connection name	BAPI_REQUISITION_CREATE
Table Name	*

This next screen shows the rest of the Direct Transfer Activity. The right column titled Target Field list contains the fields that were given to us by our SAP ABAP developers. In addition, those developers also provided the default values that we filled in on our form.

Field Mapping <input type="checkbox"/> Automatic	
Source Field List:	Target Field List:
COSTCENTER	REQUISITION_ACCOUNT_ASSIGNMENTCOST_CTR
G_L_ACCT	REQUISITION_ACCOUNT_ASSIGNMENTG_L_ACCT
APREQ_ITEM	REQUISITION_ACCOUNT_ASSIGNMENTPREQ_ITEM
PREQ_QTY	REQUISITION_ACCOUNT_ASSIGNMENTPREQ_QTY
ACCTASSCAT	REQUISITION_ITEMSACCTASSCAT
CREATED_BY	REQUISITION_ITEMS_CREATED_BY
CURRENCY	REQUISITION_ITEMS_CURRENCY
C_AMT_BAPI	REQUISITION_ITEMS_C_AMT_BAPI
DELIV_DATE	REQUISITION_ITEMS_DELIV_DATE
DOC_TYPE	REQUISITION_ITEMS_DOC_TYPE
GR_IND	REQUISITION_ITEMS_GR_IND
IR_IND	REQUISITION_ITEMS_IR_IND
ITEM_CAT	REQUISITION_ITEMS_ITEM_CAT
MAT_GRP	REQUISITION_ITEMS_MAT_GRP
PLANT	REQUISITION_ITEMS_PLANT
PREQ_ITEM	REQUISITION_ITEMS_PREQ_ITEM
PREQ_NAME	REQUISITION_ITEMS_PREQ_NAME
PUR_GROUP	REQUISITION_ITEMS_PUR_GROUP
QUANTITY	REQUISITION_ITEMS_QUANTITY
SHORT_TEXT	REQUISITION_ITEMS_SHORT_TEXT
UNIT	REQUISITION_ITEMS_UNIT
MATERIAL	REQUISITION_ITEMS_MATERIAL

The Select Statement in the Direct Transfer Activity document is a Notes formula. It limits the documents that we send to SAP to ones that have no value in either of these two fields. To have values in these fields, the activity must put them there. How? An excellent new feature called Output Parameter Support was introduced in LEI 3.2. This feature returns the results of the Direct Transfer Activity for us.

Direct Transfer Options	
Precision	
<input type="radio"/> Generate Error for any Data Loss <input checked="" type="radio"/> Allow Precision Loss Only <input type="radio"/> Truncate Data When Necessary	
Source Data Target Data	
Target Data	
<input type="checkbox"/> Create Target Metadata <input type="checkbox"/> Overwrite Existing Data <input type="checkbox"/> Try Update Before Insert <input checked="" type="checkbox"/> Target Metadata Is Stored Procedure <input type="checkbox"/> Continue On Insert Errors <input type="checkbox"/> Disable Trimming of Text Trailing Spaces <input checked="" type="checkbox"/> Accept Output Parameters into Source	
Stored Procedure Output Mapping	
Source:	Target:
EXPORTSNUMBER	EXPORTSNUMBER
RETURNMESSAGE	RETURNMESSAGE

In the Direct Transfer options under the Target Data section, check off the Target Metadata is Stored Procedure option. A new section appears in the form called Accept Output Parameters into Source. If you select this option, the Stored Procedure Output Mapping section becomes visible. Then, you can type in fields that SAP returns. These fields are mapped to the source document in Notes. Until LEI 3.2, this functionality could only be achieved via script. Now it can be done with no scripting required.

We set up scheduling to have this run as often as we need—once a day or an hour or whenever we choose. After this Direct Transfer Activity runs, our source requisition document looks like this:

SRM Enterprises Requisition Form	
Item Name	CD-ROM-DRIVE WITH LABEL
Quantity	1
Delivery Date	08/21/2002 16
Reason for Order	Gotta have it
Requisition Number	0010008324

The requisition number now displays, telling us that SAP accepted this request. If we like, we can add a Dependent Activity to this request. This activity could trigger an agent to run and send an email to our user to tell him or her that the requisition was accepted (along with the new requisition number) or that there was a problem. To do this, we must do three things:

1. Create an agent, perhaps in the Buy It Database, that will generate emails for us.
2. Create a scripted activity in LEI to run this agent for us after our BAPI_REQUISITION_CREATE activity has run.
3. Update our BAPI_REQUISITION_CREATE activity to list our new scripted activity as dependent.

Create an agent called Send Mail. The agent should run manually from the Actions menu. Here is the code for the agent and its corresponding formula for sending emails that include doclinks to the requisition documents:

```
@If(EXPORTSNUMBER != "";@MailSend(CREATED_BY;"";"Requisition " + EXPORTSNUMBER + "successfully completed in SAP";"Link to source Requisition document";"";"Error sending requisition for " + ITEM_NAME + " to SAP";"Link to source Requisition document";"";[IncludeDocLink]);"";SELECT @All
```

Here is the Scripted Activity from LEI:

Scripted Activity	
Activity Name:	Requisition Status Mail
Current Status:	Disabled
Last Completed Run:	Time: 11:26:39 AM Today Result: Activity completed successfully
Script	
Agent Server:	eitech
Agent Database:	materials.nsf
Agent Name:	Send Mail

The Scripted Activity points to our Buy It Database and agent. Finally, we add this entry in our BAPI_REQUISITION_CREATE activity and our application is complete! The following screen shows the Dependent Activity.

General Options	
General Options	
Designated Server:	
Dependent Activity(s):	Requisition Status Mail

We've successfully created a Domino application that can retrieve SAP data, that allows users to purchase items, and that sends requests back to our SAP system.

Summary

We went through a lot in this example:

- We downloaded the material inventory from SAP into Domino using LEI without writing any LotusScript.
- We built a requisition form for our Notes users, which hides all of the complexity of using

BAPI_REQUISITION_CREATE from them.

- We uploaded the requisitions into SAP, which returned either a requisition number or error information. We placed this information in the Notes source document.
- We sent an email to our users to let them know what happened.

Saving money is always a good thing, but even more so in today's economy. So how did we save money?

- We needed only one SAP user license to download the material inventory from SAP into Domino. From there, this copy was available to all Domino users.
- We needed only one SAP user license to upload the requisition requests to SAP from Domino because it was done by a server process and not live by individual users. The requestor information (name) was recorded in fields sent to BAPI_REQUISITION_CREATE, so we know who placed an order for what in both Notes and SAP.
- Our users did not require training to learn how to navigate the hundreds of possible screens of transaction ME21, the Purchase Requisition system in SAP.
- The only new software installed on the desktop of each user is the Notes icon to our Buy It Database, which lives on Domino servers. Nothing else was required.

We also enabled potentially remote Notes users, or Notes mobile users, to create SAP purchase requisitions, helping us get more out of our SAP investment without a paper-based process.

I hope that you found this article useful. For more information, I recommend that you check out the Lotus Developer Domain [Enterprise Integrators product page](#) and the [Lotus Enterprise Integrators page](#) on Lotus.com. There we have more SAP specific information, LEI, DECS, and LCLSX information as well as valuable forums, including one for SAP alone and the other for all Enterprise Integration products.

ABOUT THE AUTHOR

Scott Morris is a Senior Software Engineer with IBM. He has been with Lotus/IBM since 1990. For the past six years, he has worked on developing Domino-SAP integration products offered by the Domino Enterprise Integration group. He was previously a member of the Notes API team, the Notes (before there was a Domino) server team, and somewhere in the deep dark past, the 1-2-3 for Macintosh team. Scott has an MS in Computer Science from Boston University and a BS in Mathematics from Carnegie Mellon University.