Digital Signature for Custom Application Object

Applies to:

SAP ECC6.0 EHP4. For more information, visit the ABAP homepage.

Summary

The objective of this document is to provide guidelines on how to implement Digital Signature functionality for applications which are not enabled by default to use the subject functionality. The guidelines would involve configuration as well as development steps on implementing "*Digital System Signature with User ID and Password*". The target audience is SAP Functional and Technical Consultants.

Author: Suresh Radhakrishnan

Company: Accenture Services Ltd.

Created on: 25 July 2011

Author Bio

Suresh is an SAP Technical Manager with Accenture Services Ltd – India Development Centre. Suresh has 10 years of work experience in SAP RICEFW development.

Table of Contents

Approval using Digital Signature

Certain industries such as Pharmaceutical or Food-Processing have to comply with stricter regulations with regard to the documentation and approval of their processes (such as, the guidelines on *current Good Manufacturing Practices (cGMP)*, which were laid down by the U.S. Food and Drug Administration and are an international standard).

In addition, the increasing use of electronic data processing in companies also requires security mechanisms to protect digital data. Legislation such as the *Final Rule on Electronic Records and Electronic Signatures*, 21 CFR Part 11, issued by the FDA reflects this need.

For this reason, the SAP System contains the digital signature, a tool that enables you to sign and approve digital data. The digital signature ensures that the person signing a digital document is uniquely identified and that the signatory's name is documented along with the signed document, date, and time. You can use digital signatures to approve documents or objects in all the applications that are able to use it.

SAP provides three Signature Methods:

1. System signature with authorization by user ID and password
2. Digital user signature with verification
3. Digital user signature without verification (Only for Test Purposes)

While the second and third method requires external security product to verify the Actual Signature of users, the first method doesn't. The user name and ID are part of the signed document. This document deals with the first method.

Scenario

Let us consider an example scenario where we need to implement digital signature. Whenever there is a scheduled change to the material master, the system would trigger an approval workflow before the scheduled changes are effected. The approver has to digitally sign the changes before they are activated. This document would not elaborate on the workflow tasks, but would focus only on the digital signature part.

The following sections show the IMG customizing and development steps involved in implementing the same

IMG Steps

Define Authorization Group

IMG Path: Cross Application Components -> General Application Functions -> Digital Signature -> Signature Strategy -> Define Authorization Group

In this IMG activity you define the authorization group for digital signatures. Authorization group is used to restrict the authorization for executing digital signatures in the applications. In the user master record, you assign authorizations for the authorization group that corresponds to the user's area of responsibility (authorization object C_SIGN_BGR). Depending on the applications, it may also be necessary to specify the authorizations for the C_SIGN authorization object.

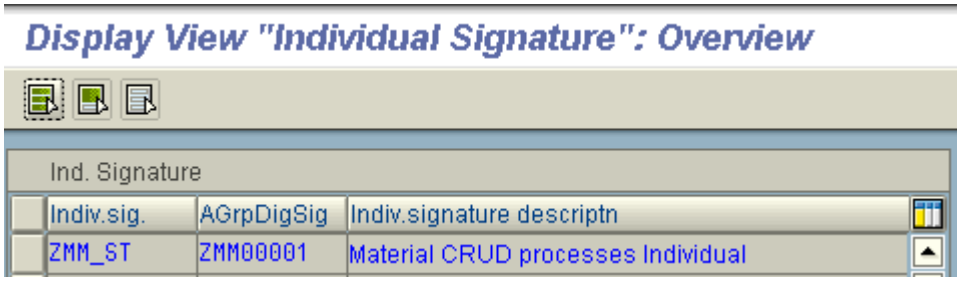
For our scenario, let us configure an authorization group ZMM00001 as shown below.

Define Individual Signatures

IMG Path: Cross Application Components -> General Application Functions -> Digital Signature -> Signature Strategy -> Define Individual Signatures

In this IMG activity, the digital individual signature that must be executed by users in a specific authorization group is defined and the Authorization Group which was created in the earlier step is assigned to it.

Let us define an individual signature ZMM_ST as shown below:

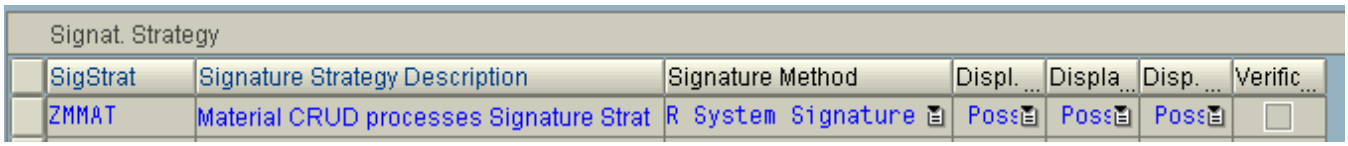


Define Signature Strategy

IMG Path: Cross Application Components -> General Application Functions -> Digital Signature -> Signature Strategy -> Define Signature Strategies

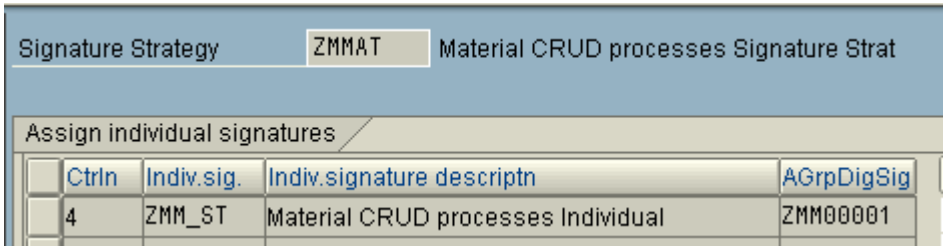
In this step, we define the Signature Strategy and assign the individual signature which we created earlier to the Signature Strategy.

Let us define a Signature Strategy ZMMAT:



In this step we choose Signature Method "*System Signature with Authorization by SAP User ID/Password*" and enable display of Comment, Remark and Document during signature dialog.

In this same IMG step, the individual signature should be assigned to the above Signature Strategy as below:

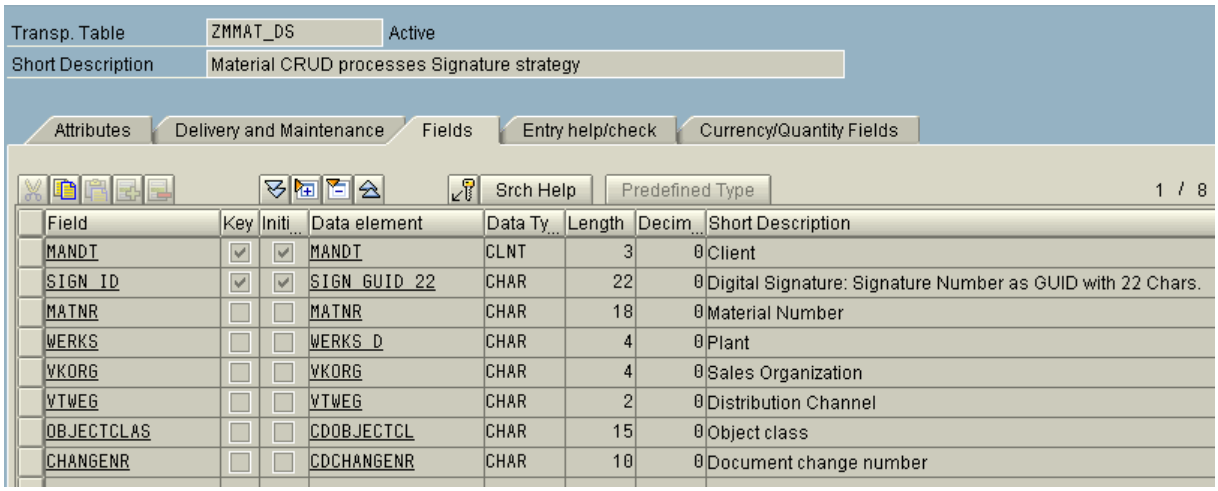


Workbench Development Steps

Standard SAP doesn't enable Material Approval as an application object registered to use Digital Signature. So the following custom developments and registration processes are required to enable the same.

Create Table for Application-Dependant Metadata

A transparent table is to be created in the data dictionary during the registration of an application for the digital signature. The MANDT and SIGN_ID must be there in the table as key fields. The other non-key fields should be application specific. The following table is created for our scenario.



Transp. Table ZMMAT_DS Active
Short Description Material CRUD processes Signature strategy

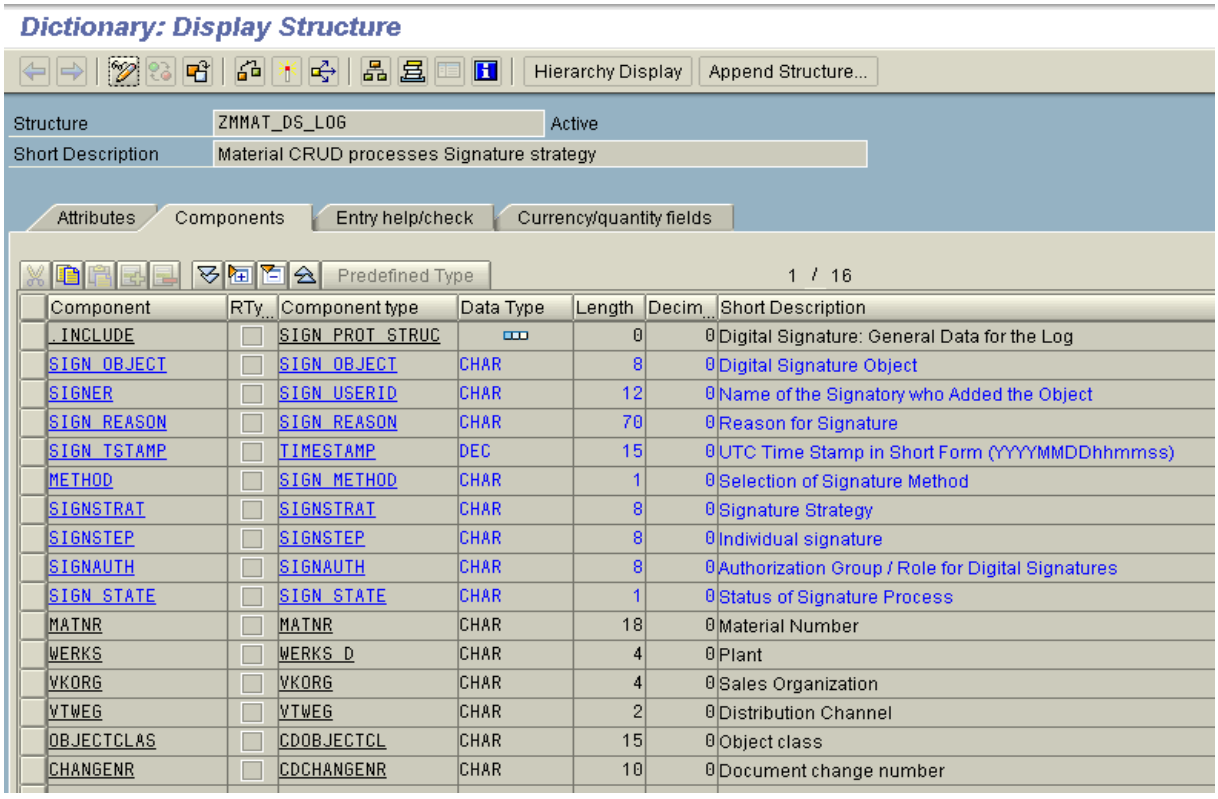
Attributes | Delivery and Maintenance | Fields | Entry help/check | Currency/Quantity Fields

Srch Help | Predefined Type | 1 / 8

Field	Key	Initi...	Data element	Data Ty...	Length	Decim...	Short Description
MANDT	☑	☑	MANDT	CLNT	3	0	Client
SIGN_ID	☑	☑	SIGN_GUID_22	CHAR	22	0	Digital Signature: Signature Number as GUID with 22 Chars.
MATNR	☐	☐	MATNR	CHAR	18	0	Material Number
WERKS	☐	☐	WERKS_D	CHAR	4	0	Plant
VKORG	☐	☐	VKORG	CHAR	4	0	Sales Organization
VTWEG	☐	☐	VTWEG	CHAR	2	0	Distribution Channel
OBJECTCLAS	☐	☐	CDOBJECTCL	CHAR	15	0	Object class
CHANGENR	☐	☐	CDCHANGENR	CHAR	10	0	Document change number

Create Structure to be displayed in Log

SAP has given a utility program to monitor the digital signature log (Transaction Code: *DSLOG*). For the purpose of displaying the log for our custom application object, we need to create a structure. It must contain the structure SIGN_PROT_STRUC as an include-structure. All the non-key fields from the above metadata table (which was created earlier) can then be included in the structure. The following structure is created for our scenario:



Dictionary: Display Structure

Hierarchy Display | Append Structure...

Structure ZMMAT_DS_LOG Active
Short Description Material CRUD processes Signature strategy

Attributes | Components | Entry help/check | Currency/quantity fields

Predefined Type | 1 / 16

Component	RTy...	Component type	Data Type	Length	Decim...	Short Description
.INCLUDE	☐	SIGN_PROT_STRUC		0	0	Digital Signature: General Data for the Log
SIGN_OBJECT	☐	SIGN_OBJECT	CHAR	8	0	Digital Signature Object
SIGNER	☐	SIGN_USERID	CHAR	12	0	Name of the Signatory who Added the Object
SIGN_REASON	☐	SIGN_REASON	CHAR	70	0	Reason for Signature
SIGN_TSTAMP	☐	TIMESTAMP	DEC	15	0	UTC Time Stamp in Short Form (YYYYMMDDhhmmss)
METHOD	☐	SIGN_METHOD	CHAR	1	0	Selection of Signature Method
SIGNSTRAT	☐	SIGNSTRAT	CHAR	8	0	Signature Strategy
SIGNSTEP	☐	SIGNSTEP	CHAR	8	0	Individual signature
SIGNAUTH	☐	SIGNAUTH	CHAR	8	0	Authorization Group / Role for Digital Signatures
SIGN_STATE	☐	SIGN_STATE	CHAR	1	0	Status of Signature Process
MATNR	☐	MATNR	CHAR	18	0	Material Number
WERKS	☐	WERKS_D	CHAR	4	0	Plant
VKORG	☐	VKORG	CHAR	4	0	Sales Organization
VTWEG	☐	VTWEG	CHAR	2	0	Distribution Channel
OBJECTCLAS	☐	CDOBJECTCL	CHAR	15	0	Object class
CHANGENR	☐	CDCHANGENR	CHAR	10	0	Document change number

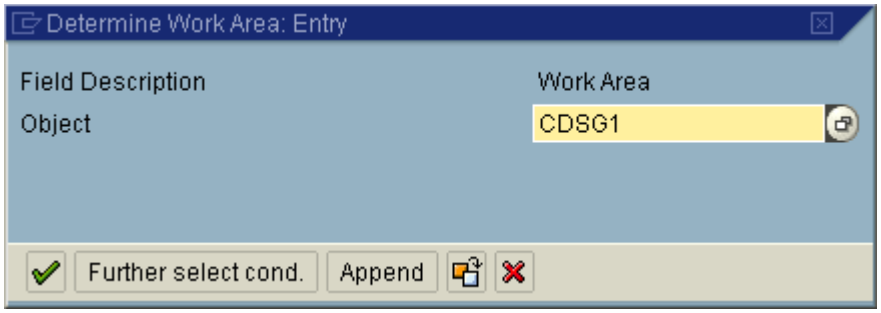
Register Signature Application

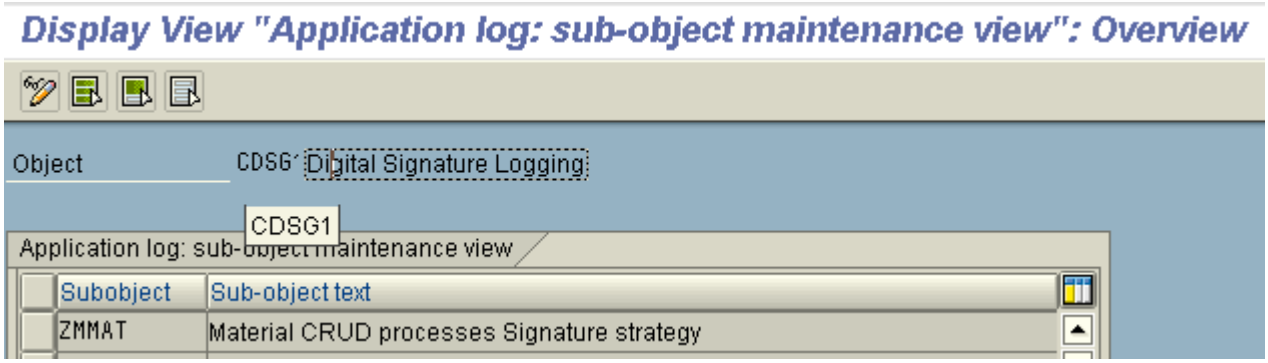
The application to be signed must be registered as a signature object in the SIGNAPPL table (Transaction Code: SIGNA):

Application Log: Sub-Objects Maintenance

This object should be configured as a sub-object for application log under the Object: *CDSG1* (Digital Signature Logging)

Maintain View V_BALSUB through SM30. This would be recorded in a workbench request:

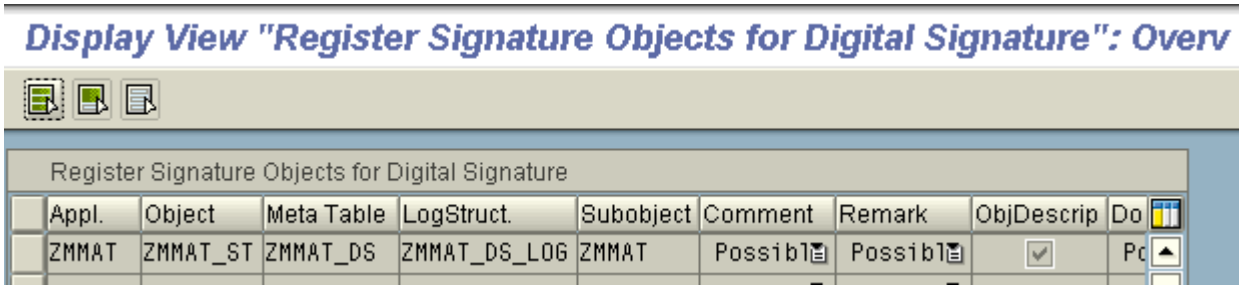




Register Signature Object

The application to be signed must be registered as a signature object in the SIGNOBJECT table (Transaction Code: SIGNO). Here we would specify the Meta Table and Log Structure which were created earlier for the custom application object along with the sub-object which was earlier registered for digital signature logging.

For our scenario we register these specifications with a name of the object as ZMMAT_ST



Develop Function Module to create Digital Signature

We need to develop a function module for creating digital signature. This function module is to be copied from QSS7_CREATE_SIGNATURE_LOT_UD and requires minor modification as below:

- Importing parameter IS_META to be typed as our metadata table(ZMMAT_DS)
- Local variable LS_META is to be typed as metadata table (ZMMAT_DS)
- The constants used for application and object, are to be defined to have values as our scenario application and object (Constants: CO_DS_APPLICATION_LOT_UD for application and CO_DS_SIGNOBJECT_LOT_UD for object). In our scenario we would declare constants for application *ZMMAT* and for the object *ZMMAT_ST*, in place of the above two constants.

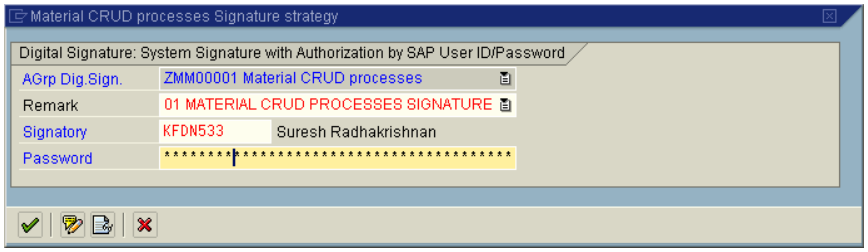
Calling Digital Signature from the application program

This function module is to be called from the application program where digital signature is required. As a pre-requisite the Signature Strategy is to be configured which is to be passes as a mandatory parameter IV_SIGNSTRAT. The following are the parameters to be passed:

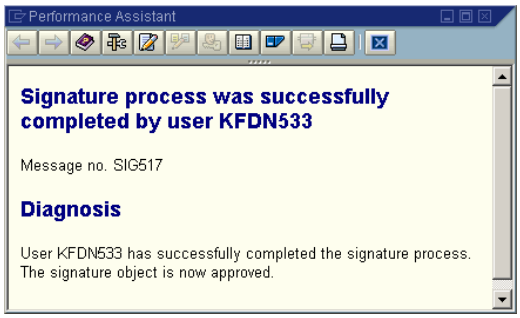
IS_META	Fill up all non-key fields of the metadata table fields from the application data to be signed
IV_DOCUMENT	This is an XSTRING – to be filled up from the required data of the application. The application data is to be converted to XSTRING using FM QSS7_CREATE_DOCUMENT.
IS_SIGN_REMARK	A version number and 68 character Remarks which would appear on Signatory's dialog as well as in the log
SIGN_TYPE	Refer to the fixed values of the domain SIGN_TYPE A Asynchronous Signature Strategy B Asynchronous Signature Strategy with Changeable Signatory C Synchronous Signature Strategy
IV_SIGNSTRAT	Pass the Signature Strategy configured for this purpose. The signature strategy should be configured through IMG. Please note that the strategy should use Signature Method: "*System Signature with authorization by SAP UserID/Password*". The current user should be assigned with the Authorization Group which is assigned to this Signature Strategy

COMMIT WORK needs to be executed after calling the above Function Module in order to successfully complete the digital signature process and logging.

On calling the above FM, the system pops up the digital signature for the signatory to sign:



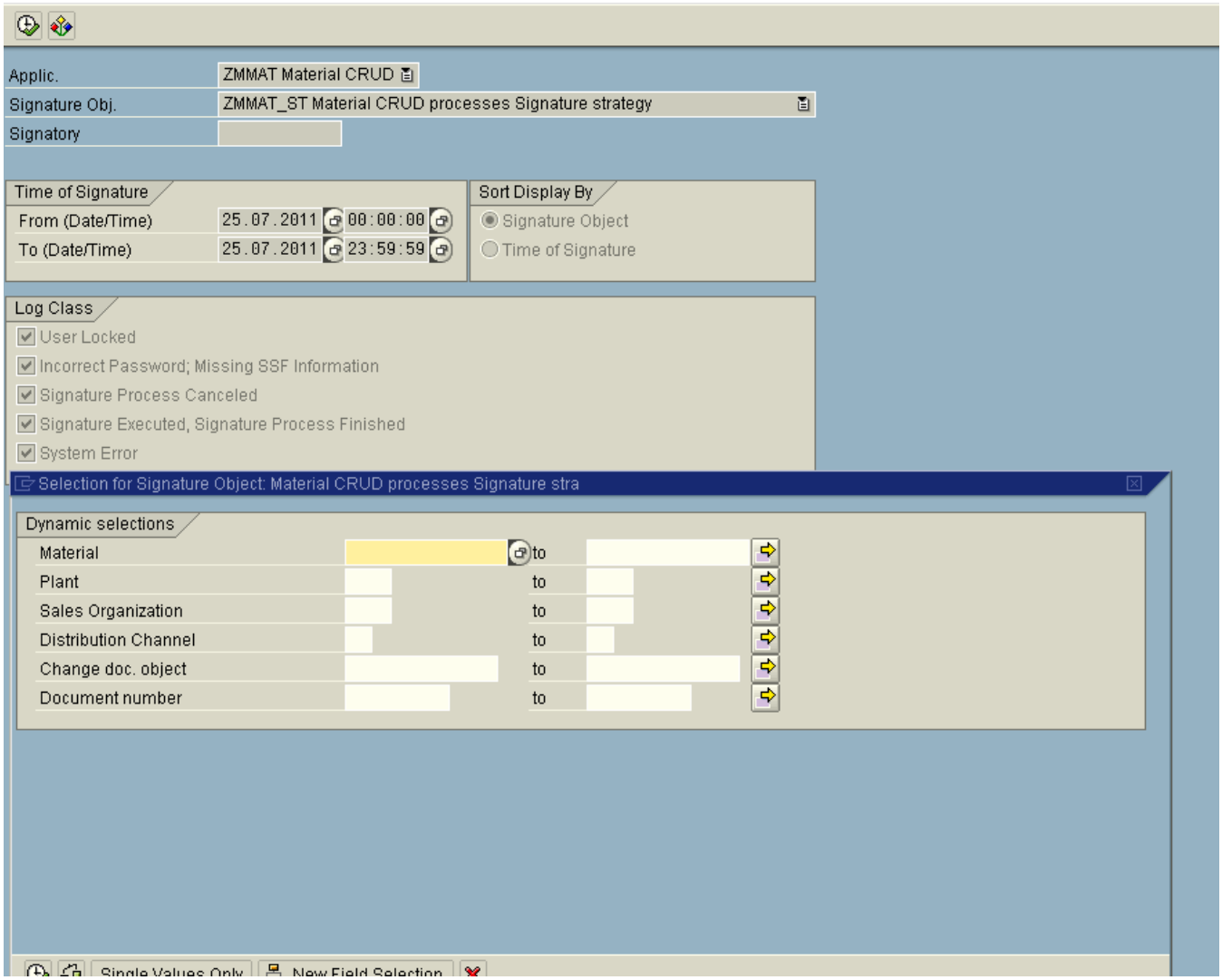
System displays a status message on successful signature:



Digital Signature Log

The digital signatures executed successfully as well as failed ones can be displayed through SAP standard utility transaction code DSLOG:

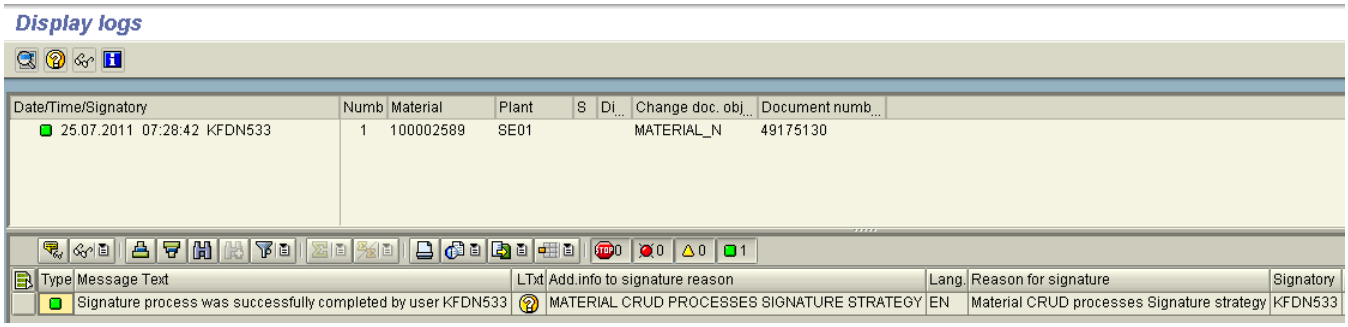
Log Display (Digital Signature)



The application specific fields can also be used in dynamic selection. In our scenario, all the non-key fields that were defined in the Meta data table ZMMAT_DS automatically appears here in dynamic selection.

The log shows date and timestamp with the object which was signed and details like remarks, signatory, etc. in the report.

Display logs



Related Content

Digital Signatures

Approval Using Digital Signatures

Digital Signature - Developer Guidelines and Best Practices

For more information, visit the ABAP homepage

Disclaimer and Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.