



FLM for HR 420

Overview of FLM Components

Author: Sam Dudley

File name: FLM for HR 420 Overview v1 1



Version History

Version	Author	Date	Reason for update
0.1	Sam Dudley	29/05/2011	Draft
0.2	Sam Dudley	31/10/2011	Revision
0.3	Sam Dudley	25/01/2012	Posting Update
1.0	Sam Dudley	29/02/2012	Release Version
1.1	Sam Dudley	01/09/2012	FLM 420 updates

Related Documents

Document	Date	Description

Table of Contents

Version History	2
Related Documents	2
Table of Contents	3
1 FLM for HR	4
2 Using FLM for HR.....	5
3 Technical Components.....	6
3.1 How is FLM for HR Delivered?	6
3.2 FLM User Exit Includes	6
4 Form Pre-population	7
4.1 Form level Include (/FLMHR/FORM_PREPOP)	7
4.2 Form level Configuration (/FLMHR/PRE_CNTRL)	8
4.3 Field level Configuration (/FLMHR/FLD_CNTRL)	9
4.4 Project Variables Configuration (/FLM/TVARVC)	11
5 Form Dropdowns.....	12
5.1 Field level Dropdown Values Include (/FLMHR/F4_PREPOP)	12
5.2 Field level Dropdown Configuration (/FLMHR/F4_CNTRL)	12
5.3 Simple F4 Dropdown (/FLMHR/HRF4_GET_SIMPLE)	13
5.4 Complex F4 Dropdown (/FLMHR/HRF4_GET_ARRAY)	15
6 Form Posting	17
6.1 Posting Options	17
6.2 Create PA Action / Infotypes (/FLMHR/FPE_ACTION_PLUS_700)	18
6.3 Swap Action (/FLMHR/FPE_SWAP_ACTION_700).....	20
6.4 Simple Infotype updates and delimits (/FLMHR/FPE_INFTOTYPE_700).....	20
7 HR for FLM in action	23
7.1 A simple example	23
7.2 Adding infotype delimiting to the example.....	24
7.3 Adding an HR Action to the example	24
8 Appendix A – Popular Pre-populations.....	26
9 Appendix B – Popular Dropdowns.....	30



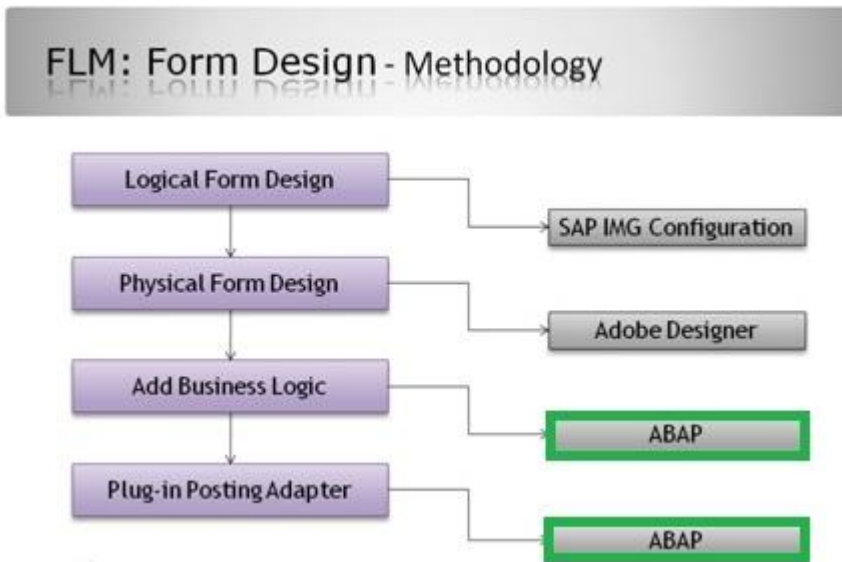
FLM for HR is comprised of a number of enhancements to the standard FLM product and supports HR specific form process development as follows

- HR Master Template with delivered subforms for common infotypes
- Configuration driven
 - Pre-population of PA and PD infotypes
 - Dropdown Lists (SAP F4)
 - Posting adaptors to process
 - PA infotypes
 - PD infotypes
 - PA40 actions

In different pre-releases Arch will deliver the FLM for HR objects in either a “HR” FLM package or by referencing items using ACL rather than the client customer code in object naming. No changes should be made to any Arch objects without prior consent. Product issues should be logged via a case using FLM Connect. Any copies of the code made are outside the warranty of Arch.



The normal form design methodology is still followed but can be rapidly accelerated by using elements from the HR Master Template and related configuration.



The FLM for HR framework further accelerates development in both the “Add Business Logic” and “Plug-in Posting Adapter” phases.



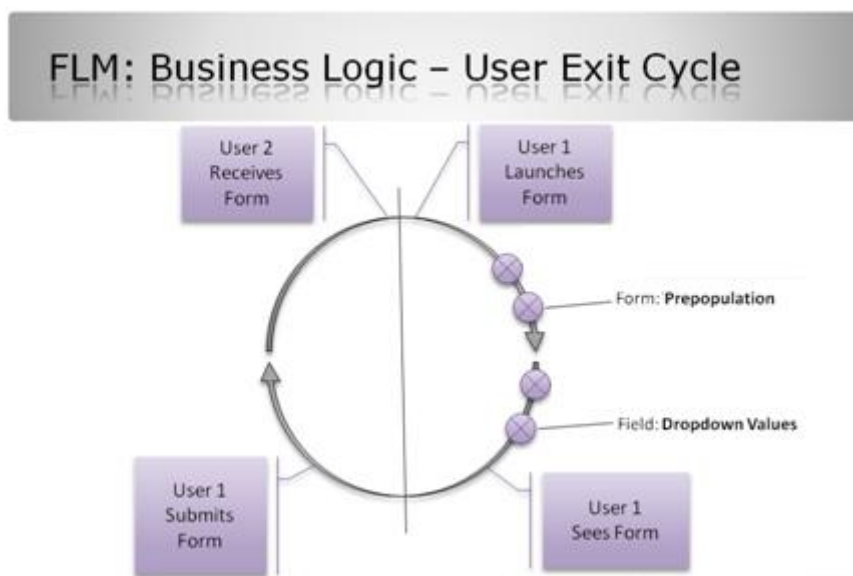
3 Technical Components

3.1 How is FLM for HR Delivered?

FLM for HR is delivered via an example form template (ACL/HRXX) configuration tables, ABAP includes and function modules. The section of the document introduces each of these objects in turn.

3.2 FLM User Exit Includes

The following section outlines the FLM for HR enhancements delivered by ABAP includes which can be implemented in the relevant FLM user exits.





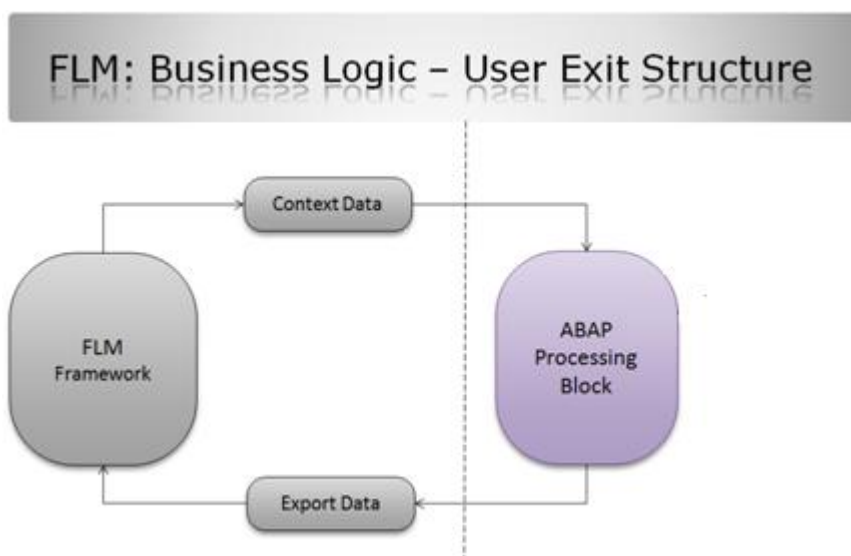
4 Form Pre-population

4.1 Form level Include (/FLMHR/FORM_PREPOP)

When coding a new Form level pre-population user exit the following steps are normally followed:

- A Remote destination, if relevant, is read from FLM configuration
- Function(s) are called passing in form data and returning variables or tables
- Single Form fields are updated from the returned variables
- Repeating row fields are updated from the returned tables

This include and related configuration table provides a single framework to increase the reusability of ABAP code and reduce complexity in the FLM form level user exit. This include must be manually added to the form level user exit code for each form using the HR for FLM framework.



The selection of remote destinations and calling of functions is driven from a configuration table (/FLMHR/PRE_CNTRL) based on form type. Each function shares a common interface, form data passed in, and a table of form fields/values to update returned. The logic and grouping of these functions is dependent on the customer requirements and can range from one function per field, one function per section or even one function per form.

Once all functions have been processed the complete table of fields and values is then processed to update single fields or repeating rows. Repeating rows are supported by ensuring the HEIGHT value in the return table is set to the correct row number.



4.2 Form level Configuration (/FLMHR/PRE_CNTRL)

This table is used by the form level user exit include (/FLMHR/FORM_PREPOP) to set the sequence of functions which must be called to successfully pre-populate a form using FLM for HR.

Table Field	Description	Example Data
CCODE	FLM Customer Code	ACL
FTYPE	FLM Form Type	TS00 (or can be left blank if relevant for ALL forms)
FVER	FLM Form Version	00
FSTATUS	FLM Form Status	I (or can be left blank if relevant for any status set for pre-population in FLM configuration)
STEP	Processing Order	1
PREPOP	FLM Pre-pop type	X (or can be C for check cycle or B if both)
PROC_FUNCTION	Name of an RFC enabled function created with correct interface	/FLMHR/PRE_GLOBAL
RFCDEST	RFC Destination to retrieve values from alternative SAP system	<blank>

In the example above the pre-delivered function /FLMHR/PRE_GLOBAL is referenced. This function should always be the first pre-population function to be configured in /FLMHR/PRE_CNTRL (per form) and performs two key tasks.

- Pre-populates some generic form fields
 - o TXT_INITIATOR (employee name read from infotype 0002)
 - o TXT_INITIATOR_PER (employee number determined from infotype 0105)
 - o DT_REQUEST DATE (current system date)
 - o TXT_MOLGA (country grouping of user logged in)
 - o TXT_MOLGA_CURRENCY (currency of company code linked to country grouping)
 - o TXT_IM_DOCUMENT (passed via URL parameter FLMIMDocument)
- Processes the entries held within table /FLMHR/FLD_CNTRL that allows infotype data to be read into fields on the form based on relationships to the initiator, IM Document or other HR Objects
- Calls further customer defined function modules which can
 - o Read previously populated values for further derivation or update
 - o Add new pre-population values for form driven complex dropdowns
 - o Add new pre-population values for fields requiring complex business logic



4.3 Field level Configuration (/FLMHR/FLD_CNTRL)

Each form field will generally be populated from an infotype relating to a person, position, org unit, job or other HR Object type. The related HR Object ID is also very likely to be either passed into the form via the form URL or derived from it, or the initiator of the form.

This table is used by the generic function (/FLMHR/PRE_GLOBAL) to set the sequence of infotype reads required to pre-populate form fields. The table is also used to support posting and as such only fields relevant for pre-population are listed below.

Table Field	Description	Example Data
CCODE	FLM Customer Code	ACL
FTYPE	FLM Form Type	TS00 (or can be left blank if relevant for ALL forms)
FVER	FLM Form Version	00
FIELD_NAME	FLM Form Field	TXT_EMP_LAST_NAME
OTYPE	HR Object Type (for Object ID below)	P
OBJID	HR Object ID	TXT_IM_DOCUMENT
PRIORITY	Priority to force infotype read to be performed out of numerical order. NB: 99 is highest priority.	00
INFTY	HR Infotype number	0002
SUBTY	HR Infotype Subtype (optional)	< left blank >
INFTY_FIELD	HR Infotype field	NACHN
PREPOP	FLM Pre-pop type	X (or can be C for check cycle or B if both)
DEF_VAL	A fixed default value	< left blank >
NOTES	Free form notes to document entry	Employee Surname

The above entry in the table would be processed for form type TS00 and is set to pre-populate form field **TXT_LAST_NAME**. Infotype **0002** for the **Person TXT_IM_DOCUMENT** (passed in URL) will be read and the value of **NACHN** passed into the form field. If multiple fields from the same infotype are requested then a single call is made to SAP once the complete field list is known. Infotypes are read in numerical order e.g. 0000 to 9999.

To encourage reuse of field names across forms and to minimise the number of entries in this configuration table the form type (FTYPE) and version (FVER) can be left blank. In the case of **TXT_EMP_LAST_NAME** this would be evaluated when each form type was initially rendered, even a form triggered based on a Position.

There are three possible outcomes in this scenario for the Position based form; the first two cause no issue.

- There is no Person record against the position object id e.g. number ranges are different so no data is read and no pre-population can occur

FLM for HR 4.20

Overview



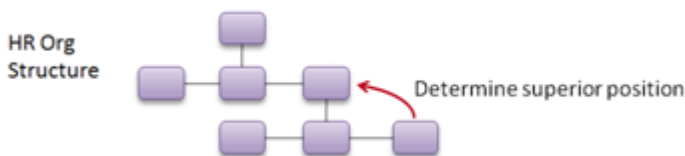
- There is an overlap in the number range and (incorrect) data is read from SAP! However in this case there is not likely to be a field on the form called TXT_EMP_LAST_NAME so it is ignored

The third however cannot be ignored but is only relevant in cases where the employee field appears on the position based form and object ID number range overlap. In these cases you must add a form specific entry for TXT_EMP_LAST_NAME which uses the correct reference for the employee Object ID. This entry must be based on the S<>P relationship (see TXT_POS_EMP_ID in Appendix A).

Where relevant the SUBTY field can be used to further restrict which records are read from the PA infotypes e.g. Objects on Loan, Payments etc. As mentioned above when dealing with infotype 1001 its use becomes very powerful to derive the value for other related form fields. This use can be extended by creating “temporary variables” in the configuration which can be used to determine the final value for a form field.

An example may be where a form, launched for an employee, needs to hold the name of their HR Administrator. At company Z the following HRP1001 relationships have been configured

- Employees (P) are assigned a position (P>S 008)
- Positions (S) are assigned to an Org unit (S>O 003)
- Key Organisational Units (O) have an expert assigned (O>XP ZXP)s
- Each HR Administrators is assigned as an expert (P>XP 601)



ABAP Code could be written into the pre-population user exit starting from the employee number in TXT_IM_DOCUMENT and determine the final value TXT_HRADM_NAME. However this can be achieved using linked configuration entries. Note however that these must be performed in a set order, so we use the priority to ensure they occur in the correct order.

FIELD_NAME	OTYPE	OBJID	INFTY	SUBTY	PRIORITY	NOTES
000_EMP_POS_ID	P	TXT_IM_DOCUMENT	1001	B008	99	Position ID link to Employee
000_EMP_ORG_ID	S	000_EMP_POS_ID	1001	A003	98	Org Unit ID link to Position
000_ORG_XP_ID	O	000_EMP_ORG_ID	1001	AZXP	97	Expert ID link to Org Unit
000_XP_EMP_ID	XP	000_ORG_XP_ID	1001	A601	96	Employee ID link to Expert
TXT_HRADM_NAME	P	000_XP_EMP_ID	0001	NACHN	00	Expert Name

After the processing of the table the “000” entries will be lost, leaving only TXT_HRADM_NAME being passed back and pre-populated on the form.

Appendix A provides a list of standard form fields and their related pre-population configuration for reference.



4.4 Project Variables Configuration (/FLM/TVARVC)

This table is delivered with the standard FLM product (as of 295 sp2) and should be used by FLM developments to hold variable values against a fixed ID in order to reduce hard coding. The table is often used to hold the references to central email addresses which may change in the future. Each entry in the table must be explicitly read by custom code within the FLM developments.

Table Field	Description	Example Data
SYSIS	SAP System ID (you will need to duplicate each entry for each SAP system using FLM for HR)	DEV
CLNT	SAP Client (not used by FLM for HR)	100
CCODE	FLM Customer Code	ACL
FTYPE	FLM Form Type	TS00 (or can be left blank if relevant for ALL forms)
FLANG	FLM Form Language (optional)	< left blank >
FVER	FLM Form Version (not used)	< left blank >
FIELD_NAME	Identifier for this entry (see special values in section below)	EMAIL_HR
SEQ	Sequence to support multiple values	0
VALUE	Range style entry Sign / Option / Low / High. Custom code reading the entry can further manipulate as required. (see special uses below)	HR@company.com
NOTES	Description of this entry	HR Email Address

This table is also used the FLM for HR enhancements for the following purposes:-

- Default values for Form field pre-population (FPRE_<form field>)
- Filtering possible entries in dropdowns (F4_<identifier>)
 - o Note that values are stored in a “ranges” style and are used to delete entries from the current list of possible entries i.e. to delete a possible value with the key “01” you should pass I,EQ,01
- Link Wagetype dropdowns to specific infotypes for filtering possible entries



5.1 Field level Dropdown Values Include (/FLMHR/F4_PREPOP)

This include and related configuration table provide a single framework to increase the reusability of ABAP code and reduce complexity in the FLM F4 user exit. This include must be manually added to the field level F4 level user exit code for each DD field using the HR for FLM framework.

The selection of remote destinations and calling of functions is driven from a configuration table (/FLMHR/F4_CNTRL) based on field name (and if necessary form type). Each function can share one of three different possible interface signatures, if relevant all form data and the current field can be passed in, and in all cases a table of possible codes and display values is returned.

There are two main sources of dropdown values, a simple F4 based on SAP screen fields or function modules (Arch delivered or custom). You can also use both simple F4 and functions in which case the results of the simple call are available for use and manipulation in the custom one. Once the list of values has been generated there are three further options that can be applied as below

- You can REMOVE entries from the list by creating an ID on the project variables table (or custom variables) and storing in here ranges or single code values to exclude
- You can define the sort to be by SAP Code or Display Value
- You can set the display to include both the code and value, separated by a dash. Note that the sort above is applied before the code and value are merged

The list is then returned to the field level F4 user exit where you can place further form and field specific ABAP if requirements have not been met. E.g. To move the “United Kingdom” entry to the top of the list of countries, previously read from SAP, filtered and then sorted by value.

5.2 Field level Dropdown Configuration (/FLMHR/F4_CNTRL)

This table is used by the field level F4 user exit include (/FLMHR/F4_PREPOP) to set the sequence of functions which must be called to successfully pre-populate any dropdown values in a form using FLM for HR.

Table Field	Description	Example Data
CCODE	FLM Customer Code	ACL
FTYPE	FLM Form Type	TS00 (or can be left blank if relevant for ALL forms)
FVER	FLM Form Version	00
FSTATUS	FLM Form Status	< ignored >



FIELD_NAME	FLM Form Fieldname	DD_GENDER
F4_SCREEN_FIELD	SAP Screen Field (F1 Tech Details)	P0002-GESCH
F4_CODE_FIELD	Code field from SAP Search Help	_LOW
F4_VALUE_FIELD	Value field from SAP Search Help	_TEXT
PROC_FUNCTION	Arch delivered / Custom F4 function	<blank>
RFCDEST	RFC Destination to retrieve values from alternative SAP system	<blank>
F4_FILTER_ID	ID referencing a key in project or customer variables table used to remove entries	F4_UNKNOWN (values in project variables table remove the "unknown" option from the default list passed by SAP)
F4_SORT	Flag to sort the final results by Code or Value, either Ascending or Descending e.g. CA,CD, VA, VD	VA
F4_CODEVAL	Display Code and Value with separator?	<blank> = value only e.g. Male X = key + value e.g. 1-Male

Appendix B provides a list of standard dropdown fields and their related configuration for reference. The next section describes how the functionality of the simple F4 driven by SAP screen fields works and how configuration can be determined and tested for new fields.

5.3 Simple F4 Dropdown (/FLMHR/HRF4_GET_SIMPLE)

SAP provides "possible values" for most input fields with F4, and in most cases for FLM Form fields can take advantage of this same functionality. Behind the screen fields are search helps, which provide a list of possible values, normally a code and description.

In all cases you must pass into this function the column name which holds the code and one that holds the text you wish to display on the form dropdown. The column names are derived by the SAP search help and will not be known until the function is run for the first time. Therefore you can (and must) run first in test mode using transaction SE37 and then review the output table (EX_RETURN_VALUES) to identify the separate fieldname for the code and relevant text.

For a worked example we will review how the Ethnic Origin dropdown from infotype 0077 would be configured. The first input parameter is the screen field (IM_SCREEN_FIELD) and needs to be in the format TABLE-FIELD and can be determined by pressing F1 on your selected SAP field and navigation to the technical details tab.

Field Description for Batch Input	
Screen Field	P0077 - RACKY

For the first test pass we enter just this single import value and execute, leave others inputs blank or with default values. The function will return in table EX_RETURN_VALUES the set of

FLM for HR 4.20

Overview



fields and values from which the “key” and “text” fields can be selected for the final run and configuration.

Tables	Value
EX_RETURN_VALUES Result:	0 Entries 489 Entries

On our system there are 489 rows returned and on inspection these are in sets of three rows.

SHLPNAME	FIELDNAME	RECO	FIELDVAL
H_T505R	MOLGA	0001	07
H_T505R	RACKY	0001	01
H_T505R	LTEXT	0001	Aboriginal/Not Visible Minority
H_T505R	MOLGA	0002	07
H_T505R	RACKY	0002	02
H_T505R	LTEXT	0002	Visible Minority/Not Aboriginal
H_T505R	MOLGA	0003	07
H_T505R	RACKY	0003	03

For each country grouping the code (RACKY) and description (LTEXT) can be seen. Please note that the FLM for HR dropdown include will automatically filter on country grouping based on TXT_MOLGA and by language based on the current user logon language.

In cases where there is no suitable key you can provide RECORDPOS as the code and the row number for this entry will be used. Please note that in the event of the number of entries changing any historical forms may not display with the correct values.

Once you have identified the screen field, code and value parameters you can rerun the function and this time a final table of keys/descriptions are returned (EX_FORM_DATA). The setting of IM_CODEVAL will determine if the key is also shown in the description. Our example results based on GB are as follows.

Export parameters	Value	VALUE
EX_FORM_DATA EX_RETURN	14 Entries 	10-Asian Other 20-Bangladeshi 25-Black African 35-Black Caribbean 40-Black other 45-Indian 50-White Irish 55-Mixed race 60-Other 65-Pakistani 75-White Other 80-Not Disclosed 85-White British 90-Chinese

For some input fields existing screen values are taken into account to further restrict the values. For example the value of “Employee Group” is used to filter “Employee Subgroup” out of possible selections. This functionality is not supported by the simple function and an existing or new F4 function module will need to be developed.

If the dependant field “Employee Group” is pre-populated at the point of form render then custom functions can read the current form values and use this as a basis for making restricted selections for “Employee Subgroup”.

5.4 Complex F4 Dropdown (/FLMHR/HRF4_GET_ARRAY)

There are many cases where dropdown values are dependent on the value of other form fields where the value is not known until a user makes an initial selection. This complex dropdown, sometimes referred to as cascading, is used to support this functionality. Following the previous examples once a user chooses an employee group, the subgroup value would need to be cleared and any possible dropdown values refreshed.

To support this functionality the form needs

- to hold ALL the child subgroups and their relationship to the parent employee group
- some scripting added to the “exit” event on the template for the parent field e.g. Employee Group

An array should consist of a single string of values holding the parent key and its related child data (key and description) separated by tilde (~) with each record ending with a semicolon (;).

So, for employee group and subgroup we might expect a string as below.

```
EG A~ESG A1~Sub Group A1; EG A~ESG A2~Sub Group A2;EG B~ESG B0~Sub Group B0
```

This string can be broken back to indicate there are two possible employee groups, EGA and EGB. For EGA there are two subgroups, ESG A1 / ESG A2, and for EGB only one. On the template the delivered script object soDropdown and related methods (splitData, filterData and loadData) can be applied to the “exit” event of employee group.

Many arrays can be built based on the results of a simple F4 dropdown where the parent/child fields are returned in the EX_RETURN_VALUES table. Delivered function /FLMHR/HRF4_GET_ARRAY is used in a similar way to /FLMHR/HRF4_GET_SIMPLE but rather than passing code and value field a list of multiple fields is passed. The values from these fields are then concatenated together with a separator (~) and at the end of each RECORDPOS the delimiter (;) is added automatically. The order of the fields in the array string will match the field order in the record structure. Therefore a review of the soDropdown script parameters will be required to confirm the correct values are being used for both parent key and child key/values.

FLM for HR 4.20

Overview



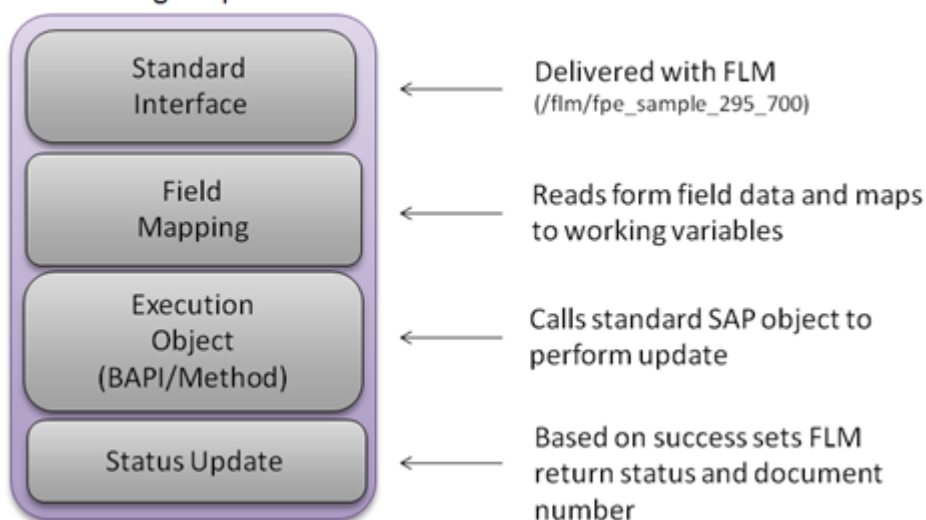
Note that for fields with large numbers of entries the size of the FLM form when opened will be increased and maybe a check-cycle or web service call should be considered as an alternative.



6.1 Posting Options

The FLM Form Posting Engine (FPE) configuration and processes are very flexible and powerful and the FLM for HR provides a number of additional adaptors and samples.

Structure of a Posting Adaptor



The delivered functions and samples all follow FLM best practices as below

- Share the standard interface.
- Reuse global field names and subform structures.
- Assume failure by setting EX_SUBRC to 4.
- Updates the FPE Log with information on progress to provide an audit trail.
- In the event of errors updated the FPE log to aid problem resolution and exit.
- Does not COMMIT WORK to ensure the FPE process retains control (BDC excluded).
- On success sets EX_SUBRC to 0 and the POSTED_DOC to the HR object e.g. Employee ID.

Depending on the version of FLM for HR and your HR Master Template the following delivered posting adaptors are provided for use.

Delivered Function	Description
/FLMHR/FPE_ACTION_PLUS_700	Create PA Action and Infotypes (No infogroup)
/FLMHR/FPE_SWAP_ACTION_700	Swap Action
/FLMHR/FPE_INFTOTYPE_700	Simple Infotype updates (PA/PD)

FLM for HR 4.20

Overview



The following sections cover in more detail when and how each can be used to update the SAP back end based on submitted forms.

6.2 Create PA Action / Infotypes (/FLMHR/FPE_ACTION_PLUS_700)

This adaptor, if required, will process an HR Action and associated infotypes based on the configured infogroup.

To use this adaptor you must provide every mandatory field for every expected infotype or the posting will fail. If the form data and business process do not allow all this data to be captured you have two options.

The first option is to copy the original HR action e.g. Z1 to create an FLM version e.g. ZF. This FLM version should have no infogroup assigned and be configured with the same “Reasons for Action” as the original. When configuring FPE the “swap action” adaptor can be used to revert back to the original action as is discussed later.

The second option is to use a custom BDC based adaptor to create just the Action then “exit” the transaction and in doing so skip further infotypes. A reduced set of infotypes can then be added to the action using this or the infotype adaptor described later.

This adaptor is designed to work off the Field level posting table /FLMHR/FLD_CNTRL where POST is set to A (Action) as in the example below.

Table Field	Description	Example Data
CCODE	FLM Customer Code	ACL
FTYPE	FLM Form Type	TS00
FVER	FLM Form Version	00
FIELD_NAME	FLM Form Field	DD_ACT_REASON
SEQ	Additional sequence (multiple keys)	< filled as required >
OTYPE	HR Object Type (for Object ID below)	P
OBJID	HR Object ID	TXT_IM_DOCUMENT
INFTY	HR Infotype number	0000
SUBTY	HR Infotype Subtype (optional)	< filled as required >
INFTY_FIELD	HR Infotype field	MASSG
DEF_VAL	A fixed default value	< filled as required >
NOTES	Free form notes to document entry	Action Reason
POST	FLM Posting Type	A (Infotype)
OPERATION	HR Infotype Operation	< always left blank >
DIFF_FIELD	FLM Form field holding old value	< always left blank >
SCREEN_FIELD	FLM Alternative Screen Field	< filled as required >

The above entry in the table would be processed for form type **TS00** version **00** and is set update the value of **MASSG** for Infotype **0000** for the Person **TXT_IM_DOCUMENT** to match the

FLM for HR 4.20

Overview



content of form field **DD_ACT_REASON**. Unlike the pre-population process the form type and version must be fully specified.

To actually complete an HR Action “Change Hours” the configuration could look as below.

FIELD_NAME	INFTY	SUBTY	INFTY_FIELD	NOTES	POST	OPERATION	SCREEN_FIELD
DD_ACT_REASON	0000		MASSG	Action Reason	A		
DD_EMPGROUP			PERSG	Employee Group	A		PSPAR-PERSG
DD_EMPSUBGROUP			PERSK	Employee Sub Group	A		PSPAR-PERSK
DD_PERSAREA			WERKS	Personnel Area	A		PSPAR-WERKS
TXT_POS_ID			PLANS	Position to hire onto	A		PSPAR-PLANS
CB_PART_TIME	0007		TEILK	Part Time indicator	A		
DD_TIME_MGMT_STATUS			ZTERF	Time Management Status	A		
DD_WORK_SCHED			SCHKZ	Work Schedule	A		
NUM_FTE			EMPCT	Employment Percentage	A		

Please note

- If there are no entries in the configuration table for the form type then the adaptor will complete successfully and pass control to the next configured adaptor, if one exists.
- The HR Action is read from the hidden subform SF_HRCONTROL and field TXT_ACTION.
- The SCREEN_FIELD column is used here in a special case to pass organisational data into to the action screen. These rows only need to be provided for new hire actions (but shown here for illustration purposes). If an entry for INFTY = 0000 / INFTY_FIELD = PERNR is passed then the current PERSG, PERSK, WERKS and PLANS values will be derived and do not need to be provided.

During the posting this function performs the following tasks.

- Reads the employee number passed via the form field linked to the 0000/PERNR entry.
- Reads the effect date (DT_BEGDA) from the form and fails if this is not found or is blank.
- Reads the HR Action (TXT_ACTION) from the form and fails if this is not found or is blank.
- Reads all Field Level posting configuration for the customer, form type, version with the POST value set to A.
- Determines the SAP field name for posting by assuming this is Pnnnn-ffff where nnnn = INFTY and ffff = INFTY_FIELD. If a value for SCREEN_FIELD is passed then this is used.
- Replaces Personnel Area/SubArea, Employee Group and Position from current employee record to support infotype 0000 creation (ignored if this is a Hire action).
- Calls standard SAP function HR_MAINTAIN_MASTERDATA processing an INSS operation for infotype 0000 only.
- Calls standard SAP function HR_MAINTAIN_MASTERDATA processing an INS operation for infotype 0001 only. This will ensure that a payroll status record (infotype 0003) is created as a basis for the next update.
- Calls standard SAP function HR_MAINTAIN_MASTERDATA processing an INS operation for all remaining infotypes other than 0000 and 0001.
- On failure (or an abort from foreground processing) the updates are stopped and SAP error messages fed back to the FPE Log

- On success the log is also updated and the next infotype update processed

Only once all expected updates are a success will they be committed to SAP and the next posting adaptor, if configured, be processed.

6.3 Swap Action (/FLMHR/FPE_SWAP_ACTION_700)

This adaptor, if required, should run directly after the Action posting and will replace the FLM specific action previously created with the equivalent customer action. This should be used in the case where no separate analysis of FLM Form updates is required such that all HR Actions can be reported together.

During the posting this function performs the following tasks.

- Reads the effect date (DT_BEGDA) from the form and fails if this is not found or is blank
- Reads the previous Posted Doc from FPE and fails if this is not found or is blank
- Read the entry from a single entry stored in the project variables under FPE_SWAP_ACTION. The “low” value should have the format of “oa/na” where “oa” = the old action and “na” the new action.
- Reads the current “old” action (just created) for the effective date
- Replaces the old action with a new action
- Modifies the previously read action and updates SAP

Only once the update succeeds will the next posting adaptor, if configured, be processed.

6.4 Simple Infotype updates and delimits (/FLMHR/FPE_INFTOYPE_700)

This adaptor is designed to work off the Field level posting table /FLMHR/FLD_CNTRL and performs simple field updates to existing SAP PA/PD records when no HR Action needs to be involved. Please note the following restrictions on its use.

- This adaptor will only update/delimit SAP records active on the effective date (DT_BEGDA) stored on the form. The Data pattern for this field (and all other date fields) must be YYYYMMDD in line with the SAP internal format. In cases where specific past or future dates should be used then a custom adaptor must be used.
- Where the FLM form or HR configuration could permit multiple records for an infotype to exist then there is a risk the incorrect record could be updated. Therefore additional unit testing should be performed or a custom adaptor used.

If there are no entries in the configuration table for the form type then the adaptor will complete successfully and pass control to the next configured adapter, if one exists.

Following on from the example used in pre-population the following table entries would be used to support the update of the last name or the employee.



Table Field	Description	Example Data
CCODE	FLM Customer Code	ACL
FTYPE	FLM Form Type	TS00
FVER	FLM Form Version	00
FIELD_NAME	FLM Form Field	TXT_EMP_LAST_NAME
SEQ	Additional sequence (multiple keys)	10
OTYPE	HR Object Type (for Object ID below)	P
OBJID	HR Object ID	TXT_IM_DOCUMENT
INFTY	HR Infotype number	0002
SUBTY	HR Infotype Subtype (optional)	< left blank >
INFTY_FIELD	HR Infotype field	NACHN
DEF_VAL	A fixed default value	< left blank >
NOTES	Free form notes to document entry	Employee Surname
POST	FLM Posting Type	I (Infotype)
OPERATION	HR Infotype Operation	< left blank >
DIFF_FIELD	FLM Form field holding old value	< left blank >
SCREEN_FIELD	FLM Alternative Screen Field	< left blank >

The above entry in the table would be processed for form type **TS00** version **00** and is set update the value of **NACHN** for Infotype **0002** for the Person **TXT_IM_DOCUMENT** to match the content of form field **TXT_LAST_NAME**. Unlike the pre-population process the form type and version must be fully specified.

Note that to support this adaptor an ABAP include (/FLMHR/ZCUST_INFTY_LIST) is provided and declares the structures of the most common infotypes likely to be updated. In the event, during development, that the posting tries to update an infotype not yet declared, the posting will fail, and the log will request that an update to this include is made.

During the posting this function performs the following tasks.

- Reads the effect date (DT_BEGDA) from the form and fails if this is not found is blank
- Read any existing Posted Doc, used with entries passing FORM_FIELD as POSTED_DOC
- Reads all Field Level posting configuration for the customer, form type, version with the POST value set to I.
- Sorts the configuration to ensure updates to many fields in a single infotype are performed once and at the same time
- Creates a dynamic table and work areas for the relevant infotype
- Reads the current record form PA/PD based on the effective date into the work area
 - o If the OPERATION is LIS9 (Delimit) and no data is found a message is placed in the log and the next infotype is processed.
 - o If the infotype structure contains the fieldname PERNR then PA update are performed otherwise a PD update be attempted.
- Replaces the effective date and field value(s) within a new work area

FLM for HR 4.20

Overview



- Performs an infotype operation via HR_CONTROL_INFOTYPE_OPERATION (or HR_INFOTYPE_OPERATION for older SAP releases) passing the new work area and operation specified.
- On failure (or an abort from foreground processing) the updates are stopped and SAP error messages fed back to the FPE Log
- On success the log is also updated and the next infotype update processed

Only once all configured updates are a success will they be committed to SAP and the next posting adaptor, if configured, be processed.



7.1 A simple example

The payroll department is streamlining the distribution of payslips to either electronic or via internal mail based on distribution from regional offices. The company needs managers to confirm or capture the county and email address for their existing employees. It is agreed that a new FLM form process will be implemented with the following requirements.

- There will be one form per employee
- The employees current country and email address will be displayed to allow the manager to review and update if required before submitting
- Only valid counties should be selected

FLM for HR components used will be

- HR Master template
 - o Source relevant subforms e.g. Header, action, effective date, employee information, address, and communications
- Form level pre-population
 - o Configure PA infotype reads for address (0006) and email (0105)
- Infotype Posting Adaptor
 - o Configure PA infotype updates for address and email

The new template and schema will be created and the FLM Form wizard run as normal. At this point the normal activity of coding large amounts of ABAP in the FLM user exits is replaced with adding a single line in the user exit and adding a few rows into FLM for HR configuration tables.

The step by step activity would be as follows

- Insert include into form level user exit
- Add entry for form type into /FLMHR/PRE_CNTRL to call Z_FLM_ACL_HR00_FPFE
- Add entries for form and field combinations for pre-population and posting
 - o DD_COUNTY
 - HR object of type P
 - Form Field TXT_IM_DOCUMENT (filled by framework from URL)
 - Infotype 0006 (Address)
 - Subtype 0001 (Permanent residence)
 - Field REGION (County)
 - Pre-population X
 - Posting (Infotype)

- TXT_EMP_EMAIL
 - HR object of type P
 - Form Field TXT_IM_DOCUMENT
 - Infotype 0105 (Communication)
 - Subtype 0010 (Email)
 - Field USRID_LONG (Email address)
 - Pre-population X
 - Posting (Infotype)
- Insert include into the county F4 user exit for DD_COUNTY
- Add entry for field DD_COUNTY based on delivered function Z_FLM_GET_REGION_F4
- Add entry to standard FLM FPE configuration to call HR for FLM infotype update adaptor /FLMHR/FPE_INFOTYPE_700

The form will now render, pre-populating data based on the employee chosen and passed via the URL when the form was launched. Changes made to the data will also be updated in SAP when the form is processed by FPE.

7.2 Adding infotype delimiting to the example

After seeing the development played back the payroll department realise that each employee currently has a “payslip” address infotype (1234). The move to internal mail or email makes this information no longer relevant and the process needs to change to also delimit any current record.

The step by step activity would be as follows

- Add an entry for form and field combinations for posting
 - DT_BEGDA
 - HR object of type P
 - Form Field TXT_IM_DOCUMENT (filled by framework from URL)
 - Infotype 0006 (Address)
 - Subtype 1234 (Payslip address)
 - Field ENDDA (End Date)
 - Posting (Infotype)
 - Operation LIS9 (Delimit)

We use the effective date field on the form to trigger the SAP data to delimit from that date. This approach can be used to delimit multiple infotypes in one updated (e.g. HR Leaver) and delimiting will occur for all current records (and skipped if none exist).

7.3 Adding an HR Action to the example

FLM for HR 4.20

Overview



For reporting purposes the HR department decide that employees that have been processed should have an HR action stored on infotype 0000. The action type will be ZA and no reason for action is configured.

The step by step activity would be as follows

- Add an entry for the form in customer variables to default action type (filled by framework as part of HR00 pre-population)
 - FPRE_TXT_ACTION
 - ZA
- Add an entry for form and field combinations for posting
 - TXT_ACTION
 - HR object of type P
 - Form Field TXT_IM_DOCUMENT (filled by framework from URL)
 - Infotype 0000 (Action)
 - Field MASSN (Action)
 - Posting (Infotype)



8 Appendix A – Popular Pre-populations

Employee Forms (TXT_IM_DOCUMENT = employee number passed in URL)

FIELD_NAME	OTYPE	OBJID	INFTY	SUBTY	INFTY_FIELD	NOTES
TXT_EMP_NAME	P	TXT_IM_DOCUMENT	0001		ENAME	Full Name
DD_BUSAREA	P	TXT_IM_DOCUMENT	0001		GSBER	Business Area
TXT_EMP_ORG_ID	P	TXT_IM_DOCUMENT	0001		ORGEH	Org ID
TXT_EMP_PER	P	TXT_IM_DOCUMENT	0001		PERNR	Employee Number
DD_EMPGRP	P	TXT_IM_DOCUMENT	0001		PERSG	Employee Group
DD_EMPSGRP	P	TXT_IM_DOCUMENT	0001		PERSK	Employee Subgroup
TXT_EMP_POS_ID	P	TXT_IM_DOCUMENT	0001		PLANS	Position ID
DD_PERSAREA	P	TXT_IM_DOCUMENT	0001		WERKS	Personnel area
DD_PERSUBAREA	P	TXT_IM_DOCUMENT	0001		BTRTL	Personnel Subarea
FIELD_NAME	OTYPE	OBJID	INFTY	SUBTY	INFTY_FIELD	NOTES
DD_TITLE	P	TXT_IM_DOCUMENT	0002		ANRED	Title
TXT_EMP_NAME	P	TXT_IM_DOCUMENT	0002		CNAME	Complete Name
DD_MARITAL_STAT	P	TXT_IM_DOCUMENT	0002		FAMST	Marital Status
DT_EMP_DOB	P	TXT_IM_DOCUMENT	0002		GBDAT	Date of Birth
DD_COB	P	TXT_IM_DOCUMENT	0002		GBLND	Country of Birth
DD_GENDER	P	TXT_IM_DOCUMENT	0002		GESCH	Gender
TXT_EMP_M_NAME	P	TXT_IM_DOCUMENT	0002		MIDNM	Middle Name
TXT_EMP_LAST_NAME	P	TXT_IM_DOCUMENT	0002		NACHN	Surname
DD_NATIONALITY	P	TXT_IM_DOCUMENT	0002		NATIO	Nationality
TXT_EMP_NATIONAL_ID	P	TXT_IM_DOCUMENT	0002		PERID	National ID

FLM for HR 4.20

Overview



TXT_EMP_KNOWN_AS	P	TXT_IM_DOCUMENT	0002		RUFNM	Known As
TXT_EMP_FIRST_NAME	P	TXT_IM_DOCUMENT	0002		VORNA	First Name
FIELD_NAME	OTYPE	OBJID	INFTY	SUBTY	INFTY_FIELD	NOTES
DD_COUNTRY	P	TXT_IM_DOCUMENT	0006		LAND1	Country
TXT_EMP_ADDR1	P	TXT_IM_DOCUMENT	0006		STRAS	Address line 1
TXT_EMP_ADDR	P	TXT_IM_DOCUMENT	0006		LOCAT	Address line 2
TXT_EMP_CITY	P	TXT_IM_DOCUMENT	0006		ORT01	City
TXT_EMP_POSTCODE	P	TXT_IM_DOCUMENT	0006		PSTLZ	Postcode
FIELD_NAME	OTYPE	OBJID	INFTY	SUBTY	INFTY_FIELD	NOTES
NUM_EMP_FTE_PERC	P	TXT_IM_DOCUMENT	0007		EMPCT	FTE %
DD_WORK_PATTERN	P	TXT_IM_DOCUMENT	0007		SCHKZ	Work Pattern
NUM_EMP_HRS_PW	P	TXT_IM_DOCUMENT	0007		WOSTD	Hours per Week
FIELD_NAME	OTYPE	OBJID	INFTY	SUBTY	INFTY_FIELD	NOTES
NUM_EMP_FTE_SALARY	P	TXT_IM_DOCUMENT	0008		ANSAL	Annual Salary
NUM_EMP_HRPAY_RATE	P	TXT_IM_DOCUMENT	0008		BET01	Hourly Rate
DD_PSCALE_TY	P	TXT_IM_DOCUMENT	0008		TRFAR	Payscale Type
DD_PSCALE_AREA	P	TXT_IM_DOCUMENT	0008		TRFGB	Payscale Area
DD_PSCALE_GRADE	P	TXT_IM_DOCUMENT	0008		TRFGR	Payscale Grade
DD_PSCALE_LEV	P	TXT_IM_DOCUMENT	0008		TRFST	Payscale Level
FIELD_NAME	OTYPE	OBJID	INFTY	SUBTY	INFTY_FIELD	NOTES
TXT_EMP_BNK_KEY	P	TXT_IM_DOCUMENT	0009		BANKL	Bank Key
NUM_EMP_BNK_ACCNO	P	TXT_IM_DOCUMENT	0009		BANKN	Bank Account Number
TXT_EMP_BNK_COUNTRY	P	TXT_IM_DOCUMENT	0009		BANKS	Bank Country Key
TXT_EMP_BNK_PAYMETH	P	TXT_IM_DOCUMENT	0009		ZLSCH	Payment Method
FIELD_NAME	OTYPE	OBJID	INFTY	SUBTY	INFTY_FIELD	NOTES
DT_START_DATE_AD	P	TXT_IM_DOCUMENT	0014		BEGDA	Payment Start date
NUM_AMOUNT_AD	P	TXT_IM_DOCUMENT	0014		BETRG	Payment Amount

FLM for HR 4.20

Overview



DT_END_DATE_AD	P	TXT_IM_DOCUMENT	0014		ENDDA	Payment End Date
DD_WAGE_TYPE_AD	P	TXT_IM_DOCUMENT	0014		LGART	Payment Wagetype
FIELD_NAME	OTYPE	OBJID	INFTY	SUBTY	INFTY_FIELD	NOTES
DT_EMP_CEND_0016	P	TXT_IM_DOCUMENT	0016		CTEDT	Contract End date
DD_CTYPE_0016	P	TXT_IM_DOCUMENT	0016		CTTYP	Contract type
DD_EE_NOTICE	P	TXT_IM_DOCUMENT	0016		KDGF2	Employee notice
DD_ER_NOTICE	P	TXT_IM_DOCUMENT	0016		KDGFR	Employer notice
DD_PROB_UNIT	P	TXT_IM_DOCUMENT	0016		PRBEH	Probation (Unit)
NUM_EMP_PROB_LENGTH	P	TXT_IM_DOCUMENT	0016		PRBZT	Probation Length

Position (TXT_IM_DOCUMENT = Position ID or use TXT_EMP_POS_ID from Employee)

FIELD_NAME	OTYPE	OBJID	INFTY	SUBTY	INFTY_FIELD	NOTES
TXT_POS_NAME	S	TXT_IM_DOCUMENT	1000		STEXT	Position Name
TXT_POS_ABRV	S	TXT_IM_DOCUMENT	1000		SHORT	Position Abbreviation
FIELD_NAME	OTYPE	OBJID	INFTY	SUBTY	INFTY_FIELD	NOTES
TXT_POS_JOB_ID	S	TXT_IM_DOCUMENT	1001	B007	SOBID	Job ID linked to Position
TXT_POS_ORG_ID	S	TXT_IM_DOCUMENT	1001	A003	SOBID	Org Unit ID linked to Position
TXT_POS_EMP_ID	S	TXT_IM_DOCUMENT	1001	A008	SOBID	Employee ID linked to Position
TXT_POS_CC_ID	S	TXT_IM_DOCUMENT	1001	A011	SOBID	Cost Centre linked to Position
DD_POS_QUALS	S	TXT_IM_DOCUMENT	1001	A031	SOBID	Qualification ID linked to Position
FIELD_NAME	OTYPE	OBJID	INFTY	SUBTY	INFTY_FIELD	NOTES
DD_POS_PSCALE_TY	S	TXT_IM_DOCUMENT	1005		TRFAR	Payscale Type
DD_POS_PSCALE_AREA	S	TXT_IM_DOCUMENT	1005		TRFGB	Payscale Area
DD_POS_PSCALE_GRADE	S	TXT_IM_DOCUMENT	1005		TRFG1	Payscale Grade
DD_POS_PSCALE_LEV	S	TXT_IM_DOCUMENT	1005		TRFS1	Payscale Level (from)

FLM for HR 4.20

Overview



DD_POS_PSCALE_LEV_TO	S	TXT_IM_DOCUMENT	1005		TRFS2	Payscale Level To
FIELD_NAME	OTYPE	OBJID	INFTY	SUBTY	INFTY_FIELD	NOTES
DD_POS_VAC_STAT	S	TXT_IM_DOCUMENT	1007		STAUS	Vacancy Status
FIELD_NAME	OTYPE	OBJID	INFTY	SUBTY	INFTY_FIELD	NOTES
DD_POS_PERSA	S	TXT_IM_DOCUMENT	1008		PERSA	Personnel area
DD_POS_PERSUBA	S	TXT_IM_DOCUMENT	1008		BTRTL	Personnel Subarea
DD_POS_BUSAREA	S	TXT_IM_DOCUMENT	1008		GSBER	Business Area
FIELD_NAME	OTYPE	OBJID	INFTY	SUBTY	INFTY_FIELD	NOTES
NUM_POS_HRS_PW	S	TXT_IM_DOCUMENT	1011		WKAVG	Standard Hours per Week
FIELD_NAME	OTYPE	OBJID	INFTY	SUBTY	INFTY_FIELD	NOTES
DD_POS_EGRP	S	TXT_IM_DOCUMENT	1013		PERSG	Employee Group
DD_POS_ESGRP	S	TXT_IM_DOCUMENT	1013		PERSK	Employee Subgroup

Org Unit (TXT_IM_DOCUMENT = Org ID or use TXT_EMP_ORG_ID from Employee)

FIELD_NAME	OTYPE	OBJID	INFTY	SUBTY	INFTY_FIELD	NOTES
TXT_ORG_NAME	O	TXT_IM_DOCUMENT	1000		STEXT	Org Unit Name
TXT_ORG_ABRV	O	TXT_IM_DOCUMENT	1000		SHORT	Org Unit Abbreviation
FIELD_NAME	OTYPE	OBJID	INFTY	SUBTY	INFTY_FIELD	NOTES
TXT_ORG_CHIEF_ID	O	TXT_IM_DOCUMENT	1001	B012	SOBID	Chief Position ID linked to Org Unit
TXT_ORG_HIGHER_ID	O	TXT_IM_DOCUMENT	1001	B002	SOBID	Higher Org Unit ID
TXT_ORG_CC_ID	O	TXT_IM_DOCUMENT	1001	B011	SOBID	Cost Centre ID linked to Org Unit
FIELD_NAME	OTYPE	OBJID	INFTY	SUBTY	INFTY_FIELD	NOTES
DD_ORG_PERSA	O	TXT_IM_DOCUMENT	1008		PERSA	Org Unit Personnel area
DD_ORG_PERSUBA	O	TXT_IM_DOCUMENT	1008		BTRTL	Org Unit Personnel Subarea



9 Appendix B – Popular Dropdowns

Entries based on SAP F4

FIELD_NAME	F4_SCREEN_FIELD	F4_CODE_FIELD	F4_VALUE_FIELD	NOTES
DD_CTYPE_0001	P0001-ANSVH	ANSVH	ATX	Contract Type (IT0001)
DD_BUSAREA	P0001-GSBER	GSBER	GTEXT	Business Area
DD_EMPGRP	P0001-PERSG	PERSG	PTEXT	Employee Group
DD_PERSAREA	P0001-WERKS	PERSA	NAME1	Personnel Area
DD_EMP_TITLE	P0002-ANRED	ANRED	ATEXT	Title (IT0002)
DD_MARITAL_STAT	P0002-FAMST	FAMST	FTEXT	Marital Status
DD_COB	P0002-GBLND	LAND1	LANDX	Country of Birth
DD_GENDER	P0002-GESCH	_LOW	_TEXT	Gender
DD_RELIGION	P0002-KONFE	KONFE	KTEXT	Religion
DD_NATIONALITY	P0002-NATIO	LAND1	NATIO	Nationality
DD_COUNTRY	P0006-LAND1	LAND1	LANDX	Country
DD_TMGMGT_STATUS	P0007-ZTERF	ZTERF	ZTEXT	Time Management Status
DD_BPAY_REASON	P0008-PREAS	PREAS	RTEXT	Reason for Basic Pay (Change)
DD_PSCALE_TY	P0008-TRFAR	TRFAR	TARTX	Pay Scale Type
DD_PSCALE_AREA	P0008-TRFGB	TRFGB	TGBTX	Pay Scale Area
DD_CTYPE_0016	P0016-CTTYP	CTTYP	CTTXT	Contract Type (IT0016)
DD_EE_NOTICE	P0016-KDGF2	KDGFR	KTEXT	Employee Notice Period
DD_ER_NOTICE	P0016-KDGFR	KDGFR	KTEXT	Employer Notice Period
DD_CDETA_0016	P0016-KONSL	KOSL1	KON	Contract Detail
DD_PROB_UNIT	P0016-PRBEH	ZEINH	ETEXT	Probation Units

FLM for HR 4.20

Overview



DD_CORP_FUNC	P0034-FUNKT	SUBTY	STEXT	Corporate Function
DD_OOL_ITEM	P0040-LEIHG	SUBTY	STEXT	Object on Loan item
DD_LOAN_TYPE	P0045-DLART	SUBTY	STEXT	Loan Type
DD_P46_STATE	P0065-P46CLS	_LOW	_TEXT	GB Tax - P46 Statement
DD_TAX_SOURCE	P0065-TDSRC	_LOW	_TEXT	GB Tax – Tax Code Source
DD_ETHNIC_ORIGIN	P0077-RACKY	RACKY	LTEXT	Ethnic Origin
DD_POS_VAC_STATUS	HRP1007-STATUS	_LOW	_TEXT	Position Vacancy Status

Entries based on delivered Function Modules

FIELD_NAME	F4_SCREEN_FIELD	NOTES
DD_ACT_REASON	/FLMHR/HRF4_GET_ACTION_REASON	Reads hidden field TXT_ACTION to filter correct reasons
DD_WAGE_TYPE_BPAY	/FLMHR/HRF4_GET_WAGETYPES_IT	Reads entry in project variables table (field name with _IT suffix) to provide infotype number (0008) for automatic filtering.
DD_WAGE_TYPE_AD	/FLMHR/HRF4_GET_WAGETYPES_IT	Reads entry in project variables table (field name with _IT suffix) to provide infotype number (0014) for automatic filtering.
DD_WAGE_TYPE_AP	/FLMHR/HRF4_GET_WAGETYPES_IT	Reads entry in project variables table (field name with _IT suffix) to provide infotype number (0015) for automatic filtering.

Entries based on available Function Modules (request sample code via support@arch.co.uk)

FIELD_NAME	F4_SCREEN_FIELD	NOTES
DD_EMPSGRP	Z_FLM_GET_EMPSUBGROUP_F4	Reads hidden field DD_EMPGROUP to filter correct reasons
DD_PERSUBAREA	Z_FLM_GET_PERSUBAREA_F4	Reads hidden field DD_PERSAREA to filter correct sub areas
DD_POSITION	Z_FLM_GET_POSITION_F4	Reads field DD_ORG_UNIT to filter positions in and below this org
DD_REGION	Z_FLM_GET_REGION_F4	Reads field DD_COUNTRY to filter correct regions
DD_POS_REP_TO	Z_FLM_GET_REP_TO_F4	Uses evaluation paths PERSCM2, O-O_DOWN and BOSSONLY