

Calculation of Deadlines Using the Factory Calendar.

(Courtesy of Helena Sveniker, SAP AG)

■ No part of this presentation may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

■ Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

■ Microsoft®, WINDOWS®, NT®, EXCEL®, Word® and SQL Server® are registered trademarks of Microsoft Corporation.

■ IBM®, DB2®, OS/2®, DB2/6000®, Parallel Sysplex®, MVS/ESA®, RS/6000®, AIX®, S/390®, AS/400®, OS/390®, and OS/400® are registered trademarks of IBM Corporation.

■ ORACLE® is a registered trademark of ORACLE Corporation, California, USA.

■ INFORMIX®-OnLine for SAP is a registered trademark of Informix Software Incorporated.

■ UNIX®, X/Open®, OSF/1®, and Motif® are registered trademarks of The Open Group.

■ HTML, DHTML, XML, XHTML are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Laboratory for Computer Science NE43-358, Massachusetts Institute of Technology, 545 Technology Square, Cambridge, MA 02139.

■ JAVA® is a registered trademark of Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, CA 94303 USA.

■ JAVASCRIPT® is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

■ SAP, SAP Logo, mySAP.com, mySAP.com Marketplace, mySAP.com Workplace, mySAP.com Business Scenarios, mySAP.com Application Hosting, WebFlow, R/2, R/3, RIVA, ABAP, SAP Business Workflow, SAP EarlyWatch, SAP ArchiveLink, BAPI, SAPPHIRE, Management Cockpit, SEM, are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other products mentioned are trademarks or registered trademarks of their respective companies.

1 Aim There are two algorithms for including the factory calendar in your workflow deadlines. Either the deadline is calculated as an attribute, which is better from a performance view because no additional RFC call must be made, or the deadline is calculated using a background method, which makes everything more flexible, as parameters can be transferred here, but which is not ideal from a performance view.

2 Prerequisites

A sub type must be created for the object type to be used.

3 Method Implementation

A new method must be created with the following parameters:

Method: Get_deadline deadline_determine

3.1 Description of the "GET_DEADLINE" method

The relevant workday is determined using the transferred parameter. The following parameters can or must be transferred:

Date_from Date from which a deadline should be calculated
Factorycalendarid ID for the factory calendar to be used
Offset Number of days that are to be added to the Date_from
Correction Specifies if the first day before or after the work day should be outputted if the date does not fall on a work day.

Deadline Contains the calculated deadline

3.2 Function modules used

DATE_CONVERT_TO_FACTORYDATE
 FACTORYDATE_CONVERT_TO_DATE

3.3 Installation of the "GET_DEADLINE" method

Use of the "GET_DEADLINE" method in a single-step task that is as a background task, the "GET_DEADLINE" method is defined for the "BUS1001" object type. You should answer yes to the questions after the transfer of the missing container elements.

You then have to ensure that the parameter named above is provided. This can either take place using a binding or initial value within the single-step task or be managed using a binding from the workflow container to the task container. Constants can also be entered here.

3.4 Method Code

This method determines a deadline suitable for the factory calendar using the parameter transferred.

3.4.1 Mandatory parameters

Element name	Name	Description	Reference
DATE_FROM	from_date	The deadline should be calculated from this deadline	SCAL-DATE
FACTORYCALENDARID	Factory calendar ID	Abbreviation for the factory calendar to be used	SCAL-FCALID

3.4.2 Optional parameters

Element name	Name	Description	Reference
Correction	Correction name	Possible values are: '+' if the date entered is not a workday, the first workday after the date is delivered (default). '-' if the date entered is not a workday, the first workday before the date is delivered.	SCAL-INDICATOR
Offset	Offset	Specifies how many days for the from_date have to be calculated. If no offset has been specified, the factory calendar date belonging to Date_from is simply determined.	SWD_SDYNP-LATE_END_O

3.4.3 Return parameters

Element name	Name	Description	Reference
Deadline	Deadline_factorycalendar	The deadline determined is returned in this element	SCAL-DATE

3.5 Coding method

```
BEGIN_METHOD GET_DEADLINE CHANGING CONTAINER.  
  DATA: FACTORYCALENDARID LIKE SCAL-FCALID,  
        DATE_FROM LIKE SCAL-DATE,  
        FACTORYDATE LIKE SCAL-FACDATE,  
        CORRECTION LIKE SCAL-INDICATOR,  
        DEADLINE LIKE SCAL-DATE,  
        OFFSET TYPE I.
```

```
SWC_GET_ELEMENT CONTAINER 'FactoryCalendarID' FACTORYCALENDARID.  
SWC_GET_ELEMENT CONTAINER 'Date_from' DATE_FROM.  
SWC_GET_ELEMENT CONTAINER 'correction' CORRECTION.  
SWC_GET_ELEMENT CONTAINER 'Offset' OFFSET.
```

```
*Calculate offset for transferred deadline  
IF NOT OFFSET IS INITIAL.  
  DATE_FROM = DATE_FROM + OFFSET.  
ENDIF.
```

```
*Set default value for the correction if it has not been transferred  
IF CORRECTION IS INITIAL.  
  CORRECTION = '+' .  
ENDIF.
```

```
*Determine factory calendar date for the date transferred  
CALL FUNCTION 'DATE_CONVERT_TO_FACTORYDATE'  
EXPORTING  
  CORRECT_OPTION = CORRECTION  
  DATE = DATE_FROM  
  FACTORY_CALENDAR_ID = FACTORYCALENDARID
```

```

IMPORTING
*      DATE =
FACTORYDATE = FACTORYDATE
*      WORKINGDAY_INDICATOR =
EXCEPTIONS
CALENDAR_BUFFER_NOT_LOADABLE = 1
CORRECT_OPTION_INVALID = 2
DATE_AFTER_RANGE = 3
DATE_BEFORE_RANGE = 4
DATE_INVALID = 5
FACTORY_CALENDAR_NOT_FOUND = 6
OTHERS = 7.

*Convert factory date to a normal calendar date
CALL FUNCTION 'FACTORYDATE_CONVERT_TO_DATE'
EXPORTING
FACTORYDATE = FACTORYDATE
FACTORY_CALENDAR_ID = FACTORYCALENDARID
IMPORTING
DATE = DEADLINE
EXCEPTIONS
CALENDAR_BUFFER_NOT_LOADABLE = 1
FACTORYDATE_AFTER_RANGE = 2
FACTORYDATE_BEFORE_RANGE = 3
FACTORYDATE_INVALID = 4
FACTORY_CALENDAR_ID_MISSING = 5
FACTORY_CALENDAR_NOT_FOUND = 6
OTHERS = 7.

*Put result in the 'deadline' parameter
SWC_SET_ELEMENT_CONTAINER 'Deadline' DEADLINE.

END_METHOD.

```

4 Virtual Attribute Implementation

Create virtual attribute 'DEADLINE_2DAYS' with reference to SYST-DATE.

4.1 Description of the virtual attributes

Four virtual attributes are determined. These return deadlines taking into account the Swiss factory calendar. The deadlines refer to the creation time of the BUS10001 object (material or article). A deadline for 2, 4, 6 and 8 days after creation of the object is calculated for each one.

2, 4, 6 or 8 days are added to the date from the system date. The "DATE_CONVERT_TO_FACTORYDATE" function module is called with this new date. This returns the next work day in the factory calendar format for the date transferred.

The date returned is converted to a normal calendar date using the "FACTORYDATE_CONVERT_TO_DATE" function module.

The new factory calendar date is placed in the relevant virtual attribute of the BUS1001 object.

4.2 Coding of the virtual attributes

```

GET_PROPERTY DEADLINE_2DAYS CHANGING CONTAINER.
  DATA: DEADLINE LIKE SCAL-DATE, "Factory calendar deadline
         DATE LIKE SCAL-DATE, "Outbound date

```

```

FACTORYDATE LIKE SCAL-FACDATE,          "Factory calendar date

* Factory calendar ID
FACTORY_CALENDAR_ID LIKE SCAL-FCALID VALUE 'CH',

* '+' If the date entered is not a workday, the first workday
*   after the date is returned.
CORRECTION LIKE SCAL-INDICATOR VALUE '+'.

CONSTANTS: D2 VALUE '2',
            D4 VALUE '4',
            D6 VALUE '6',
            D8 VALUE '8',
            D10 VALUE '10'.

*swc_get_element container '_WF_Start_Date' date.
* Determine date +2 days
*date = date + d2.
DATE = SY-DATUM + D2.

*Determine factory calendar date
CALL FUNCTION 'DATE_CONVERT_TO_FACTORYDATE'
EXPORTING
CORRECT_OPTION          = '+'
DATE                    = DATE
FACTORY_CALENDAR_ID    = FACTORY_CALENDAR_ID
IMPORTING
*      DATE              =
FACTORYDATE             = FACTORYDATE
*      WORKINGDAY_INDICATOR =
EXCEPTIONS
CALENDAR_BUFFER_NOT_LOADABLE = 1
CORRECT_OPTION_INVALID       = 2
DATE_AFTER_RANGE             = 3
DATE_BEFORE_RANGE            = 4
DATE_INVALID                 = 5
FACTORY_CALENDAR_NOT_FOUND   = 6
OTHERS                       = 7.

* The factory calendar date determined is converted to a normal
calendar date again
CALL FUNCTION 'FACTORYDATE_CONVERT_TO_DATE'
EXPORTING
FACTORYDATE             = FACTORYDATE
FACTORY_CALENDAR_ID    = FACTORY_CALENDAR_ID
IMPORTING
DATE                    = DEADLINE
EXCEPTIONS
CALENDAR_BUFFER_NOT_LOADABLE = 1
FACTORYDATE_AFTER_RANGE      = 2
FACTORYDATE_BEFORE_RANGE     = 3
FACTORYDATE_INVALID         = 4
FACTORY_CALENDAR_ID_MISSING  = 5
FACTORY_CALENDAR_NOT_FOUND   = 6
OTHERS                      = 7.

IF SY-SUBRC = 0.
OBJECT-DEADLINE_2DAYS = DEADLINE.
SWC_SET_PROPERTY SELF 'Deadline_2Days' DEADLINE.
ENDIF.                                "endif sy-subrc = 0.

```

END_PROPERTY.

End of document