Oracle Forms, Microsoft and more

Gerd Volberg
OPITZ CONSULTING Deutschland GmbH
Gummersbach

Schlüsselworte
Oracle Forms, Microsoft, Exchange-Server, Emails, Jira, Issue-Tracking, Sharepoint, Outlook, Confluence, Wiki

Einleitung

Moderne Forms-Applikationen sind am produktivsten, wenn sie die gesamte IT-Infrastruktur unterstützen.
Im Folgenden werden Sourcecodes vorgestellt, die zur EMail-Erzeugung und Versendung von CSV-Dateien genutzt werden können, die Issue-Tracking- und Wiki-Systeme unterstützen und Einträge im Kalender vornehmen.

EMails erzeugen und versenden

Die Grundlage für die Parameterübergabe an Subsysteme sind häufig EMails. Somit ist die erste und wichtigste Prozedur, die man in diesem Kontext benötigt eine EMail-Senden-Routine.
Diese kann fortan für automatische Benachrichtigungen der Forms-Applikation an den User dienen und auch zur Steuerung von Kalendern, Issue-Tracking-Systemen, o.v.m.

Die Prozedur Sende_EMail reicht aus, um in allen folgenden Beispielen eine ausreichend umfangreiche Hilfsroutine zu haben.

```
PROCEDURE SENDE_EMAIL
 (P_VON       IN VARCHAR2
 ,P_VON_NAME  IN VARCHAR2 := NULL
 ,P_AN        IN VARCHAR2
 ,P_AN_NAME   IN VARCHAR2 := NULL
 ,P_CC        IN VARCHAR2 := NULL
 ,P_CC_NAME   IN VARCHAR2 := NULL
 ,P_BCC       IN VARCHAR2 := NULL
 ,P_BETREFF   IN VARCHAR2 := '(Betreff ist leer)'
 ,P_NACHRICHT IN VARCHAR2 := '(Nachricht ist leer)'
 ,P_EMAIL_TYP IN VARCHAR2 :=
   'Content-Type: text/plain; charset="ISO-8859-1"'
 ) IS
  V_Host       VARCHAR2 (100);
  V_Connection utl_smtp.connection;

  PROCEDURE Schreibe_Daten (P_To_From_Str IN VARCHAR2,
                            P_Name        IN VARCHAR2,
                            P_EMail       IN VARCHAR2) IS
  BEGIN
    IF  P_Name  IS NOT NULL
    AND P_EMail IS NOT NULL THEN
      utl_smtp.write_data (V_Connection, P_To_From_Str || ': ' ||
        P_Name  || ' <' || P_EMail || '>' || CHR (10));
    ELSIF P_EMail IS NOT NULL THEN
      utl_smtp.write_data (V_Connection, P_To_From_Str || ': ' ||
                           P_EMail ||        CHR (10));
    END IF;
```

```
  END;

  PROCEDURE Schreibe_Adressat (P_EMail IN VARCHAR2) IS
  BEGIN
    IF P_EMail IS NOT NULL THEN
      utl_smtp.rcpt (V_Connection, P_EMail);
    END IF;
  END;
BEGIN
  V_Host := 'mailrelay.oc.de';
  V_Connection := utl_smtp.open_connection (V_Host, 25);
  utl_smtp.helo (V_Connection, V_Host);
  utl_smtp.mail (V_Connection, P_Von);

  Schreibe_Adressat (P_An);
  Schreibe_Adressat (P_CC);
  Schreibe_Adressat (P_BCC);

  utl_smtp.open_data (V_Connection);

  Schreibe_Daten ('From', P_Von_Name, P_Von);
  Schreibe_Daten ('To',   P_An_Name,  P_An);
  Schreibe_Daten ('CC',   P_CC_Name,  P_CC);

  utl_smtp.write_data (V_Connection, 'Subject: ' || P_Betreff || CHR (10));
  utl_smtp.write_data (V_Connection, P_Email_Typ || CHR (10) || CHR (10));
  utl_smtp.write_data (V_Connection, P_Nachricht);
  utl_smtp.close_data (V_Connection);
  utl_smtp.quit       (V_Connection);
END;
```

Ein Aufruf dieser Prozedur zum Verschicken einer einfachen EMail könnte zum Beispiel so aussehen:

```
Sende_EMail (P_VON       => 'formsapp@oc.de'
            ,P_VON_NAME  => 'Forms-Applikation'
            ,P_AN        => 'gerd.volberg@oc.de'
            ,P_AN_NAME   => 'Gerd Volberg'
            ,P_BETREFF   => 'Test-EMail'
            ,P_NACHRICHT => 'Hello World');
```

Kalendereinträge erzeugen

Neben dem reinen Versenden von EMails werden automatisch erzeugte Kalendereinträge immer wichtiger.

Dieses Feature lässt sich ebenfalls sehr einfach in Forms integrieren. In diesem Beispiel arbeitet im Hintergrund ein MS-Exchange-Server. Die User arbeiten mit MS-Outlook.

```
PROCEDURE SEND_VCALENDAR
 (P_DATUM_START    IN DATE
 ,P_DATUM_ENDE     IN DATE     := NULL
 ,P_EMAIL_AN       IN VARCHAR2 := NULL
 ,P_BETREFF        IN VARCHAR2
```

```
 ,P_ORT            IN VARCHAR2 := NULL
 ,P_BESCHREIBUNG   IN VARCHAR2 := NULL
 ,P_GANZTAEGIG_KNZ IN VARCHAR2 := 'J'
 ) IS
  V_DATUM_ENDE    DATE;
  V_CLOB          CLOB;

  FUNCTION Format_Date (P_Datum IN DATE) RETURN VARCHAR2 IS
    -- Timezone-Korrektur
    V_DELTA_STUNDEN NUMBER(2) := 2;
  BEGIN
    IF P_GANZTAEGIG_KNZ = 'J' THEN
      RETURN (';VALUE=DATE:' || TO_CHAR (P_Datum, 'YYYYMMDD'));
    ELSE
      RETURN (':' || TO_CHAR (P_Datum-V_Delta_Stunden/24, 'YYYYMMDD') ||
              'T' || TO_CHAR (P_Datum-V_Delta_Stunden/24, 'HH24MISS') ||
              'Z');
    END IF;
  END;
BEGIN
  IF P_Datum_Ende IS NULL THEN
    V_Datum_Ende := P_Datum_Start;
  ELSIF P_Datum_Start = TRUNC (P_Datum_Start)
  AND   P_Datum_Ende  = TRUNC (P_Datum_Ende)
  AND   P_Datum_Ende > P_Datum_Start THEN
    -- Ein Zeitraum wird jeweils von 0 Uhr bis 0 Uhr gespeichert.
    -- Deswegen muss das BIS-Datum um 1 Tag vergrössert werden.
    V_Datum_Ende := P_Datum_Ende + 1;
  END IF;

  V_CLOB := 'BEGIN:VCALENDAR'                              || CHR (10) ||
            'VERSION:2.0'                                  || CHR (10) ||
            'BEGIN:VEVENT'                                 || CHR (10) ||
            'DESCRIPTION:' || P_Beschreibung               || CHR (10) ||
            'DTEND'        || Format_Date (V_Datum_Ende)   || CHR (10) ||
            'DTSTART'      || Format_Date (P_Datum_Start)  || CHR (10) ||
            'LOCATION:'    || P_Ort                        || CHR (10) ||
            'SUMMARY:'     || P_Betreff                    || CHR (10) ||
            'TRANSP:OPAQUE'                                || CHR (10) ||
            'X-MICROSOFT-CDO-BUSYSTATUS:OOF'               || CHR (10) ||
            'END:VEVENT'                                   || CHR (10) ||
            'END:VCALENDAR';

  Sende_EMail (
    P_VON       => 'formsapp@oc.de',
    P_AN        => 'exchange@oc.de',
    P_Betreff   => 'Kalendereintrag: ' || TO_CHAR (P_Datum_Start,
                   'DD.MM.YYYY') || ' ' || P_Betreff,
    P_Nachricht => V_CLOB);
END;
```

Ein Kalendereintrag könnte nun mit gerade einmal vier Parametern erzeugt werden:

```
SEND_VCALENDAR (P_DATUM_START    => to_date ('27.11.2013', 'DD.MM.YYYY')
               ,P_DATUM_ENDE     => to_date ('27.11.2013', 'DD.MM.YYYY')
               ,P_EMAIL_AN       => 'gerd.volberg@oc.de'
```

```
                ,P_BETREFF        => 'Urlaub');
```

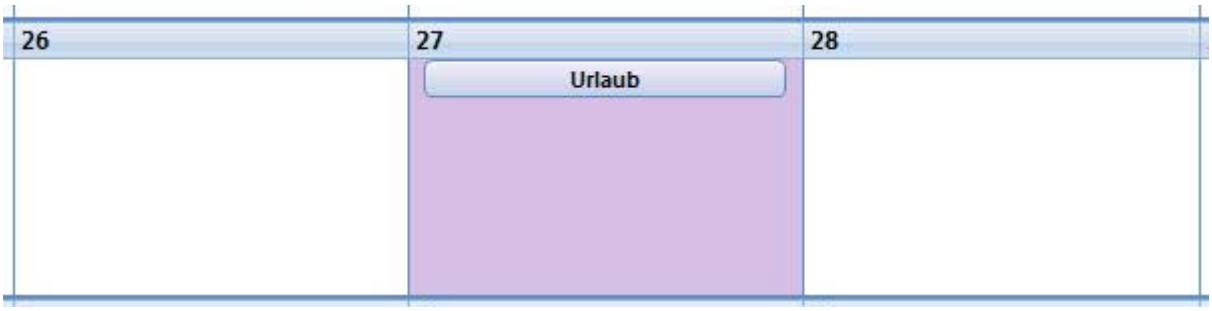

Abb. 1: Automatisch erzeugter Kalendereintrag

Wiki-Systeme integrieren

Applikationen, die dem User auf jeder Formsmaske eine eigene Hilfeseite anbieten möchten, arbeiten am effektivsten mit einem Wikisystem zusammen.
In diesem Beispiel wird Confluence benutzt, dass sich im Microsoft-Umfeld immer grösserer Beliebtheit erfreut.

Damit jede Maske eine genau definierte Wikiseite aufrufen kann, benötigen wir ein kleines Datenmodell zur Steuerung der Metainformationen.

Wir speichern alle Formsmasken in einer Tabelle namens FORMS.

FORM_ID	1	1	N	NUMBER (9)		PK
FORM_FILE_NAME	2		N	VARCHAR2 (14 Byte)		Phys. Name der Datei
FORM_NAME	3		N	VARCHAR2 (30 Byte)		Name der Maske

Abb. 2: Tabelle FORMS

Alle Wikiseiten werden in der Tabelle WIKIS gespeichert.

Column Name	ID	Pk	Null?	Data Type	Default	Comments
WIKI_ID	1	1	N	NUMBER (9)		PK
WIKI_FORM_ID	2		N	NUMBER (9)		FK auf Forms
WIKI_ITEM_NAME	3		Y	VARCHAR2 (100 Byte)		Itemname (wenn NULL, dann gilt's für die ganze Maske)
WIKI_OFFSET	4		Y	VARCHAR2 (100 Byte)		URL-Offset, bezogen auf die Start-Wiki-Seite

Abb. 3: Tabelle WIKIS

In unserem Beispiel sind alle Wikiseiten unter folgender URL erreichbar:
`https://confluence.oc.de/display/Formsapp/<Name-der-Wiki-Seite>`

Der statische Teil dieser URL sollte an einer zentrale Stelle als Konstante definiert sein, so dass ein Zugriff von allen Applikationsteilen möglichst einfach gewährleistet werden kann. Z.B.
`C_Startseite_Wiki := 'https://confluence.oc.de/display/Formsapp/'`

In allen Formsmasken in denen man eine Hilfeseite anbieten möchte, kann man nun Wiki-Buttons hinterlegen, wie z.B. folgenden:

Abb. 4: Wiki-Button

Der Button benötigt einen WHEN-BUTTON-PRESSED-Trigger mit dem Aufruf der zentralen Wiki-Routine namens Starte_Wiki.

```
Trigger WHEN-BUTTON-PRESSED:
Starte_Wiki;
```

Die Routine Starte_Wiki ermittelt auf Basis des Buttonnamens und des aktuellen Maskennamens einen Offset, der zusammen mit der Wikistartseite eine URL ergibt, die dann aus Forms heraus gestartet wird.

```
PROCEDURE Starte_Wiki IS
  V_FORM_Name  VARCHAR2 (30);
  V_ITEM_Name  VARCHAR2 (61);
  V_Offset     WIKIS.WIKI_Offset%TYPE;
BEGIN
  V_FORM_Name := NAME_IN ('System.Current_Form');
  V_ITEM_Name := NAME_IN ('System.Trigger_Item');

  SELECT WIKI_Offset
    INTO V_Offset
    FROM Wikis, Forms
   WHERE WIKI_FORM_ID   = FORM_ID
     AND FORM_NAME      = V_FORM_Name;
     AND WIKI_Item_Name = V_ITEM_Name;

  WEB.Show_Document (C_Startseite_Wiki || V_Offset);
END;
```

WIKI_ID	WIKI_FORM_ID	WIKI_ITEM_NAME	WIKI_OFFSET
5	54	WIKI.BT_PROJEKT_BESCHREIBUNG	Projektmaske+Beschreibung
6	54	WIKI.BT_PROJEKT_CONTROLLING	Projektmaske+Controlling
7	54	WIKI.BT_PROJEKT_MITARBEITER	Projektmaske+Projektmitarbeiter
8	54	WIKI.BT_PROJEKT_AKTIVITAETEN	Projektmaske+Aktivitäten
9	54	WIKI.BT_PROJEKT_ARBEITSPAKETE	Projektmaske+Arbeitspakete
10	54	WIKI.BT_PROJEKT_BERICHTE	Projektmaske+Projektberichte
11	54	WIKI.BT_PROJEKT_HISTORIE	Projektmaske+Historie
12	54	WIKI.BT_PROJEKT_UNTERPROJEKTE	Projektmaske+Unterprojekte
13	52	WIKI.BT_AKTIONEN_KOPIERE_AKTION	Aktionenmaske+Kopiere+Aktion
14	52	WIKI.BT_AKTIONEN_AP_ZUORDNEN	Aktionenmaske+AP+zuordnen

Abb. 5: Exemplarische Stammdaten der Tabelle Wikis

Wird nun in der Maske mit der ID 54 der Wiki-Button WIKI.BT_Projekt_Beschreibung betätigt, startet sofort ein neues Browserfenster mit der Wikiseite „Projektmaske Beschreibung“

Abb. 6: Wikiseite mit URL

Issue-Tracking anbinden

Applikationen, die eine Anbindung an das firmeninterne Issue-Tracking anbieten, erleichtern ihren Usern die Erzeugung und Verfolgung von Problemfällen und Bugs. Die folgende Prozedur erzeugt beispielsweise Jira-Tickets.

```
PROCEDURE Create_Jira_Ticket
 (P_VON              IN VARCHAR2
 ,P_VON_NAME         IN VARCHAR2 := NULL
 ,P_BETREFF          IN VARCHAR2
 ,P_PROJEKT          IN VARCHAR2
 ,P_VORGANGSART      IN VARCHAR2 := 'Anfrage'
 ,P_PRIO             IN VARCHAR2 := 'Normal'
 ,P_KOMPONENTE       IN VARCHAR2 := NULL
 ,P_SICHERHEITSSTUFE IN VARCHAR2 := NULL
 ,P_TEXT             IN VARCHAR2 ) IS

  V_Komponente       VARCHAR2(100);
  V_SICHERHEITSSTUFE VARCHAR2(100);
BEGIN
  IF  P_Von         IS NOT NULL
  AND P_Betreff     IS NOT NULL
  AND P_Projekt     IS NOT NULL
  AND P_Vorgangsart IS NOT NULL
  AND P_Prio        IS NOT NULL
  AND P_Text        IS NOT NULL THEN
    IF P_Komponente IS NOT NULL THEN
      V_Komponente := '@Komponente = ' ||
                      P_Komponente || CHR (10);
    END IF;
    IF P_Sicherheitsstufe IS NOT NULL THEN
      V_Sicherheitsstufe := '@Sicherheitsstufe = ' ||
                            P_Sicherheitsstufe || CHR (10);
    END IF;
    Sende_EMail (
      P_Von       => P_Von,
      P_Von_Name  => P_Von_Name,
      P_An        => 'jira@oc.de',
      P_An_Name   => 'Jira',
      P_Betreff   => P_Betreff,
      P_Nachricht => '@Projekt = '      || P_Projekt     || CHR (10) ||
                     '@Vorgangsart = '  || P_Vorgangsart || CHR (10) ||
                     '@Prio = '         || P_Prio        || CHR (10) ||
                     V_Komponente       ||
```

```
                    V_Sicherheitsstufe ||
                    P_Text);
  END IF;
END;
```

Diese Routine kann nun innerhalb der Applikation jederzeit mit wenigen Parametern aufgerufen werden:

```
Create_Jira_Ticket (P_Von         => 'gerd.volberg@oc.de',
                    P_Betreff     => 'Automatisch erzeugtes Ticket',
                    P_Projekt     => 'YAPUT',
                    P_Text        => 'Aus Forms erzeugtes Ticket.');
```

Ein Ticket, das vom User in der Jira-Oberfläche erzeugt würde, sähe so aus:

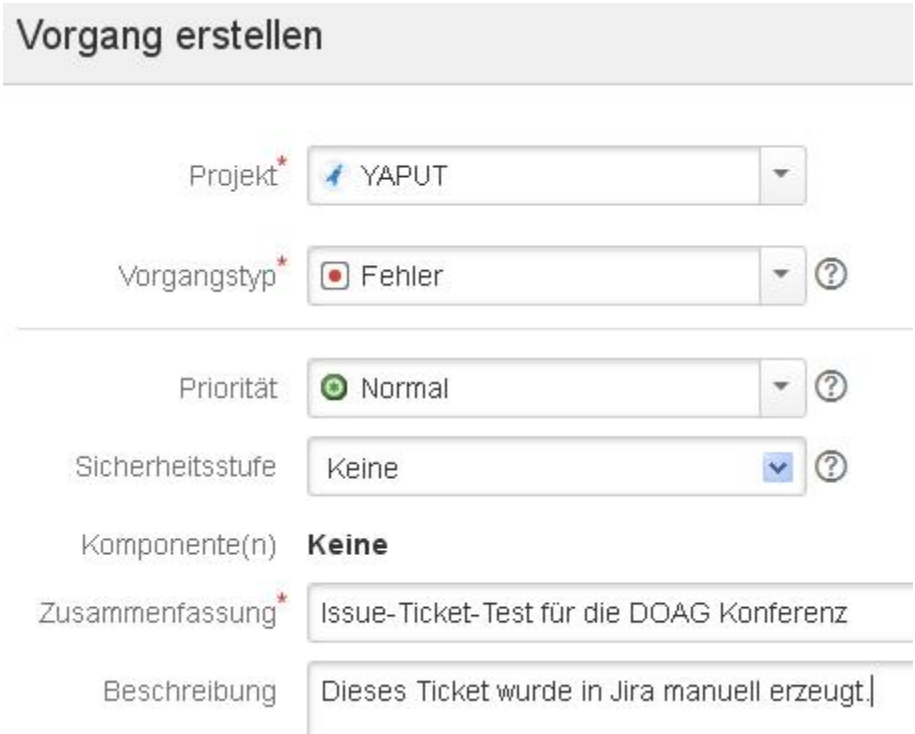

Abb. 7: Manuell erzeugtes Jira-Ticket

In Jira sind die manuell und automatisch erzeugten Tickets gleichwertig untereinander angezeigt.

Schlüssel	Zusammenfassung	Bearbeiter
YAPUT-3	Automatisch erzeugtes Ticket	*Nicht zugewiesen*
YAPUT-2	Issue-Ticket-Test für die DOAG Konferenz	*Nicht zugewiesen*

Abb. 8: Übersicht aller Jira-Tickets

Kontaktadresse:
Gerd Volberg
OPITZ CONSULTING GmbH
Kirchstr. 6

D-51647 Gummersbach

Telefon: +49 (0) 2261-6001 0
Fax: +49 (0) 2261-6001 4200
E-Mail: gerd.volberg@opitz-consulting.com
Blog: http://talk2gerd.blogspot.com