

Outside In and Inside Out— Data from and to Outside the Database

John Flack

Synectics for Management Decisions,
Inc.

Introduction

- Data in our database originates from outside of the database
- Data from our database is destined outside of the database
- How can we get data from:
 - The file system
 - E-mail
 - Other computersInto the database and data from the database back out?
- We'll look at specific cases where we used:
 - External tables
 - Java Stored Procedures
 - PL /SQL

Situation 1: Mass Mailings

- Database with the names, fax numbers, e-mail addresses, and mailing addresses of about 6,000 physicians
- Requirement: Send mass mailings to these physicians
 - Try faxes first
 - If the faxes didn't get through, or we didn't have a fax number, e-mail
 - If the e-mail didn't get through or we didn't have an e-mail address, print letters and send them.

Faxes

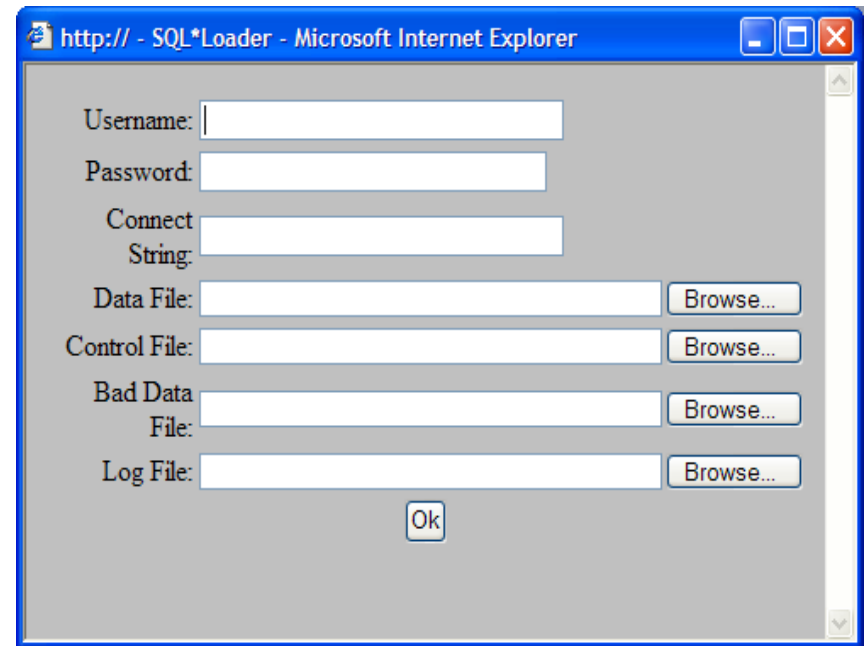
- Query database for names and fax numbers
- Output as CSV
- Upload to fax provider with the document to be sent
- Fax provider returns a CSV with list of faxes that didn't go through
- Got to upload the “bad” list.

Uploading the Bad Fax List

- SQL*Loader was a good tool for the job

```
OPTIONS ( SKIP=1)
LOAD DATA
INTO TABLE "UNDELIVERED_FAXES"
TRUNCATE
FIELDS TERMINATED BY ','
OPTIONALLY ENCLOSED BY '"'
TRAILING NULLCOLS
(TRANSMISSION_DATE SYSDATE,
ADDRESSEE_NAME,
FAX,
ERROR_TYPE,
JOB_NAME)
```

- But it's not user friendly
 - So we needed to put a good face on it
 - VBScript can do this



E -M ail

- With the list of Undelivered Faxes
- A nother Query gets a CSV of e-mail addresses
- Feed that into M ailman to create a mailing list
- Now we need the list of bounced e-mails
- Find it in M ailman's bounce log file:
 - `/var/log/mailman/bounce`
 - Located on the M ailman server, which is also our database server

Reading the Mailman Log

- SQL*Loader would work
- An External Table is better
 - File is on the database server
 - Needs some parsing to use it
- Steps:
 - Create a Directory
 - Create a Table
 - Like combining CREATE TABLE with a SQL*Loader control file

Creating a Directory

CREATE OR REPLACE DIRECTORY

```
MAILMAN_LOGS AS '/var/log/mailman/';
```

- Grantable Privileges
 - READ
 - WRITE
 - Apply only to files in that directory, not to subdirectories
- Other uses of Directories
 - BFILE Datatype
 - UTL_FILE - for Oracle 9i & 10g
 - Directory is preferred to UTL_FILE_DIR init.ora parameter

Creating the External Table

```
CREATE TABLE MAILMAN_BOUNCES
( BOUNCE_DATE DATE,
  BOUNCE_CODE NUMBER(6),
  DETAILS VARCHAR2(127))
ORGANIZATION EXTERNAL
( TYPE ORACLE_LOADER
  DEFAULT DIRECTORY MAILMAN_LOGS
  ACCESS PARAMETERS
    ( RECORDS DELIMITED BY NEWLINE
      NOBADFILE
      NODISCARDFILE
      NOLOGFILE
      SKIP 0
      FIELDS TERMINATED BY ''
      MISSING FIELD VALUES ARE NULL
      REJECT ROWS WITH ALL NULL FIELDS
      (
        BOUNCE_DATE POSITION (1:20) CHAR DATE_FORMAT DATE MASK "Mon dd hh24:mi:ss yyyy",
        BOUNCE_CODE POSITION (21) CHAR ENCLOSED BY '(' AND ')',
        DETAILS POSITION (*:120) CHAR
      )
    )
  LOCATION (MAILMAN_LOGS:'bounce')
)
REJECT LIMIT UNLIMITED
NOPARALLEL
NOMONITORING
/
```

Data from the Bounce Log

BOUNCE_DATE	BOUNCE_CODE	DETAILS
4/11/2006 1:13:52 PM	10151	<BounceRunner at -1210916948> processing 7 queued bounces
4/11/2006 1:13:52 PM	10151	buprenorphine-info: doctor1@organization.org bounce score: 1.0
4/11/2006 1:13:52 PM	10151	buprenorphine-info: doctor2@whatever.net bounce score: 1.0
4/11/2006 1:13:52 PM	10151	buprenorphine-info: doctor3@another.org bounce score: 1.0
4/11/2006 1:13:52 PM	10151	buprenorphine-info: doctor4@earthlink.net bounce score: 1.0
4/11/2006 1:13:52 PM	10151	buprenorphine-info: doctor5@internetpro.net bounce score: 1.0
4/11/2006 1:13:52 PM	10151	buprenorphine-info: doctor6@otponline.net bounce score: 1.0
4/11/2006 1:13:52 PM	10151	buprenorphine-info: doctor7@someother.net bounce score: 1.0

Finishing Up

- Used new Regular Expression functions to get the e-mail addresses
 - Yes, you can use INSTR and SUBSTR, but this is easier
- Matched them to the original list
- One more query to get a mailing list for paper
 - Output to CSV
 - Did mail merge in MS Word



Situation 2: Send a Report in E-Mail

- Report originally written as PL /SQL Server Page (PSP)
- Requirement:
 - Run automatically on the first of each month
 - Send to a list of e-mail addresses
 - Don't re-write if you don't have to

A Little about Web PL/SQL

- Writes to a buffer
- mod_plsql reads the buffer
 - Probably uses OWA package
 - Sends to web browser
 - Normally HTML is in the buffer, but it can contain anything
- OWA_showpage can dump the buffer to DBMS_OUTPUT
- Requires some setup - CGI Environment
 - If you don't, you'll get an ORA-06512

A Little about E-Mail from Oracle

- UTL_SMTP - Oracle 8i and 9i
 - Several shells built on this
 - We use the sample code from OTN
http://otn.oracle.com/sample_code/tech/pl_sql/htdocs/Utl_Smtp.pl
- UTL_MAIL - Oracle 10g
 - Can only send one attachment
- Need an SMTP server
 - Ours is the local sendmail or postfix program (Linux)

The Code: Run the PSP

- A PL/SQL Server Page is just a stored procedure
- To generalize this, we use EXECUTE IMMEDIATE and AUTHID CURRENT_USER

```
/** Runs a procedure that uses the Web PL/SQL Toolkit to write to  
the http buffer */
```

```
PROCEDURE run_procedure (p_proc_spec VARCHAR2)
```

```
IS
```

```
    env_name_arr    OWA.vc_arr;
```

```
    env_val_arr     OWA.vc_arr;
```

```
BEGIN
```

```
    OWA.init_cgi_env (0, env_name_arr, env_val_arr);
```

```
    HTP.init;
```

```
    EXECUTE IMMEDIATE 'begin ' || p_proc_spec || '; end;';
```

```
END run_procedure;
```

The Code: Send the E-Mail

```
HTP.get_page (pagetext, irows);
IF irows > 0 THEN
    page_generated := TRUE;
    conn           := common.e_mail.begin_session;
    common.e_mail.begin_mail_in_session
        (conn      => conn,
         sender     => USER,
         recipients => p_send_to,
         subject    => p_subject,
         mime_type  => common.e_mail.multipart_mime_type
        );
    common.e_mail.attach_text (conn => conn, DATA => p_message);
    -- Write the report as an attachment - we assume that the report
    -- itself has the mime type and other headers embedded in it.
    common.e_mail.write_text (conn, common.e_mail.first_boundary);
END IF;
WHILE irows > 0
LOOP
    FOR i IN 1 .. irows
    LOOP
```

Write the Attachment

- Loop through the array of data from the web toolkit's buffer
- Write it to the attachment.

- Change new lines to carriage return/line feed

```
text_out := utl_string.nl2crlf (pagetext (i));
common.e_mail.write_text (conn, text_out);
```

- Look for the content type in the HTTP header that we just wrote

```
IF UPPER (text_out) LIKE 'CONTENT-TYPE%' THEN
```

- It helps compute a file name, if not provided.

```
filename :=
  get_filename (p_proc_spec, p_filename,
               get_mime_type (text_out));
common.e_mail.write_text
  (conn, 'Content-Disposition: attachment; filename="'
   || filename || '"' || utl_string.crlf);
```

```
END IF;
```

```
END LOOP;
```

- Get the next batch of data from the buffer

```
HTP.get_page (pagetext, irows);
END LOOP;
```

Situation 3: Submissions from the States

- Requirement:
 - List available files
 - Validate a data file and Load it into the database
- Options:
 - SQL*Loader
 - But would need to start it from the database
 - UTL_FILE
 - But would need to parse each record
 - External Table
 - But need to change which file underlies the table

Listing a Directory

- Several great enhancements to UTL_FILE in 9i and 10g
- Listing a directory is not one of them
- When in Doubt, search the WWW
 - Found the utility I needed on ORACLE-BASE (<http://www.oracle-base.com/>) run by Tim Hall
 - The File-API Java Stored Procedure can read a directory
 - Original returns the directory as a comma-delimited list

Adapting File-API

- Two problems with the original
 - Gives access to ANY directory accessible to the Oracle Owner - on *NIX, that is usually “oracle”
 - Easier to use if the directory appears as a table or view.
- Used Tim Hall’s Java but adapted the PL/SQL front-end
 - Based security around Directory objects
 - Returns a user defined data type - `vc_arraytype`
 - Collections can be cast to Table
 - I made a View for each Directory

Changing the External Table on the Fly

- Decided to use an External Table to read the submissions
- To run a V alidate and L oad we have to point the table at this file
 - Need a procedure to do an ALTER TABLE to change LOCATION
 - This is DDL so it will COMMIT implicitly
 - Used the AUTONOMOUS TRANSACTION pragma
 - Warning: ALTER happens even if an exception occurs somewhere else.

Set_Filenames

```
PROCEDURE set_filenames (  
    p_filename      IN    user_external_locations.LOCATION%TYPE  
) IS  
    PRAGMA AUTONOMOUS_TRANSACTION;  
    file_not_found  EXCEPTION;  
    PRAGMA EXCEPTION_INIT (file_not_found, -29913);  
    dummy           VARCHAR2 (1);  
BEGIN  
    EXECUTE IMMEDIATE      'ALTER TABLE submission_raw LOCATION('' ||  
p_filename || '' )';  
    ■ Now let's make sure that the file exists:  
  
    SELECT 'x'  
        INTO dummy  
        FROM submission_raw  
        WHERE ROWNUM < 2;  
EXCEPTION  
    WHEN file_not_found THEN  
        utl_error.raise_error  
            ('TEDS',utl_error.get_errcode ('TEDS','SUBMISSION FILE NOT FOUND'),  
            vc_arraytype (p_table_name, p_filename));  
    WHEN OTHERS THEN  
        RAISE;  
END set_filenames;
```

Multi-User?

- Change for one, Change for all
 - What if someone does `set_filenames` while a load is running?
 - If you have a lot of users, don't do this
- Possible solutions if you don't have a lot of users
 - Give each user a copy of the external table
 - `Set_filenames` now should be `AUTHID CURRENT USER`
 - Changes location of only that user's table
 - Immediately copy the data to a `GLOBAL TEMPORARY TABLE`

Situation 4: Record an E-Mail Response

- Government Form on the Web used by physicians
- Requires a “signature”
 - We send an e-mail to the physician requesting a reply to confirm the information on the form
 - Physician sends reply by e-mail
 - System receives e-mail and records the “signature”
- Requirement
 - Capture the e-mail
 - Record the signature on the proper row in the database

Unix/ Linux E-M ail Capabilities

- Our Mail Server runs Red Hat Linux
- Linux and other Unix-like OSes have powerful mail utilities
 - Sendmail and Postfix can use .forward files to re-route mail
 - Or Procmail can parse the mail
 - Can decide how to handle each incoming message based on heading and/or contents
- We wrote a procmail control file
- P.S. If anyone has done this in Windows, we'd like to hear about it.

Our .procmailrc

```
# .procmailrc
# routes incoming mail
PATH=/bin:/usr/bin:/usr/local/bin:$HOME/bin
SHELL=/bin/bash
LOGFILE=$HOME/procmail.log

# Process Waiver Signature replies
:0
* !^FROM_DAEMON
* !^FROM_MAILER
* ^Subject:.*Waiver Notification #
|process_signature

# Send a copy of anything else to the BWNS Administrator
:0c
```

Shell Script: process_signature

```
#!/bin/bash
# This processes the signature e-mail from the
# practitioner and calls
# sign_waiver to update the database with the
# signature information.

today=`date`
wvr_id=`awk -F# '/^Subject/ {print $2;exit}'
hold.mail`
echo Waiver Notification $wvr_id signature received
$today >>wvr_signature.log
sign_waiver $wvr_id
exit
```

Other Uses for Data Received in E-Mail

- One version we tested called SQL*Loader to archive a copy of the e-mail in a CLOB column
- Extract an attachment with shell script
 - Binary attachments
 - Decode binary attachment with `UTL_ENCODE.BASE64_DECODE`
 - Store it in a BLOB
 - Text Attachment or message
 - Use SQL*Loader to load data from CSV
 - Store XML with XMLDB

Summary

- Made a log file in the OS look like an Oracle table for processing with SQL and PL /SQL
- Reports written with the Web PL /SQL Toolkit were sent as e-mail attachments.
- Read a directory on the database server and displayed it as a dropdown list
- Pointed an external table to the selected file and inserted it into a temporary table for further processing.
- Received an e-mail and used it to update the database.

Wait - There's More!

- Let Web Forms upload files: See my earlier presentation, “Files, Uploads, and Downloads with Web PL /SQL
- Dump a Table to a CSV File: See “Ask Tom”
- Use XMLDB to let people put a file in the database, especially XML files, with ordinary file utilities - FTP, WebDAV: Various authors have written papers on this.
- Get data out as XML with no programming: Lewis Cunningham has a good series in his blog.
- Mod_oradav extension to the OHS, part of Intermedia, is a good alternative to XMLDB - WebDAV with no XML needed.

URLs

- My Web Page (all the code and more): <http://www.smdi.com/employee/JohnF/>
- OTN Sample code for E-Mail utility:
http://otn.oracle.com/sample_code/tech/pl_sql/htdocs/
.
- File-API from Tim Hall (Oracle-Base):
<http://www.oracle-base.com/articles/8i/FileHandlingFr>

More URLs

- Dump a Table to a CSV from Ask Tom:
http://asktom.oracle.com/pls/ask/f?p=4950:8:::::F4950_P8_DISI
- Lewis Cunningham tells how to get XML from the database without code:
<http://blogs.ittoolbox.com/oracle/guide/archives/009160.asp>
- Mod_oradav:
<http://www.oracle.com/technology/software/products/intermedia>

About Me

- Address:
 - Synectics for Management Decisions, Inc.
 - 1901 N. Moore St.
 - Arlington, VA 22209
- E-Mail - Mention ODTUG in Subject line
 - JohnF@smdi.com
- Web Site
 - <http://www.smdi.com/employee/JohnF/>