# PSP_MAIL – DYNAMIC PSP™ ADD-ON
# RELEASE NOTES

Version 1.4.5

## COPYRIGHT INFORMATION AND ACKNOWLEDGEMENTS

DPSP, Dynamic PSP, OPSP and Objective PSP are trademarks of Hit-Media LLC.
DPSP Interpreter, PSP_Mail and this document are Copyright© 2001-2004 by Hit-Media LLC.

Oracle is registered trademark of Oracle Corporation.
PL/SQL, Oracle8*i*, Oracle9*i*, Oracle Internet Application Server, Oracle WebServer and Oracle WebServer Option are trademarks of Oracle Corporation.

Sun, Sun Microsystems, the Sun Logo and Java are trademarks or registered trademarks of Sun Microsystems Inc. in United States and other countries.

Other company or product names are mentioned for identification purposes only and may be service marks, trademarks, or registered trademarks of their respective owners.

Although every effort was taken to make this document as accurate and complete as possible, no guarantees whatsoever are given in regard to document's accuracy and completeness. Also, no guarantees are given that this document fully covers the functionality of the product it describes.

Information in this document is subject to change without notice.

## WHAT IS PSP_MAIL.

PSP_Mail is a Dynamic PSP™ add-on package that allows Oracle8*i*/9*i* developers to send various types of e-mail from DPSP-based or any other PL/SQL applications. PSP_Mail uses the standard `UTL_SMTP` package introduced in Oracle8*i* Release 2 as well as several custom Java™ classes to implement its functionality. PSP_Mail can be used in conjunction with Dynamic PSP or standalone.

These release notes briefly outline changes made to the product in new releases. For complete development history, please refer to `HISTORY.TXT` file accompanying the product.

## WHAT'S NEW IN VERSION 1.4.5

Below is summary of changes and enhancements implemented in PSP_Mail Version 1.4.5 since Version 1.4.4.

❖ **Changed:** updated `UTL_B64` package to Version 1.2.0. The new version support automatic translation to the specified character set in `encode_mime_header`. This feature is controlled by new Boolean flag argument, which defaults to false thus keeping the package compatible with the old code. The new automatic translation feature is only supported on Oracle releases that expose `UTL_GDK`/`UTL_I18N` packages.

❖ **Changed:** updated `UTL_BinFile` to Version 1.0.7, including more complete file manipulation API and new column for file modification time in the directory/archive lists table.

❖ **Changed:** updated license enforcement package, which now does not require `SELECT` privilege on `V$SESSION` view as well as fixes a few rare bugs.

❖ **Fixed:** sometimes space characters were incorrectly removed from headers while folding.

❖ Miscellaneous internal optimizations and fixes.

## WHAT'S NEW IN VERSION 1.4.4

Below is summary of changes and enhancements implemented in PSP_Mail Version 1.4.4 since Version 1.4.3.

❖ **Changed:** header folding now doesn't take place within angle brackets – folding there could confuse some MTAs and cause them to corrupt the data inside the brackets.

❖ **Changed:** `UTL_BinFile` updated to Version 1.0.6, including important workaround for a JDBC driver issue in Oracle9*i* and later.

❖ **Changed:** `reqcheck.sql` script optimized.

## WHAT'S NEW IN VERSION 1.4.3

Below is summary of changes and enhancements implemented in PSP_Mail Version 1.4.3 since Version 1.4.2.

❖ **Fixed:** UTC offset was incorrectly calculated for `Date` header when `GMT_DIFF` contained fractional part (wrong minutes were calculated). This has been fixed.

❖ **Changed:** on Oracle9*i* and later PSP_Mail now uses session time zone information for `Date` header generation. `GMT_DIFF` is now ignored on 9*i* and later, though it is still used on 8*i*.

❖ **Fixed:** `MXLookup` function was not resolving standalone hosts. For example, if you attempted to send email directly to `user@host.domain`, `MX` lookup for `host.domain` domain was performed, but no `A` lookup to verify if `host.domain` is a standalone host and can accept mail by itself was done if `MX` lookup failed. This has been fixed.

## WHAT'S NEW IN VERSION 1.4.2

Below is summary of changes and enhancements implemented in PSP_Mail Version 1.4.2 since Version 1.41.

❖ **Added:** SMTP Authentication Extension ([RFC-2554](#)) support for all `SendMailXXX` functions. Two new variables were added to the package specification: `AUTH_USERNAME` and `AUTH_PASSWORD`. These two variables are used to authenticate with the SMTP server if both are not `NULL` and the server supports

LOGIN or CRAM-MD5 authentication method. CRAM-MD5 method is used by default, with automatic fallback to the less secure LOGIN if CRAM-MD5 is not supported. com.nnetworks.CryptoHash class was added to the psp_mail.jar archive – this class implements HMAC-MD5 ([RFC-2104](#)) algorithm used in CRAM-MD5 authentication method. AUTH_USERNAME and AUTH_PASSWORD variables are NULL by default and should be initialized and reset by the caller as needed before initiating SMTP transactions with SendMail*XXX* functions.

❖ **Changed** versioning to more intuitive major.minor.release.build style.

## WHAT'S NEW IN VERSION 1.41

Below is summary of changes and enhancements implemented in PSP_Mail Version 1.41 since Version 1.40.

❖ **Added:** better error reporting for multiple recipient transactions through the relay server: now rejected recipients are reported through new 511 Recipient rejected error.

❖ **Improved:** while sending to multiple recipients, the package now checks for RCPT command reply code and detects when server's recipient buffer is full. When such reply code is received, the package commits the mail transaction and starts new transaction for remaining recipients. This change is to comply with RFC-2821 minimum limit of 100 recipients per transaction (RFC says that the server must accept at least 100 recipients per transaction, but not necessarily more than that, so we need to watch when the server's buffer is full and initiate new transaction. We do not want to break the list in chunks of predefined length though, because the server may accept more than 100 recipients at once and in this case fewer transactions may be needed.)

❖ In SendMailEx2() V_TO argument is now not supposed to contain a valid email address list, it can be any string. Note that in this case at least one address should be supplied in *either* V_CC or V_BCC; otherwise an error will be generated. Also note that presence of single @ character in V_TO indicates that this argument contains at least one email address and will cause address line decoding to take place.

❖ CLOB_To_Strings() now won't raise an exception if NULL CLOB is passed in, empty PSP_Mail.Strings array will be returned instead.

❖ **Fixed:** in SendMailEx2() Bcc: recipients could appear in To: header of the message. This has been fixed.

❖ **Fixed:** exceptions were not sometimes handled properly and were propagated to the caller (in contrast with what the documentation says.) This has been fixed.

❖ **Added:** added automatic EHLO negotiation for 8-bit MIME transport if Content-Transfer-Encoding is not explicitly specified in OTHER_HEADERS or specified as 8bit. If the remote server does not support EHLO or by occasion does not support 8BITMIME extension, the message will still be delivered with Content-Transfer-Encoding: 8bit and *may* be crippled by the server. For 100% guaranteed delivery you should encode the message and/or headers with non-ASCII characters using Base64 or Quoted-Printable.

❖ **Updated:** UTL_BinFile package is updated to v1.04. This version adds ability to create and modify (add, update and delete files and directories) ZIP archives stored as internal BLOBs. External archives may be manipulated as well by loading them into BLOBs using Load_BLOB_From_File(), altering and then saving back to disk using Save_BLOB_To_File(). This version also fixes a bug in ZIP archive contents listing routines, which caused IndexOutOfBoundsException to be thrown in most cases.

❖ **Updated:** loadmail9i scripts now load correct FileOps9i.jar file (UTL_BinFile supporting Java classes.)

## WHAT'S NEW IN VERSION 1.40

Below is summary of changes and enhancements implemented in PSP_Mail Version 1.40 since Version 1.35.

❖ **Updated:** UTL_BinFile package is updated to v1.03. This version adds ability to delete and rename files and directories, create new directories, and compress/decompress BLOBs using 'deflate' and 'gzip' compression algorithms.

❖ **Added:** In ALWAYS_RELAY mode multiple recipient versions of SendMailEx and SendMailEx2 are now sending to all recipients in one mail transaction. Speed improvement and bandwidth savings are enormous, but nothing's free: the functions can't determine delivery status immediately in this mode (unless, of course, the relay server connection fails) and always report success for all targets - the relay

server will then report actual status to the address specified as SENDER or in Return-Path header. This prevents from verifying addresses by return codes, which is possible in direct send mode.

This feature is further controlled by the new **RELAY_IN_SINGLETRANS** package variable, which is TRUE by default. If you do not want the new behavior in ALWAYS_RELAY mode, set RELAY_IN_SINGLETRANS to FALSE before calling any SendMailEx multi-recipient functions and the functions will generate and send a copy of the message for each recipient in separate transactions, as before.

❖ **Fix:** fixed prerequisites check script (reqcheck.sql) so that it no longer looks for some TCP-related Java classes on Oracle9i Release 2 (9.2) and later. These classes are no longer necessary and not shipped, as UTL_TCP was re-implemented in native C starting with this release.

❖ **Added**: installation scripts for Oracle9i (loadmail9i), backported them to 1.35 along with new reqcheck.sql.

❖ Limited error report lines to 2000 characters, so that they can be logged in a VARCHAR2 table column.

## WHAT'S NEW IN VERSION 1.35

Below is summary of changes and enhancements implemented in PSP_Mail Version 1.35 since Version 1.34.

❖ **Fix:** not all versions of SendMailEx were parsing the Return-Path header and assigning the proper reverse path to the message.

❖ **Fix:** small bug in the MIME header parser, which could cause an invalid trailing character to be appended to the extracted header value in some cases.

❖ Repackaged JAR files without compression to make them compatible with loadjava utility version 8.1.6, which does not accept compressed JAR archives.

## WHAT'S NEW IN VERSION 1.34

Below is summary of changes and enhancements implemented in PSP_Mail Version 1.34 since Version 1.33.

❖ **Fix:** line breaks in message body were not always properly checked and fixed.

❖ **Fix:** fixed bug introduced in v1.33 where some headers could be improperly placed into message body corrupting the message if certain values for OTHER_HEADERS were specified with SendMailEx call with attachment(s).

❖ Added support for Return-Path header: if you supply this header in OTHER_HEADERS, PSP_Mail will use the address specified in it with SMTP MAIL command to ensure that delivery failure notifications are sent to this address, otherwise sender address will be used. This extra measure is taken because some SMTP servers do not honor Return-Path header and always bounce to the address specified with MAIL command. See FAQ Section 2.2.8 for details.

❖ Improved internal long headers wrapping logic.

❖ Updated UTL_B64 to version 1.11.007 - line length for encoded LOBs is increased to 76 characters + CR/LF. This makes generated messages a bit shorter and more standard-compliant. Also, LOB encoding speed improved significantly (2 to 3 times faster) by adding LOB read buffering.

❖ Removed automatic Content-Length header generation for attachment sections as it is not necessary.

❖ Installation: reqcheck.sql now verifies that running user is SYS.

❖ Made the package resistant to DBMS_SESSION.RESET_PACKAGE effect of resetting package variables to their defaults while NOT re-executing the package init block. Changed Version constant to function to avoid the said effect.

## WHAT'S NEW IN VERSION 1.33

Below is summary of changes and enhancements implemented in PSP_Mail Version 1.33 since Version 1.32.

❖ UTL_BINFILE package upgraded to v1.01.005, now the package supports listing and extracting files from external (OS files) and internal (BLOB) ZIP/JAR archives. Documented new functionality in PSP_Mail User's Guide and Reference.

❖ Internal loops over PL/SQL collections (arrays) are rewritten to be resistant to holes in arrays (missing/unassigned elements.)

❖ **Fix:** recipient names encoded to Base64 are now enclosed in double quotes, otherwise mail clients like Outlook/Outlook Express may misinterpret them and display them as separate recipient addresses.

❖ **Fix:** internal long header wrapping routine now does not wrap Base64-encoded headers, as these should be already correctly broken into pieces. It is assumed that Base64-encoded headers are generated with `UTL_B64.ENCODE_MIME_HEADER` function, which correctly wraps the resulting header.

❖ **Fix:** fixed `OTHER_HEADERS` not getting into the main email headers section (they were only inserted into mail body section.) `Content-Transfer-Encoding` header, if present, is stripped from main email headers and is only issued in body section to prevent mail client attempts to decode the whole message since it's not actually encoded.

❖ Internal long header wrapping routine now forcibly wraps headers at 80th character if no delimiter is located in current part.

## WHAT'S NEW IN VERSION 1.32

Below is summary of changes and enhancements implemented in PSP_Mail Version 1.32 since Version 1.31.

❖ Exposed `MXLookup` function, which was previously used only internally. This function allows performing DNS queries for MX for host or domain. This function always performs DNS query regardless the `ALWAYS_RELAY` setting while `getMXForEmail` always returns default relay server address if `ALWAYS_RELAY` is set to `TRUE`. See User's Guide for function reference and examples.

❖ LOB arguments to `SendMailXXX` functions are no longer `IN OUT`. All LOBs are now read-only and their state is unaltered during mail assembly and transfer.

❖ Package now uses `UTL_SMTP.WRITE_RAW_DATA()` to avoid automatic conversion of message content to `US7ASCII` and allow for correct 8-bit data transfers. Encoding messages and headers composed in national character sets to Base64 or quoted-printable is now optional; PSP_Mail sets `8bit` content transfer encoding for messages by default if different encoding is not specified in `OTHER_HEADERS` argument.

❖ Optimized `UTL_B64` package LOB operations: added LOB write buffering and minimized LOB write calls, made all LOB arguments `IN` to enable using LOB-related functions in SQL. Package upgraded to v1.1.0005

## WHAT'S NEW IN VERSION 1.31

Below is summary of changes and enhancements implemented in PSP_Mail Version 1.31 since Version 1.30.

❖ **Fix:** Fixed minor bug introduced in v1.30 where PSP_Mail would incorrectly convert address in form 'user@host.com' into canonic representation.

## WHAT'S NEW IN VERSION 1.30

Below is summary of changes and enhancements implemented in PSP_Mail Version 1.30 since Version 1.20.

❖ Addresses in form 'User Name email@host.com' (where email address is not enclosed in angle brackets) are now supported and correctly parsed by `extractName` and `extractEmail` functions. If possible, you should adhere to [RFC-822](#) format though.

❖ Added `CLOB_To_Strings` function that converts input `CLOB` to `PSP_Mail.Strings` collection. May be used to pass `CLOB` mail body to `SendMail` routines. See documentation for example of use of this new function.

❖ Potential problems with DBCS databases: `DBMS_LOB` treats amount for `CLOB` as number of characters requested while `VARCHAR2` always treats size as bytes - this can lead to internal buffer overflows on DBCS systems as buffer should be twice as large as amount of data requested from `CLOB`. This fact is taken into account and all `CLOB` read operations are now performed with amount being half of the buffer size. This may have slight negative impact on performance when working with large `CLOB`s.

❖ Installation script now prompts for package configuration parameters so you don't need to edit package header file (`PSP_MAIL.PLH`) before installing the product. If you need to change configuration later though, you will still need to edit and recompile package header.

❖ Renamed internal `TO_BLOB` conversion routine to avoid name conflict with standard function introduced in 9i.

❖ Identified issue where `ORA-1460` could be signaled on 8.1.7.0 when email with attachments was sent as side effect of Oracle bug #1512417, which is fixed in 8.1.7.1 patchset. Customers hitting this error when sending email with attachments should apply this or later patchset (for Oracle8*i* Release 2 Version 8.1.6, one-off patch may be required, please consult with Oracle Support Services about patch availability.)

❖ Made a few more optimizations to the code.

## WHAT'S NEW IN VERSION 1.20

Below is summary of changes and enhancements implemented in PSP_Mail Version 1.20 since Version 1.12.

❖ Optimized internal `CLOB` to `BLOB` conversion routine.
❖ Improved performance when email is sent to multiple recipients.
❖ Optimized forced close of failed SMTP connections.
❖ Several other minor optimizations and tweaks.

## WHAT'S NEW IN VERSION 1.12

Below is summary of changes and enhancements implemented in PSP_Mail Version 1.12 since Version 1.11. For complete description and examples of use of new and updated functions and structures please refer to PSP_Mail User's Guide and Reference.

❖ Added `GMT_DIFF` configuration parameter used to generate correct timestamps for email messages. Added automatic Date header generation.
❖ Added `UTL_BINFILE` package and corresponding Java classes. `UTL_BINFILE` allows to list files in OS directories, load binary OS files into `BLOB`s and save `BLOB`s into binary OS files. Correct Java `java.io.FilePermission` permissions must be set for invoking Oracle user with `DBMS_JAVA.GRANT_PERMISSION()` for `UTL_BINFILE` to be able to list directories, read and/or write OS files.
❖ Documented `UTL_BINFILE` in PSP_Mail main documentation.
❖ PSP_Mail now sets `Content-Length` header for attached files to provide mail clients with a faster way to determine attachment's size.

## WHAT'S NEW IN VERSION 1.11

Below is summary of changes and enhancements implemented in PSP_Mail Version 1.11 since Version 1.1. For complete description and examples of use of new and updated functions and structures please refer to PSP_Mail User's Guide and Reference.

❖ Added `ENCODE_MIME_HEADER` function to `UTL_B64` package to allow easy encoding of headers composed in national character sets according to RFC-2047.
❖ Added support for encoded headers to PSP_Mail (disabled quoting of such header strings as this is illegal per RFC-2047).
❖ Fully documented `UTL_B64` package in User's Guide and Reference and added some examples of sending emails composed in national character sets.
❖ Enhanced MIME headers generation to conform with RFC-822 80 characters per header line limitation.
❖ Several minor bug fixes.

## WHAT'S NEW IN VERSION 1.1

Below is summary of changes and enhancements implemented in PSP_Mail Version 1.1 since Version 1.0. For complete description and examples of use of new and updated functions and structures please refer to PSP_Mail User's Guide and Reference.

- ❖ Added new function `SendMailEx2` which simplifies sending email even further by allowing to specify three recipient lists for To:, Cc: and Bcc: headers respectively, and optionally specify one or more attachments to the message.
- ❖ Added new functions for converting email addresses to canonic form.
- ❖ Added new field to the attachment descriptor record – `content_id` – allowing to assign each attachment its unique content ID, which can be referenced from main mail body (for example, to include images into HTML-formatted message, by specifying `src="cid:content-id-of-the-attached-image"` in `<IMG>` tag).
- ❖ Automatic canonization of email addresses (all addresses are internally automatically converted to ["literal"] <email@host> form as per [RFC-822](#)) for full compatibility with strict SMTP servers.
- ❖ Enhanced compatibility with newer versions of SMTP servers, like QMail and sendmail 9.
- ❖ Automatic conversion of lone or improperly ordered CR and/or LF sequences to conform to [RFC-822](#).
- ❖ New `ALWAYS_RELAY` option in package header to force PSP_Mail to send all mail through relay server.
- ❖ Added `encodeva` function to supplemental `UTL_B64` package, which encodes Strings array using Base64 encoding and returns encoded content in another Strings array. This function can be used to encode message body to ensure correct delivery in case when national (non-english) character set is used. An example of use:

```
declare
 msg psp_mail.Strings;
begin
 msg(1) := 'Some text in national language';
 msg := utl_b64.encodeva(msg);
 psp_mail.SendMailEx2(
          v_to => '"scott tiger" <scott@test.us.oracle.com>',
          v_subject => 'Test',
          v_body => msg,
          other_headers => 'Content-Transfer-Encoding: base64',
          content_type => 'text/plain;'||psp_mail.nlt||'charset="national-cs"',
          attachments => psp_mail.tbatt_empty
          );
end;
```

  The above example will send the message using 'national-cs' character set and Base64 encoding (please note that some SMTP servers may automatically convert Base64 encoding to 8bit encoding if they detect that encoded content can be represented in 8bit).
- ❖ Several bug fixes.