

Improving the business efficiency of OEBS workflow

Part 1: Missing e-mail addresses

By
Gary Piper
WWW.PIPER-Rx.com

Introduction

Recently whilst reviewing my OEBS workflow notification subsystem, as well as the very large number of unanswered workflow notifications (that's a whole other story) I came across a number of records in both the [wf_local_users](#) and [wf_local_roles](#) objects with a **null** e-mail address.

Adding to the complexity is that both the [wf_local_roles](#) and [wf_local_roles](#) tables store information including e-mail addresses originating from various other Oracle Applications modules such as [applsyst.fnd_users](#), [hr.per_all_people_f](#) and [ar.ra_contacts](#)... The information in these objects is synchronised from base application tables using the WF synchronisation concurrent programs. Oh, and to make matters worse, the source information is very OEBS version dependant.

So the big questions are:

- ❖ When the workflow wants to send a notification to a workflow user or role where the email address is null, where does it go?
- ❖ What happens to the workflow?
- ❖ Does the business process stop until the notification issue is rectified?

The one saving grace in these scenarios is that if there are any missing or invalid e-mail address it will most likely result a notification error and adding an entry into the SYSADMIN work list.... ☺ I am sure someone will get sick of the entries and correct the problem.....

In part 1 of this series on improving the business efficiency of OEBS workflow, we will cover how to identify missing (null) e-mail addresses, and in part 2 to we will discuss how to validate existing e-mail addresses.

A known issue?

This must be a known issue as Oracle provides at least two (2) notification e-mail address checks.

The first is the **wfdirchk.sql** program.

This program runs a number of checks, one of which is to check for any role with a notification preference of receiving e-mail notifications that has a null e-mail address.

The second is an Oracle Workflow Diagnostics test "Notification preference validation test".

This test checks that all roles with a notification preference of receiving e-mail notifications have an e-mail address. It does not check the address for validity....

The two (2) checks quite likely the same check but accessed in two ways.....However both do not check the address for validity.... This is something we will cover in part 2 of this series.

The importance of a SYSADMIN E-mail Address

By default the application's SYSADMIN account has a default WF notification preference of receive e-mail notifications. This account will typically be the recipient of WF notification errors.

You can however set the WF preference for SYSADMIN to "Do not send me mail" in which case SYSADMIN will not be sent any e-mail notifications.... I liken this to hiding the problem as the SYSADMIN must now review their notification via the Oracle Workflow Work list Web page. Out of sight out of mind?.... Possibly; given the number of SYSADMIN notifications I generally find when reviewing OEBS applications.

So please make sure the SYSADMIN account in [fnd_users](#) has a valid e-mail address and that the WF notification preference is set to receive e-mail notifications....

To quickly find the role that currently has WF admin privileges:

```
Select text
From applsys.wf_resources
Where name = 'WF_ADMIN_ROLE'
```

To check the current e-mail address for the SYSADMIN account

```
SELECT user_name,
       description,
       email_address
```

```
FROM applsys.fnd_user  
WHERE user_name = 'SYSADMIN'
```

What happens when an invalid e-mail address is encountered?

When the notification mailer cannot deliver a notification as a result of the e-mail address being invalid, it:

- ❖ Sets the notification mail status to **FAILED**
- ❖ Adds the e-mail address to its own invalid e-mail address list. On any subsequent request to send a notification to an addresses recorded in the notification mailers invalid address list, the notification mailer sets the mail status to **FAILED** without attempting to send the notification.
- ❖ Changes the notification preference of the recipient to **DISABLED**.
- ❖ Sends a notification to the SYSADMIN (default) user detailing that one or more notifications could not be sent and that the offending users notification preference has been set to **DISABLED**

Note: I am lead to believe that each notification mailer stores up to 100 e-mail addresses in its invalid address list. When more e-mail addresses are added to the list the oldest addresses drop out first. Also, stopping and starting the notification mailer clears the list.

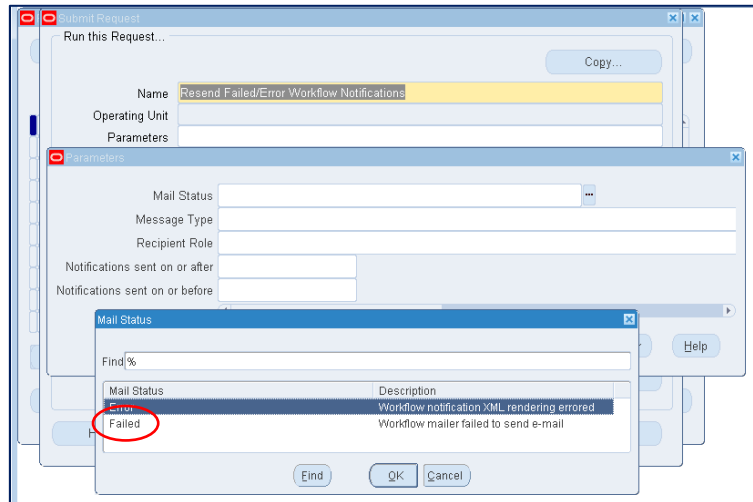
Fixing the issue

Once the invalid e-mail address issue has been corrected at the application module level, remembering the [wf_local_roles](#) and [wf_local_roles](#) tables are synchronised.

You will then need to reset the user's notification preference. Once this is done you can run the concurrent program **Resend Failed/Error Workflow Notifications** to resend workflow notifications that could not be sent.

Example Screen Shot

Improving the business efficiency of OEBS workflow Part 1: Missing e-mail addresses



Note: The number of parameters are OEBS version dependant

Of course as always, you will have tested this process first

Assessing the damage

Everyone likes to think their application does not have any issues.... Or as one CIO said to me once “we don’t know what we don’t know....”

Also just because you don’t have any missing e-mail addresses now, new employees, customer’s etc... are being added all the time, so it is worth adding a null e-mail check to your regular maintenance program.

So first let’s see if the workflow notification mailer has detected and disabled any currently active roles:

```
SELECT 'WF_LOCAL_ROLES' source,
       count(*)
FROM applsys.wf_local_roles wlr
WHERE wlr.email_address is null
      and (wlr.expiration_date is null
            or wlr.expiration_date > sysdate)
      and wlr.status = 'ACTIVE'
      and wlr.notification_preference = 'DISABLED'
UNION ALL
SELECT 'WF_LOCAL_USERS' source,
       count(*)
FROM applsys.wf_local_users wlu
WHERE wlu.email_address is null
      and (wlu.expiration_date is null
            or wlu.expiration_date > sysdate)
      and wlu.status = 'ACTIVE'
      and wlu.notification_preference = 'DISABLED';
```

I won't ask how many you found.....

Remember in part 1 we are only covering null e-mail addresses; in part 2 we will cover e-mail address validation.

Seeking out the guilty

The following SQL will list all those entries in both *wf_local_roles* and *wf_local_roles* objects that have a null e-mail address:

```
SELECT 'WF_LOCAL_ROLES' source,
       wlr.name,
       wlr.orig_system_id,
       wlr.orig_system,
       wlr.email_address
FROM   applsys.wf_local_roles wlr
WHERE  wlr.email_address is null
       and (wlr.expiration_date is null
            or wlr.expiration_date > sysdate)
       and wlr.status = 'ACTIVE'
       and wlr.notification_preference != 'QUERY'
UNION ALL
SELECT 'WF_LOCAL_USERS' source,
       wlu.name,
       wlu.orig_system_id,
       wlu.orig_system,
       wlu.email_address
FROM   applsys.wf_local_users wlu
WHERE  wlu.email_address is null
       and (wlu.expiration_date is null
            or wlu.expiration_date > sysdate)
       and wlu.status = 'ACTIVE'
       and wlu.notification_preference not in 'QUERY'
ORDER by 1, 2;
```

Now you have a list of roles with a null e-mail address from the *wf_local_roles* and *wf_local_roles* tables. You will need to use the *orig_system* identifier to find the base table that holds the master e-mail address and correct the issue at the source ready for the next synchronisation run (should be daily). Some of the base application module objects include *applsys.fnd_users*, *hr.per_all_people_f* and *ar.ra_contacts*... However, many of the base objects used are very OEBS version dependent.

Now it's up to you to fix the e-mail address. Time to practice your delegation skills ☺

Conclusion

Given the propensity for null and invalid e-mail addresses to clog up workflow, if you find just one e-mail address that is fixed where subsequent workflow notifications do not get sent to an invalid address or generate a workflow error, it was worth the effort.

Want to know more?

There is loads more **FREE** information on this topic and all aspects of OEBS Application Administration at the **PIPER-Rx** website. After over 20+ years working with Oracle (the

product, not the Company) and Oracle E-Business Suite (since Release 5) I have visited countless sites and pretty much seen it all when it comes to Applications Administration. Since the late 1990's I have spent more time sharing these learnings and the most popular papers and case studies I have presented are available at the PIPER-Rx.com website as well as a whole host of Tips and Reports I have used throughout my career.

All information at the PIPER-Rx.com website is **FREE** so why not check it out....I hope you find it useful! – **30,000+ downloaders to date can't be wrong!**

Disclaimer

The material contained in this document is provided by the author "as is" and any express or implied warranties, including, but not limited to, any implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall the author be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of any content or information, even if advised of the possibility of such damage. It is always recommended that you seek independent, professional advice before implementing any ideas or changes to ensure that they are appropriate.

*Oracle®, Oracle Applications® & Oracle E-Business Suite® are registered trademarks of Oracle Corporation
TOAD® is a registered trademark of Quest Software*