

A large, semi-transparent blue diamond shape is positioned in the center-left of the image, containing the text 'ALERTAS' on top, '6.4.3' in the middle, and '02/09/2022' at the bottom. The background is a photograph of a modern, multi-story building with a large, illuminated 'D' logo on its facade. The 'D' is light blue and has the words 'D' in large letters, 'INNOVACION' in smaller letters above it, and 'CENTRO DE' and 'C' on the right side. The building has a glass and steel facade with many windows.

ALERTAS
6.4.3
02/09/2022

Tabla de contenido

INTRODUCCIÓN	3
REQUISITOS	3
PARAMETRIZACIÓN	4
MOTOR DE ALERTAS	4
SERVIDORES SMTP	6
SERVIDORES SMTP ADICIONALES	7
PROVEEDOR ENVÍO SMS	8
<i>Esendex</i>	8
CREAR UNA ALERTA	9
DATOS BÁSICOS	9
PARÁMETROS AVANZADOS	11
PLANTILLA DE NOTIFICACIÓN	11
DESTINATARIOS DE CORREO ELECTRÓNICO Y ADJUNTOS	12
ADJUNTOS	14
DESTINATARIOS DE CORREO ELECTRÓNICO	15
OTROS DESTINATARIOS	15
LOG ALERTA	16
EJECUTAR / PROBAR TAREA	16
GESTIÓN DE FECHA DE ÚLTIMA EJECUCIÓN	17
CONSULTAR ERRORES DE LAS ALERTAS	18
INVOCAR UNA ALERTA DESDE UN CÓDIGO PL/SQL	18
GRABAR COLA DE ALERTA CON ARCHIVOS ADJUNTOS EN FORMATO BLOB	21
GESTIÓN DE LA ALERTA A TRAVÉS DEL PL/SQL	21
PROPIEDADES	21
AÑADIR INFORME DEL GENERADOR DE INFORMES DESDE EL PL/SQL DE LA ALERTA	23
AÑADIR ARCHIVO BINARIO BLOB DESDE EL PL/SQL DE LA ALERTA	24
PASO DE PARÁMETROS A INFORMES DEL GENERADOR DE INFORMES	25
EJEMPLO ENVÍO DE UN PDF CON UNA FACTURA DESDE LA ALERTA	25
PARSER	26
<i>Introducción</i>	26
<i>Tipos de etiquetas</i>	26
<i>Propiedades generales del proceso</i>	28
<i>Variables</i>	28
<i>Propiedades de tabla</i>	28
<i>Propiedades de columna</i>	29
CÓDIGO PL/SQL:	29
ASUNTO	30
TEXTO	30
TEXTO HTML	30
RESULTADO EN FORMATO TEXTO PLANO	31
RESULTADO EN FORMATO HTML	31
APÉNDICES	32
AUTORIZAR DESCARGAS DE DIRECCIONES WEB	32
GENERACIÓN DE REPORTS Y ENVÍO POR ALERTA	32

Introducción

El objetivo principal de las alertas es la de realizar notificaciones de forma automatizada, aunque también se pueden usar para realizar procesos de cálculo que no generen ningún tipo de notificación.

Las alertas pueden ser dos tipos:

- **Tareas programadas:** Se ejecutan con periodicidad preestablecida.
- **Invocados desde cualquier código PL/SQL de Libra:** Se ejecutan en cualquier momento sin necesidad de establecer una periodicidad.

Las notificaciones que pueden generar las alertas son de los siguientes tipos:

- **Correo electrónico:** La notificación la recibe el usuario en su correo electrónico, pudiendo llevar adjuntos:
 - Hoja de cálculo realizada por el generador de informes de Libra.
 - Archivos generados desde el PL/SQL de la alerta.
- **Al menú de Libra:** La notificación la verá el usuario en el menú de Libra, no puede incluir adjuntos y el formato del texto no puede ser HTML.
- **SMS:** Tiene que estar configurado un proveedor de SMS en parámetros generales de menú.

Requisitos

- Para el envío de correos electrónicos, la base de datos debe de tener conexión directa a un servidor SMTP (que puede estar dentro de la misma red de la base de datos), en caso de ser un servidor SMTP que se encuentre en Internet, la base de datos deberá tener conexión directa a Internet. También es compatible con el envío de correos electrónicos a través de Galileo, por lo que si la base de datos no tiene conexión directa a Internet habrá que configurar el envío de correos electrónicos a través de la pasarela Galileo.
- Para el envío de notificaciones por Twitter debe de estar instalado el servicio GAL_TWITTER.
- Debe de estar configurado el paquete JMAIL (No es necesario si se usa GAL_MAIL).
- Para poder incluir adjuntos debe de estar configurado el directorio BLOB_TEMP.
- El usuario de Oracle debe de tener el GRANT CREATE_JOB asignado.
- El parámetro de Oracle JOB_QUEUE_PROCESSES debe de tener un valor superior a 0.

Parametrización

Motor de Alertas.



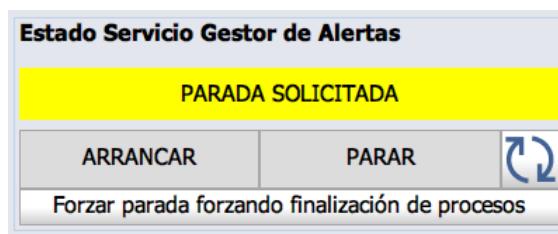
Para el funcionamiento de las Alertas es necesario que se encuentre arrancado el motor de Alertas. Para arrancarlo hay que ir a los parámetros generales de Libra (programa U_MPRMEN) a la pestaña “Notificaciones / Alertas” y pulsar en el botón “ARRANCAR”. También se puede hacer desde el mantenimiento de alertas o desde el programa de colas de alertas, en el plug-in “Estado Servicio Gestor de Alertas”



Si está arrancado y se desea parar (necesario para aplicar actualizaciones de Libra) simplemente hay que pulsar el botón “PARAR”.

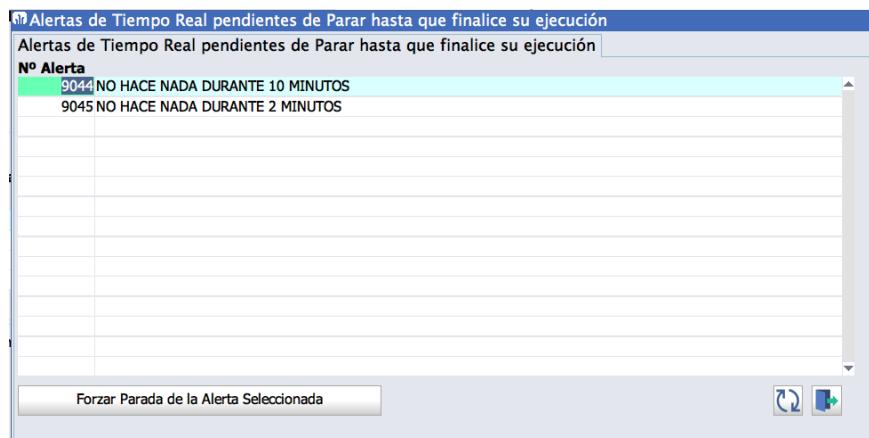


Al pulsar el botón de parada, se indica al motor que debe finalizar la ejecución de una forma ordenada, en el caso de que se esté ejecutando en ese momento una alerta se pasará al estado “SOLICITADA PARADA” y se habilitará el botón “Forzar parada forzando finalización de procesos”, al usar ese botón la alerta que se esté ejecutando en ese momento se le forzará su finalización.



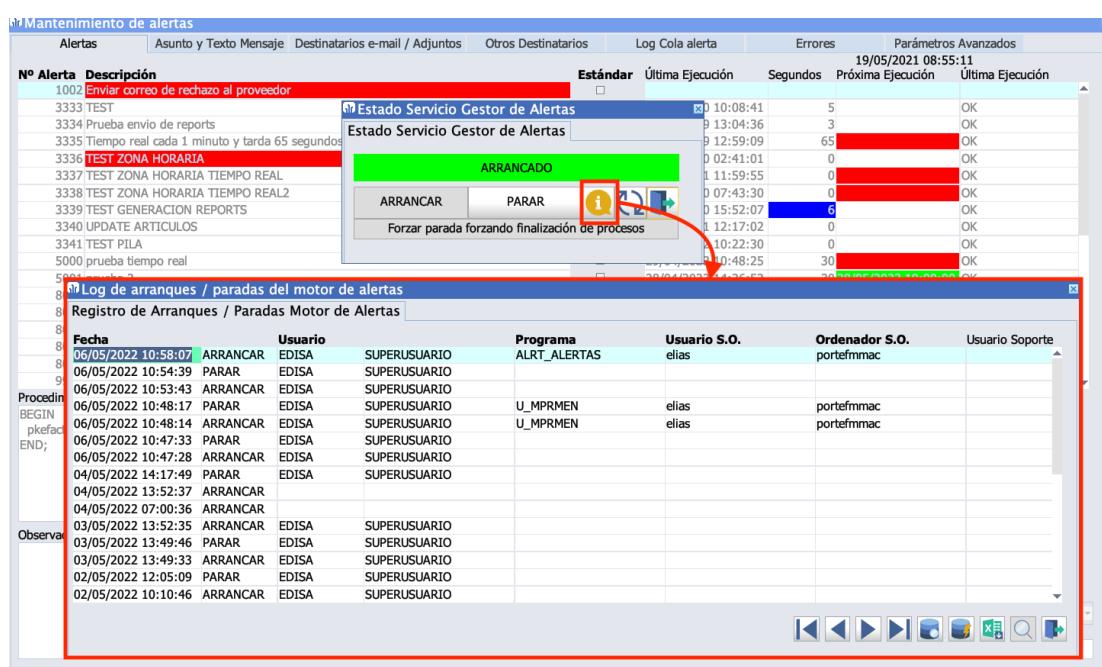
Al parar el motor de alertas, también se pararán también las alertas de tiempo real (salvo las que tengan activada la check “Bloquear Arranque / Parada con motor de Alertas”). En el caso de que

en el momento de solicitar la parada del motor de alertas alguna de las alertas de tiempo real se esté ejecutando, se mostrarán en una ventana y se podrá forzar su finalización de manera individual abortando la ejecución de esta.



- Verificar motor de alertas cada (horas):** Si una alerta se queda bloqueada por cualquier motivo, al ser una ejecución en cola, bloquea el resto de las alertas. Mediante este parámetro se añade un servicio adicional que verifica que no exista ninguna alerta bloqueando el servicio, en caso de que la alerta en ejecución lleve más del tiempo indicado en ejecución, será finalizada. Es importante indicar un tiempo en horas lo suficientemente alto para que en un funcionamiento normal de todas las alertas tengan tiempo para finalizarse.
- Zona Horaria Ejecución Alertas:** Libra puede trabajar con empresas en la misma base de datos con zonas horarias diferentes, pero el servicio de alertas se ejecuta de forma independiente de cualquier empresa de Libra. Mediante este parámetro se indica en qué zona horaria se desea que se ejecute el motor de alertas. Para buscar una zona horaria correcta se puede usar la lista de valores.
- Activar mensaje de alertas paradas:** Al activarse esa check se les informará a todos los usuarios que están marcados como "Superusuario" al salir y entrar de Libra que el motor de alertas está parado. Cuando se pulsa el botón "PARAR" automáticamente se activa la check.

Con el botón  se puede consultar el log de arranques y paradas del motor de alertas.



Mantenimiento de alertas

Alertas | Asunto y Texto Mensaje: Destinatarios e-mail / Adjuntos | Otros Destinatarios | Log Cola alerta | Errores | Parámetros Avanzados

19/05/2021 08:55:11

Nº Alerta	Descripción	Estado	Última Ejecución	Segundos	Próxima Ejecución	Última Ejecución
1002	Enviar correo de rechazo al proveedor	ARRANCAR	10:08:41	5	OK	
3333	TEST	ARRANCAR	13:04:36	3	OK	
3334	Prueba envio de reports	ARRANCAR	12:59:09	65	OK	
3335	Tiempo real cada 1 minuto y tarda 65 segundos	ARRANCAR	02:41:01	0	OK	
3336	TEST ZONA HORARIA	ARRANCAR	11:59:55	0	OK	
3337	TEST ZONA HORARIA TIEMPO REAL	ARRANCAR	07:43:30	0	OK	
3338	TEST ZONA HORARIA TIEMPO REAL2	ARRANCAR	15:52:07	6	OK	
3339	TEST GENERACION REPORTS	ARRANCAR	12:17:02	0	OK	
3340	UPDATE ARTICULOS	ARRANCAR	10:22:30	0	OK	
3341	TEST PILA	ARRANCAR	10:48:25	30	OK	
5000	prueba tiempo real	ARRANCAR	20/05/2021 08:55:11	0	OK	
5001	prueba 2	ARRANCAR	20/05/2021 08:55:11	0	OK	

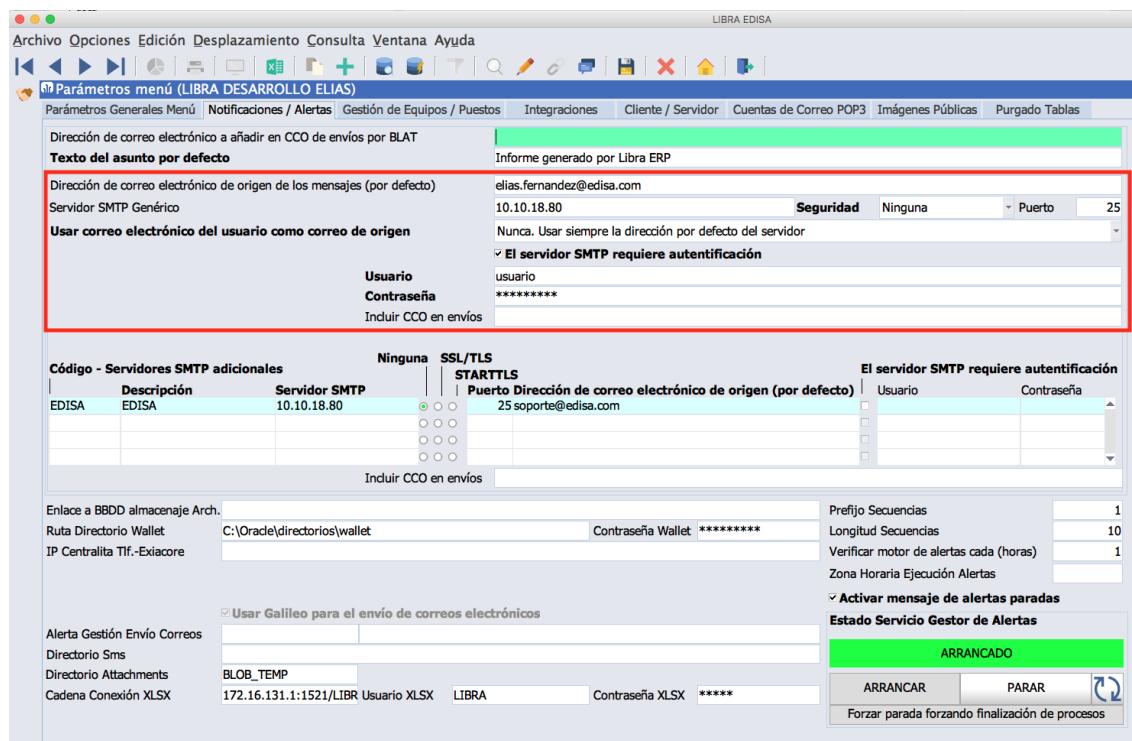
Log de arranques / paradas del motor de alertas

Registro de Arranques / Paradas Motor de Alertas

Fecha	Usuario	Programa	Usuario S.O.	Ordenador S.O.	Usuario Soporte
06/05/2022 10:58:07	EDISA	ALRT_ALERTAS	elias	portefmmac	
06/05/2022 10:54:39	EDISA	SUPERUSUARIO			
06/05/2022 10:53:43	EDISA	SUPERUSUARIO			
06/05/2022 10:48:17	EDISA	SUPERUSUARIO			
06/05/2022 10:48:14	EDISA	SUPERUSUARIO			
06/05/2022 10:47:33	EDISA	SUPERUSUARIO			
06/05/2022 10:47:28	EDISA	SUPERUSUARIO			
04/05/2022 14:17:49	EDISA	SUPERUSUARIO			
04/05/2022 13:52:37	EDISA	SUPERUSUARIO			
04/05/2022 07:00:36	EDISA	SUPERUSUARIO			
03/05/2022 13:52:35	EDISA	SUPERUSUARIO			
03/05/2022 13:49:46	EDISA	SUPERUSUARIO			
03/05/2022 13:49:33	EDISA	SUPERUSUARIO			
02/05/2022 12:05:09	EDISA	SUPERUSUARIO			
02/05/2022 10:10:46	EDISA	SUPERUSUARIO			

Servidores SMTP

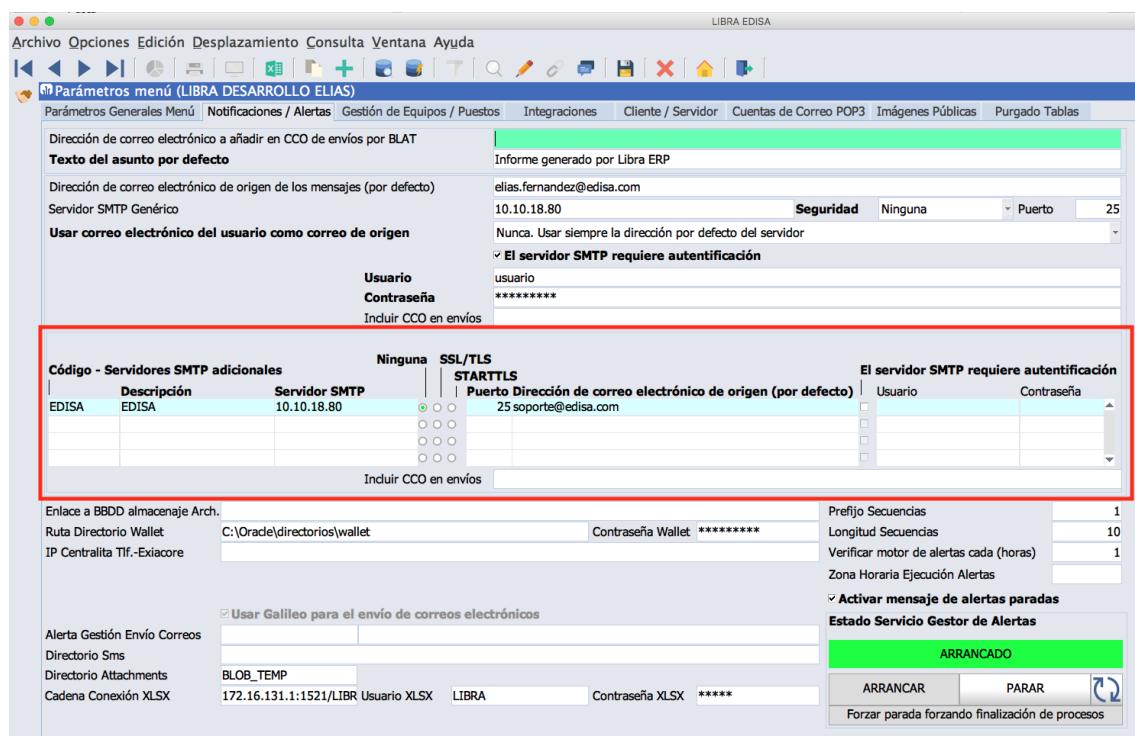
Es recomendable configurar un servidor SMTP genérico que será usado cuando en las alertas no se defina uno específico.



- **Dirección de correo electrónico de origen de los mensajes (por defecto):** El que recibe el mensaje de correo electrónico verá esta dirección de correo electrónico como remitente del mensaje. En las alertas se puede especificar otra, la especificada en este campo será usada en el caso de que en la alerta no se especifique una.
- **Servidor SMTP Genérico:** Dirección IP o nombre del servidor SMTP.
- **Puerto:** Puerto TCP en el que escucha el servidor SMTP.
- **El servidor SMTP requiere autentificación:** Se debe de marcar en el caso de que el servidor SMTP requiera usuario y contraseña, en caso de activarse deberán de cubrirse los campos “Usuario” y “Password”.
- **Incluir CCO en envíos:** Si se indica una dirección de correo en todos los envíos se le añadirá de forma automática esa cuenta de correo de forma oculta. Esta opción es útil cuando se está monitorizando el sistema.

Servidores SMTP Adicionales

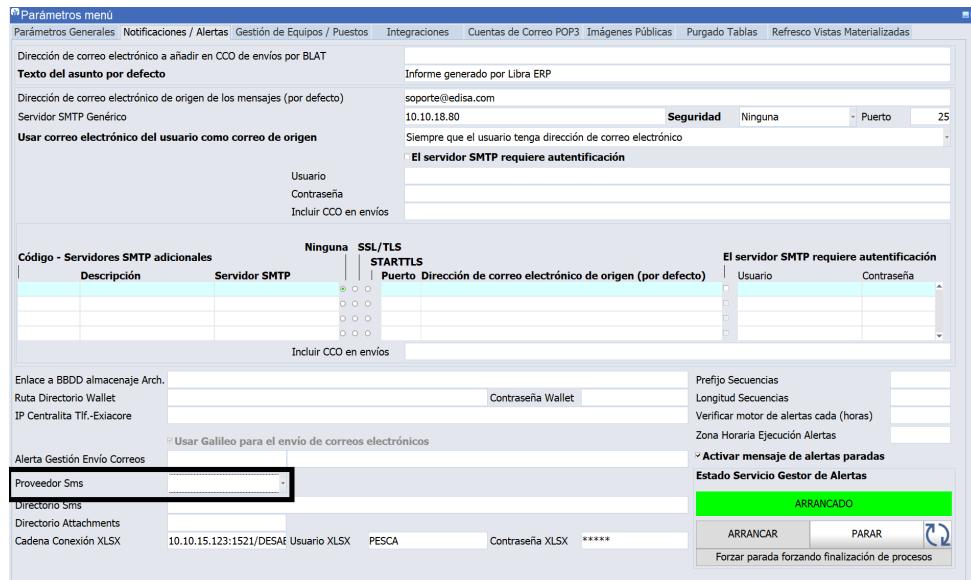
En cada alerta se puede indicar que los correos electrónicos generados se tramiten a través de un determinado servidor SMTP distinto al genérico. El servidor debe de encontrarse parametrizado en los parámetros generales de Notificaciones / Alertas. Los campos tienen el mismo significado / funcionamiento que el explicado en el servidor SMTP Genérico.



Proveedor Envío SMS

Para que una alerta pueda realizar envíos de SMS es necesario parametrizar el “Proveedor Sms” en la pestaña “Notificaciones / Alertas” desde el programa “Parámetros de Menú” [U_MPRMEN].

Una vez seleccionado un proveedor, se activarán Plugins asociados al mismo, así como otros campos de configuración si son necesarios.

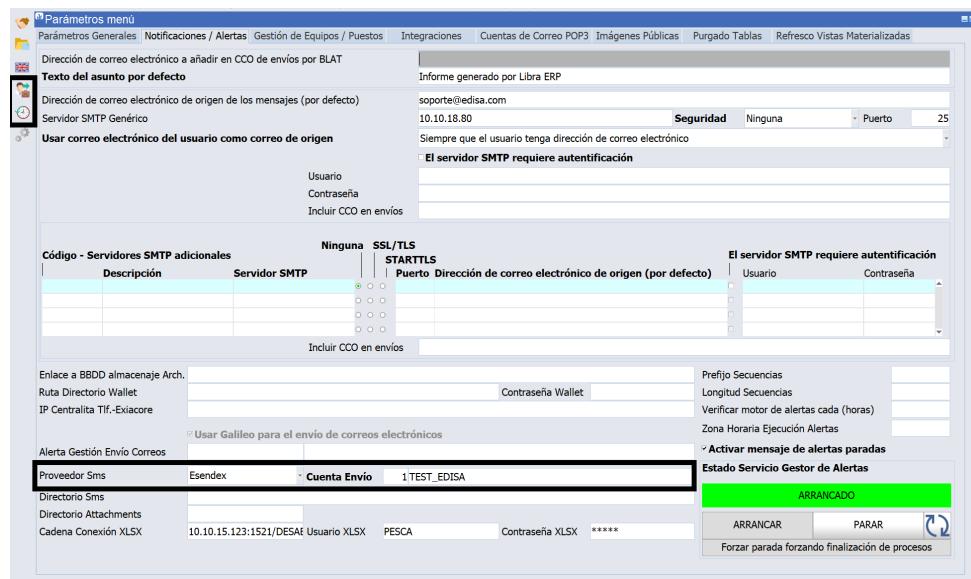


The screenshot shows the 'Parámetros menú' window with the 'Notificaciones / Alertas' tab selected. In the 'Alerta Gestión Envío Correos' section, the 'Proveedor Sms' provider is chosen. The 'Cuenta Envío' field is highlighted in red, indicating it is the current focus or has an error.

Esendex

Tras seleccionar Esendex se habilitará un nuevo campo “Cuenta Envío” para indicar que cuenta se utilizará para el envío. Una cuenta de Esendex se corresponde con los datos de autorización ante el servicio remitente, así como la cuota de SMS disponibles.

En la región de Plug-Ins verticales, aparecerán dos nuevos iconos desde el cual se podrán configurar las cuentas y el registro de mensajes enviados.

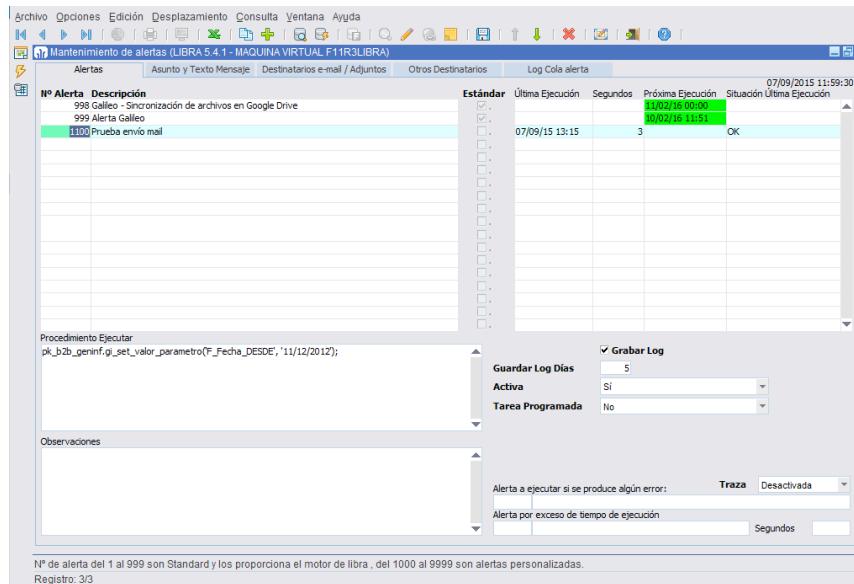


The screenshot shows the 'Parámetros menú' window with the 'Notificaciones / Alertas' tab selected. In the 'Alerta Gestión Envío Correos' section, the 'Proveedor Sms' provider is chosen. The 'Cuenta Envío' field is highlighted in red, indicating it is the current focus or has an error.

Crear una alerta

Las alertas se definen en el mantenimiento de alertas (Programa ALRT_ALERTAS)

Datos básicos



- **Nº Alerta:** Los números de alerta del 1 al 999 están reservados para alertas internas de Libra, por lo tanto, se pueden codificar alertas personalizadas a partir del número 1000.
- **Descripción:** Nombre que se le asigna a la alerta.
- **Estándar:** Estará activado únicamente en alertas que son distribuidas con Libra (números de alerta entre el 1 y 999).
- **Activa:** Determina si debe o no de ejecutarse: Valores posibles:
 - **Sí:** Se ejecuta.
 - **No:** No se ejecuta nunca.
 - **Sí - Excepto en BBDD de pruebas:** Si la base de datos en donde se ejecuta Libra tiene una licencia de pruebas, la alerta no se ejecutará.
- **Última Ejecución:** Es un campo no modificable por el usuario e informa de la fecha y hora en la que se ha ejecutado por última vez la alerta.
- **Segundos:** Tiempo en segundos que ha tardado la última ejecución.
- **Próxima Ejecución:** Es un campo no modificable por el usuario y en el caso de tareas programadas se informa de la fecha y hora en la que será ejecutada la alerta.
- **Situación Última Ejecución:** Es un campo no modificable por el usuario, en el caso de que la alerta fuese ejecutada alguna vez, informa del resultado de la última ejecución. Mostrará OK en caso de que un resultado correcto o ERRORES en el caso de haberse producido algún fallo, en ese caso, en la pestaña "Log cola alerta" se puede ver el motivo del error.
- **Procedimiento Ejecutar:** Código PL/SQL que ejecuta la alerta, en este código PL/SQL se puede realizar cualquier tipo de proceso que pueda ejecutarse en la base de datos, además también se puede cambiar cualquier propiedad de la alerta que se está ejecutando. Ver apartado "Gestión de la alerta a través del PL/SQL." para más información.
- **Observaciones:** Campo de uso libre para poder dejar ahí información sobre el funcionamiento de la alerta, su objetivo, quien la hizo, ...
- **Grabar Log:** Si se activa esta check cada vez que se ejecute la alerta quedará registrada la ejecución, si no se marca la check únicamente se registra la ejecución en el caso de que se produzca algún error.
- **Guardar Log Días:** Número de días que se guardará el log de las ejecuciones de la alerta.

- **Tarea Programa:** Si se activa la opción de Tarea Programada, en cualquiera de sus modalidades, se puede indicar la periodicidad con la que se ha de ejecutar la alerta de forma automática.
 - **Modalidades de Tareas Programadas:**
 - **No:** No es una tarea programada, su ejecución tiene que invocarse a través de un código PL/SQL.
 - **Sí:** Se ejecutará mediante la cola compartida para todas las alertas.
 - **Sí - Ejecución Tiempo Real:** Cada alerta de este tipo tiene su propio arranque o parada, por lo que el motor de alertas podría estar parado pero las alertas de tiempo real continuar con su ejecución programada de forma independiente. Este tipo está pensado para aquellas alertas que tengan una larga duración, o una baja frecuencia de ejecución, pero que su ejecución a una determinada hora sea crítico, por lo que **no se permite** indicar ejecución cada x minutos o Siempre, hay que indicar cada x horas, días o meses. **IMPORTANTE:** Las alertas de tiempo real se paran y arrancan de forma automática con el motor de alertas, si para una alerta en concreto no se quiere este comportamiento hay que activar la check **“Bloquear Arranque / Parada con motor de Alertas”**.
 - **Sí - Generación de Reports:** Alertas destinadas a realizar una distribución de informes de “Oracle Reports” a múltiples usuarios, para ello se basa en indicar un informe de “Generador de Informes” que devuelva la lista de correos a los que enviar la alerta, los parámetros de generación, y los datos a incluir en el correo electrónico. Para más información ver el apéndice “Generación de Reports y envío por Alerta”.
 - **Sí - Generación de Reports – Tiempo Real:** Igual que “Sí – Generación de Reports” con ejecución igual que las de “Sí – Ejecución Tiempo Real”, es decir, no se ponen en cola, cuando llega la hora programada a su ejecución se ejecutan directamente.
 - **Sí - Ejecución Remota:** Será ignorada por el motor de alertas, se utiliza en módulo de sincronización de movilidad. Consultar específica de ese módulo para más información.
 - **Ejecutar cada:** Periodicidad con la que se ejecuta la alerta:
 - **Siempre:** Se ejecuta en cada ciclo de proceso de las alertas, por tanto, cada 15 segundos (siempre y cuando el motor no esté ejecutando una alerta con mayor duración). Se pueden indicar los días de la semana que se ejecutará, y desde que hora y hasta que hora se ha de ejecutar.
 - **Segundos:** Únicamente se puede utilizar en alertas de tipo “Tiempo Real”. La alerta se ejecutará en ciclos de los segundos indicados en “Ejecutar Cada”. Al ser únicamente para alertas de tiempo real, no se verá afectada por otras alertas.
 - **Minutos:** Funcionamiento similar al de “Siempre”, pero se puede espaciar los minutos indicados entre una ejecución de otra.
 - **Horas:** Funcionamiento similar al de “Siempre”, pero se puede espaciar las horas indicadas entre una ejecución y otra.
 - **Días:** Se puede espaciar una ejecución de otra en los días indicados, también se pueden indicar los días de la semana en los que se ha de ejecutar. Será necesario indicar a qué hora del día se debe de ejecutar cubriendo el campo “Ejecutar a esta hora”. En el campo “Ejecutar el día” se pueden indicar separados por comas los días en los que se debe de ejecutar (si se pone un número negativo hace el desplazamiento sobre el último día del mes). Si se utiliza esta opción desaparecerán las checks de los días de la semana ya que son incompatibles.
 - **Meses:** Se puede espaciar una ejecución de otra en los meses indicados. Se puede indicar qué día del mes se ha de ejecutar cubriendo el campo “Ejecutar el día”. En el caso de indicar el número de día en negativo se considera un desplazamiento sobre el último día del mes, por tanto, para ejecutar una alerta el último día del mes se pondrá -1, para el penúltimo -2, etc. También se puede indicar la hora en la que se debe de ejecutar cubriendo el campo “Ejecutar a esta hora”.

- **Traza:** Permite forzar que se registren en LIBRA_LOG las llamadas a pkpantallas.traza, según el nivel que se indique. El nivel "Errores" registra todos los pkpantallas.traza que no pasen por parámetro el nivel de traza, mientras que el "Todos" registra incluso todas las trazas de paquetes de entorno.

En la tabla LIBRA_LOG quedarán los registros asociados a la ejecución de la alerta mediante los campos NUMERO_ALERTA e ID_COLA_ALERTA.

Desde el mantenimiento de alertas en la pestaña "Log Cola Alerta" se puede descargar la traza a un archivo usando el botón "Exportar LOG" de la botonera vertical.

Al entrar en Libra un superusuario y hay alguna alerta con la traza activada, se le mostrará un mensaje, ya que esté funcionando una alerta programada de forma desatendida generando trazas puede generar una gran cantidad de registros en la tabla LIBRA_LOG.

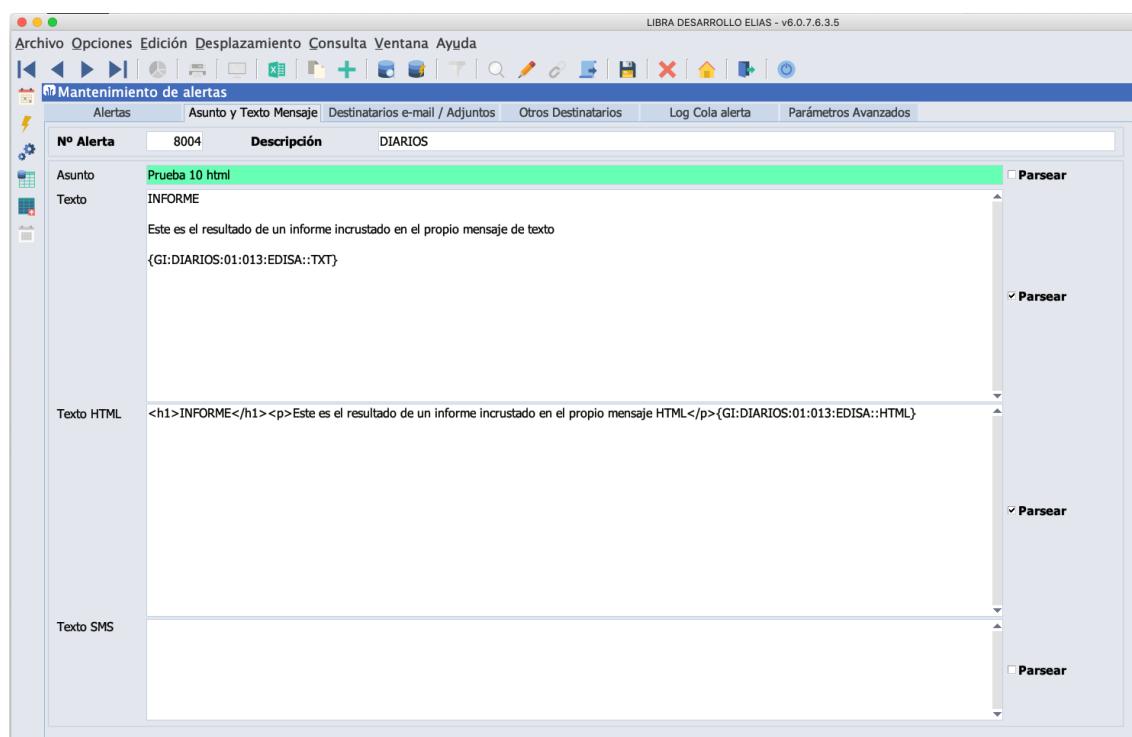
- **Alerta a ejecutar si se produce algún error:** Número de alerta que se ejecuta cuando el resultado de la ejecución no es OK. Será una alerta destinada a los administradores del sistema para que solucionen el problema.
- **Alerta por exceso de tiempo de ejecución:** Número de alerta que se ejecuta cuando el tiempo de la última ejecución excede un umbral, este umbral se indicará en el campo "Segundos" que se encuentra a la derecha.

Parámetros Avanzados

- **NLS_NUMERIC_CHARACTERS:** Permite configurar por cada alerta con qué configuración debe de ejecutarse:
 - .,: Separador decimal , y separador de millares .
 - .,: Separador decimal . y separador de millares ,

Plantilla de notificación

En la pestaña "Asunto y Texto Mensaje" se usan para definir el texto que irá en la notificación que se realiza al usuario.



Cualquiera de los campos de esta pestaña puede ser alterados en tiempo de ejecución de la alerta desde el código PL/SQL. También se puede aplicar el PARSER a cada uno de los campos, para activar el PARSER simplemente hay que activar la check “Parsear” que se encuentra a la derecha de cada campo.

- **Asunto:** Asunto del correo electrónico o del mensaje al menú de Libra de los usuarios.
- **Texto:** Texto en formato plano del correo electrónico o del mensaje al menú de Libra. Si se activa “Parsear” se puede incrustar el resultado de un informe del generador de informes introduciendo dentro del texto la siguiente cadena: {GI:<INFORME>:<IDIOMA>:<EMPRESA>:<USUARIO>:<PLANTILLA DE VALORES DE FILTRO>:**txt**:<CODIGO CONFIGURACIÓN>}

Ejemplo:

INFORME

Este es el resultado de un informe incrustado en el propio mensaje de texto
{GI:DIARIOS:01:013:EDISA::txt;}

- **Texto HTML:** Texto HTML del correo electrónico. Si se activa “Parsear” se puede incrustar el resultado de un informe del generador de informes introduciendo dentro del texto la siguiente cadena: {GI:<INFORME>:<IDIOMA>:<EMPRESA>:<USUARIO>:<PLANTILLA DE VALORES DE FILTRO>:**HTML**:<CODIGO CONFIGURACION>}

Ejemplo: <h1>INFORME</h1><p>Este es el resultado de un informe incrustado en el propio mensaje HTML</p>{GI:DIARIOS:01:013:EDISA::HTML;}

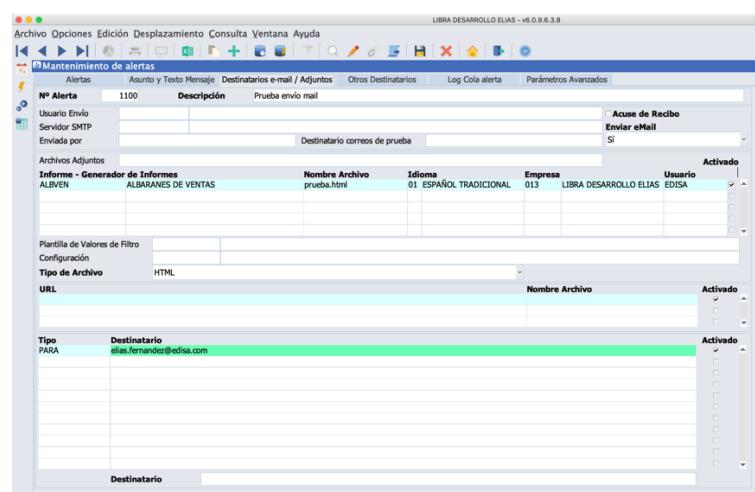
- **Texto SMS:** Texto a enviar en el mensaje SMS. En la pestaña “Otros Destinatarios” se indicarán los números de teléfono a los que hacer el envío. Estos destinatarios se pueden gestionar desde el PL/SQL de la alerta. **NOTA:** Requiere configuración de envío de SMS en parámetros generales de menú.

En una notificación por correo electrónico donde esté parametrizado un texto plano y un texto HTML, en el correo electrónico irán los dos marcándolos como alternativos, es decir, dependiendo del dispositivo que abra el mensaje, si puede renderizar HTML se mostrará en HTML y en caso de no poder renderizar el HTML se mostrará en texto plano.

Destinatarios de correo electrónico y Adjuntos

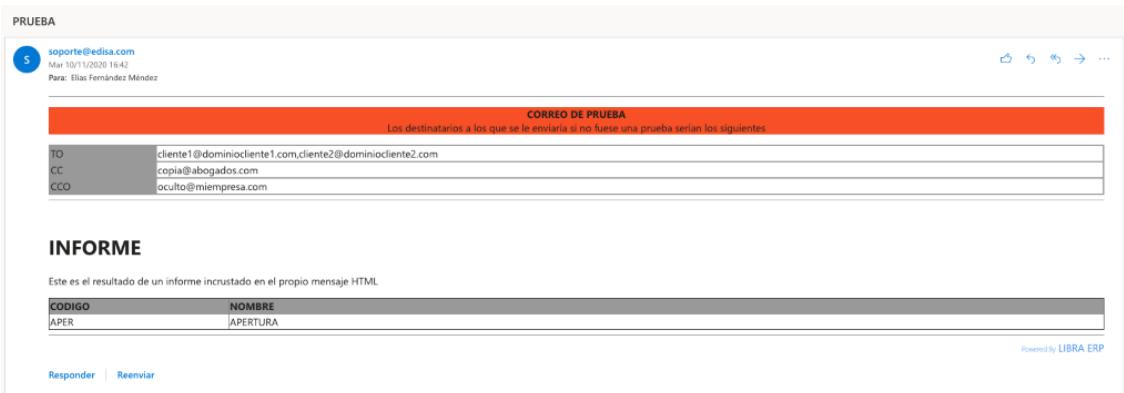
En la pestaña Destinatarios e-mail / Adjuntos se pueden indicar los adjuntos que llevará el correo electrónico y las direcciones a las que debe de enviarse la notificación.

Cualquiera de los campos indicados en esta pestaña también puede ser modificado desde el código PL/SQL de la alerta.



- **Usuario Envío:** Permite indicar que el envío se realice según la parametrización de servidor SMTP que tenga un determinado usuario.
- **Servidor SMTP:** Permite forzar que la alerta salga por un determinado servidor SMTP, en caso de dejarlo en blanco la alerta será enviada a través del servidor SMTP genérico. Se explica la configuración de los servidores SMTP en el apartado “Parametrización”
- **Enviada por:** Dirección de correo electrónico que envía la alerta, es decir, la dirección de remite del mensaje que verá el destinatario de este y a la que se responderá en caso de que el destinatario use la opción de responder. En caso de dejarla en blanco usará la “Dirección de correo electrónico de origen” del servidor SMTP indicado o del genérico en el caso de no indicar un servidor SMTP.
- **Destinatario correos de prueba:** Cuando se desarrolla una alerta y hay envíos a múltiples destinatarios a veces es complicado validarla ya que no se ven los correos de los destinatarios y por tanto su contenido y adjuntos. Para solucionar este inconveniente se ha incorporado la posibilidad de indicar un correo de prueba en la alerta, con ese correo indicado, todos los correos que se originen durante la ejecución de esa alerta se redirigen a esa dirección de correo y en el cuerpo del correo se inyecta una cabecera indicando a quien hubiese llegado ese correo.
- **Acuse de recibo:** Se marcará el correo electrónico con solicitud de confirmación de lectura.
- **Enviar eMail:** Puede contener los siguientes valores (pudiéndose alterar en el PL/SQL de la alerta):
 - **Sí:** El correo electrónico parametrizado será enviado.
 - **No:** El correo electrónico parametrizado no se envía.
 - **Sólo si hay informes con datos:** Únicamente se enviará el correo si la alerta ejecuta algún informe del generador de informes que devuelva datos, tanto si es incluido como archivo adjunto o incrustado dentro del cuerpo del correo electrónico.

Ejemplo de correo enviado con la “Destinatario correos de prueba” cubierto:



PRUEBA

soporte@edisa.com
Mar 10/11/2020 16:42
Para: Elias Fernández Méndez

CORREO DE PRUEBA
Los destinatarios a los que se le enviaría si no fuese una prueba serían los siguientes

TO	cliente1@dominiocliente1.com,cliente2@dominiocliente2.com
CC	copia@abogados.com
CCO	oculto@miempresa.com

INFORME

Este es el resultado de un informe incrustado en el propio mensaje HTML.

CODIGO	NOMBRE
APER	APERTURA

Powered by LIBRA ERP

Responder Reenviar

Adjuntos.

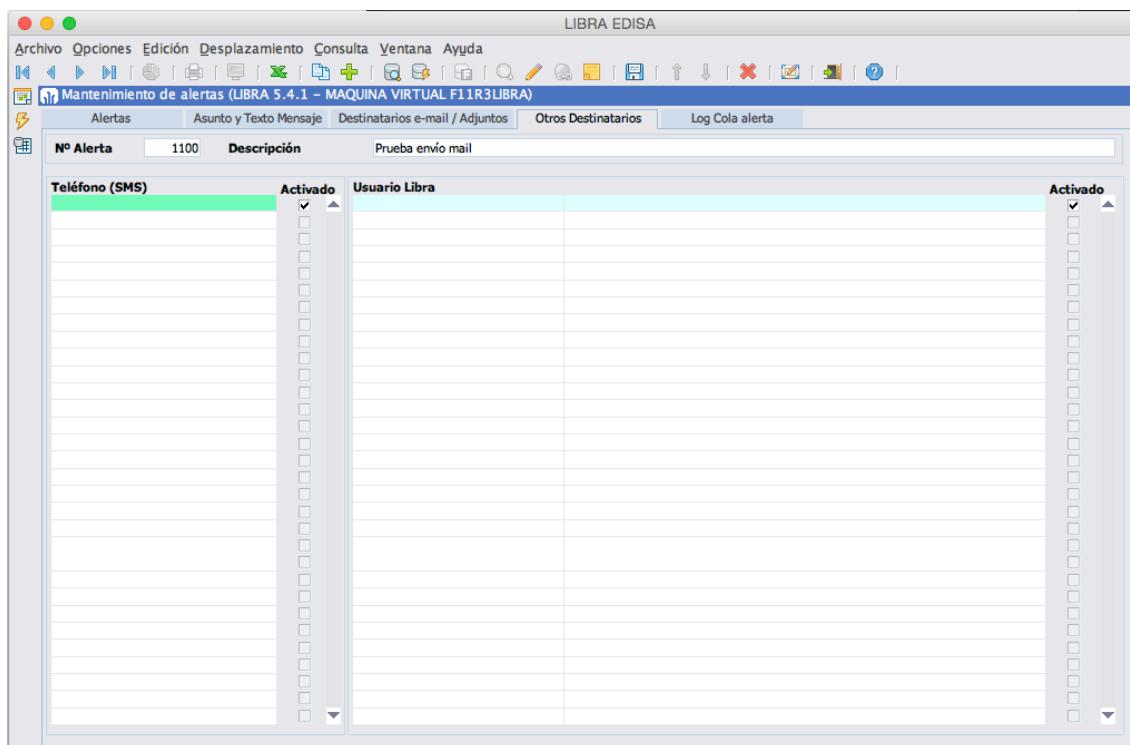
- **Archivos Adjuntos:** Lista de archivos separados por comas que serán adjuntados en el correo electrónico, los archivos pueden ser generados desde el propio código pl/sql de la alerta.
- **Informe – Generador de Informes:** En el correo electrónico se pueden adjuntar archivos generados por el generador de informes de Libra:
 - **Informe:** Código del informe a ejecutar.
 - **Nombre Archivo:** Nombre del archivo que contendrá el resultado del informe. Si el archivo es una hoja de cálculo y se desea enviar en formato XLS o XLSX nativo, hay que configurar el servicio GAL_EXCEL de Galileo. Si no se dispone de este servicio, hay que indicar la extensión XML. En el caso de XML la hoja de cálculo únicamente podrá ser abierta con Microsoft Office o con LibreOffice. Manual de instalación de GAL_EXCEL en sección de apéndices.
 - **Idioma:** Código de idioma con el que se ejecutará el informe, con el fin de obtener las etiquetas.
 - **Empresa:** Código de la empresa de Libra en la que se ejecutará el informe.
 - **Usuario:** Código del usuario de Libra con el que se ejecutará el informe, es importante ya que en base al usuario se determinan los permisos de los datos a exportar.
 - **Activado:** Si no se encuentra activada, el adjunto será ignorado.
 - **Plantilla de Valores de Filtro:** En el generador de informes, en la pantalla de filtros se puede guardar una plantilla con los valores de los filtros, el código asignado a esa plantilla se puede indicar en este campo para que el informe se ejecute con esos filtros aplicados.
 - **Configuración:** Si un informe tiene varias configuraciones de columnas, se puede indicar con qué configuración se quiere generar el informe.
 - **Tipo de Archivo:** Puede indicarse uno de los siguientes tipos:
 - Hoja de Cálculo
 - HTML
 - Texto Plano
 - Texto separado por comas (CSV).
 - **Codificación:** Cuando en Tipo de Archivo se indica “Texto Plano” o “Texto separado por comas (CSV)” se puede indicar el tipo de codificación del archivo. Las codificaciones posibles son (Microsoft Excel para que abra un CSV tienen que ser codificados en ANSI o UTF+BOM):
 - **ANSI**
 - **UTF8**
 - **UTF8+BOM**
- **Archivos descargados de direcciones web:** Se permite descargar archivos de una dirección web y ser incluidos en el correo electrónico. Mediante esta funcionalidad se pueden incluir informes de Reports cuando exista un servidor de Reports de Forms 12C.
- **URL:** Dirección web de la que descargará el archivo.
- **Nombre Archivo:** Nombre que se le dará al archivo al ser adjuntado al correo electrónico.
- **Activado:** Si no se encuentra activada, el adjunto será ignorado.

Destinatarios de correo electrónico.

Lista de correos electrónicos a los que se enviará el mensaje:

- **Tipo:** Puede tener los siguientes valores:
 - **PARA:** Dirección de correo electrónico de los destinatarios principales del mensaje.
 - **CC:** Dirección de correo electrónico de los destinatarios con copia.
 - **CCO:** Dirección de correo electrónico de los destinatarios con copia Oculta.
 - **PARA_USU, CC_USU, CCO_USU:** El campo "Destinatario" se validará contra la lista de valores de USUARIOS y el correo electrónico se enviará al correo que tenga configurado el usuario en la ficha en el momento de la ejecución de la alerta, siempre que en el momento de la ejecución el usuario no esté de baja.
 - **PARA_SU, CC_SU, CCO_SU:** El correo electrónico se envía a todos los usuarios que tengan correo electrónico, que no estén de baja y que sean superusuario. En este caso el campo "Destinatario" no se introduce.
 - **PARA_PER, CC_PER, CCO_SU:** El campo "Destinatario" se validará contra la lista de valores de PERFILES y el correo electrónico se enviará a todos los usuarios que pertenezcan al perfil en el momento de la ejecución de la alerta y tengan cubierto el correo electrónico en su ficha y no estén de baja.
- **Destinatario:** Dirección de correo electrónico.
- **Activado:** Si no se encuentra activada, el destinatario será ignorado.

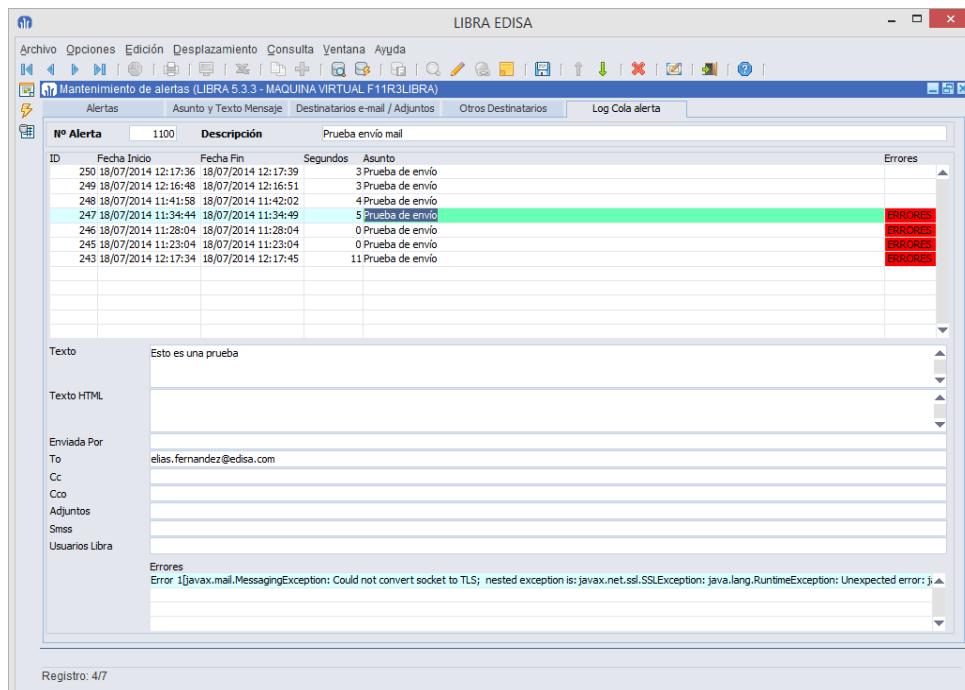
Otros Destinatarios.



- **Teléfono (SMS):** Lista de teléfonos a los que enviar la alerta por SMS. Es posible su envío de forma automática tras configurar el proveedor de Envío de SMS. En caso contrario, únicamente se utiliza como almacenamiento de los números de teléfono y su envío se debe de gestionar desde el PL/SQL de la alerta.
- **Usuario Libra:** Usuarios de Libra a los que se les enviará la notificación como mensaje a su menú.
- **Activado:** Si no se encuentra activada, el destinatario será ignorado.

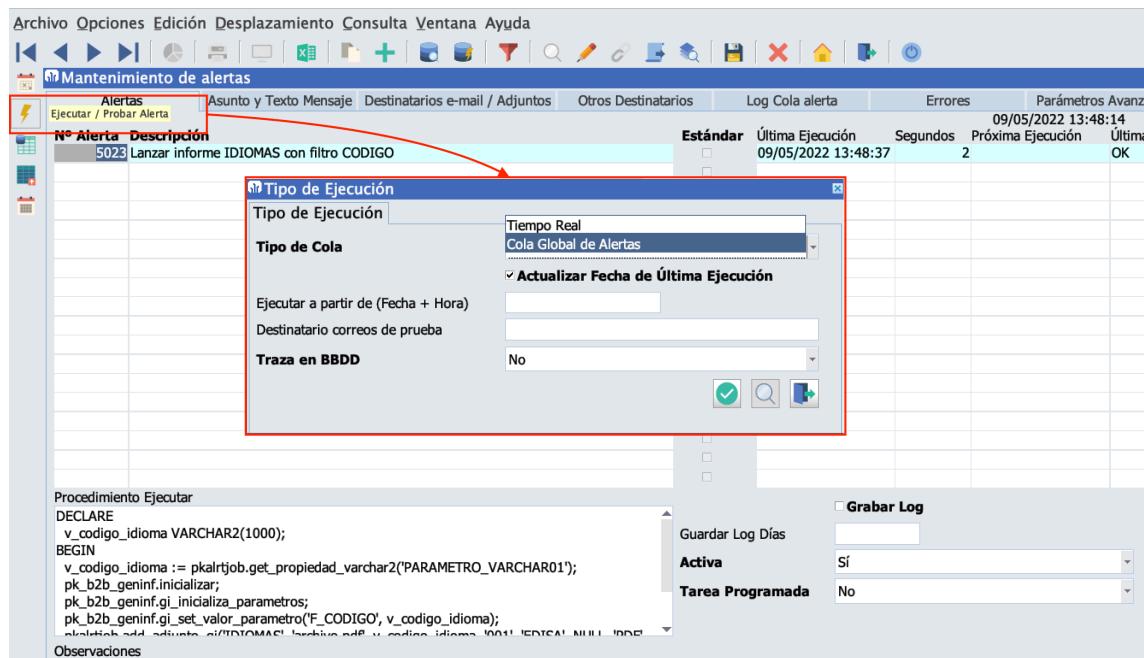
Log Alerta

En la pestaña “Log Cola Alerta” se pueden consultar las ejecuciones de la alerta, en el caso de que la alerta tenga activada la check “Guardar Log”, en el caso de que no esté activada únicamente se mostrarán aquellas ejecuciones que no se han realizado correctamente.



Ejecutar / Probar tarea

Con el plugin “Ejecutar / Probar tarea ahora” abrirá una ventana para seleccionar el modo en que se va a poner en cola la alerta y también permite ejecutar la alerta sin que se actualice la fecha de última ejecución y por tanto no cambiar la planificación de la ejecución.



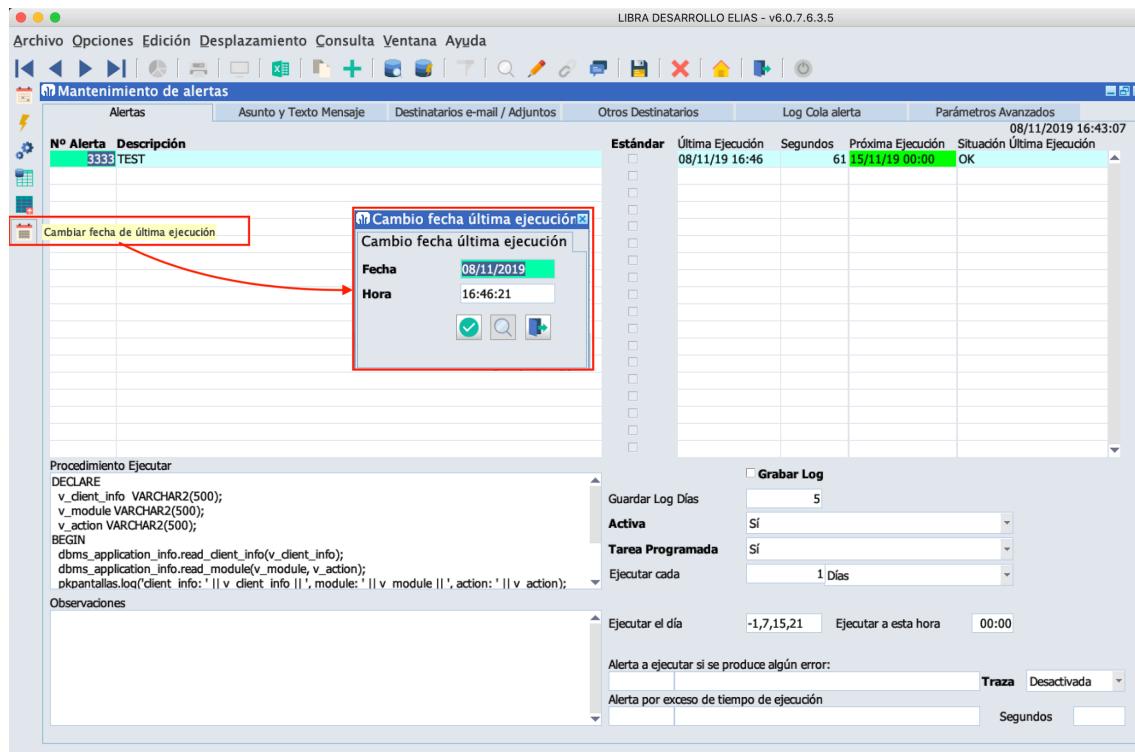
- **Tipo de Cola:** Por defecto va a proponer la cola en base a la configuración de la alerta, pero para esa ejecución se puede elegir libremente la cola a usar, es decir, una tarea

programada que no es de tiempo real se puede ejecutar como tiempo real para evitar que bloquee a otras alertas.

- **Actualizar Fecha de Última Ejecución:** El desactivar esta check tiene el mismo comportamiento que tenía el plug-in de "Probar tarea ahora", es decir, ejecuta la tarea, pero, no cambia la fecha de última ejecución por lo que se mantiene intacto el calendario previsto de ejecuciones. Si se desactiva esta check va a cargar en "Destinatario correos de prueba" la dirección de correo electrónico del usuario actual.
- **Ejecutar a partir de (Fecha + Hora):** Permite diferir la ejecución al momento que se indique. Si el "Tipo de Cola" es de tipo "Tiempo Real" se ejecutará a esa hora, mientras si es de tipo "Cola Global de Alertas" se ejecutará en algún momento después de esa hora cuando la cola quede libre.
- **Destinatario correos de prueba:** Si se cubre, todos los correos que se envíen por la ejecución de la alerta se mandarán a esa dirección de correo electrónico. Si no se indica, pero en la alerta tiene cubierto el campo "Destinatarios correos de prueba" se mandarán a la dirección que se indique en la alerta.
- **Traza en BBDD:** Permite definir el tipo de captura de trazas de rendimiento de Oracle para esa ejecución de la alerta.
 - **No:** No se capturan trazas.
 - **Perfilador de Código:** Se activa el análisis de tiempo y número de veces que se ejecuta cada línea de código. Para ver el resultado de este análisis después de haberse ejecutado la alerta, hay que ir a la pestaña "Log Cola alerta" y usar el plug-in "Perfilador de Código".
 - **SQL Trace – TKPROF:** Se activa el análisis de la ejecución de consultas en base de datos. Este análisis hay que procesarlo con la herramienta TKPROF de Oracle. Desde Libra se puede hacer desde el "Administrador de Archivos" (programa U_ADMINFILES), en la pestaña "Archivos Base de Datos", seleccionando el directorio BBDD_TRACE. Los archivos de traza de alertas comienzan por A y en el nombre llevan el número de alerta y el identificador de la cola. Para procesar ese archivo de traza hay que usar el plug-in Generar TKPROF.

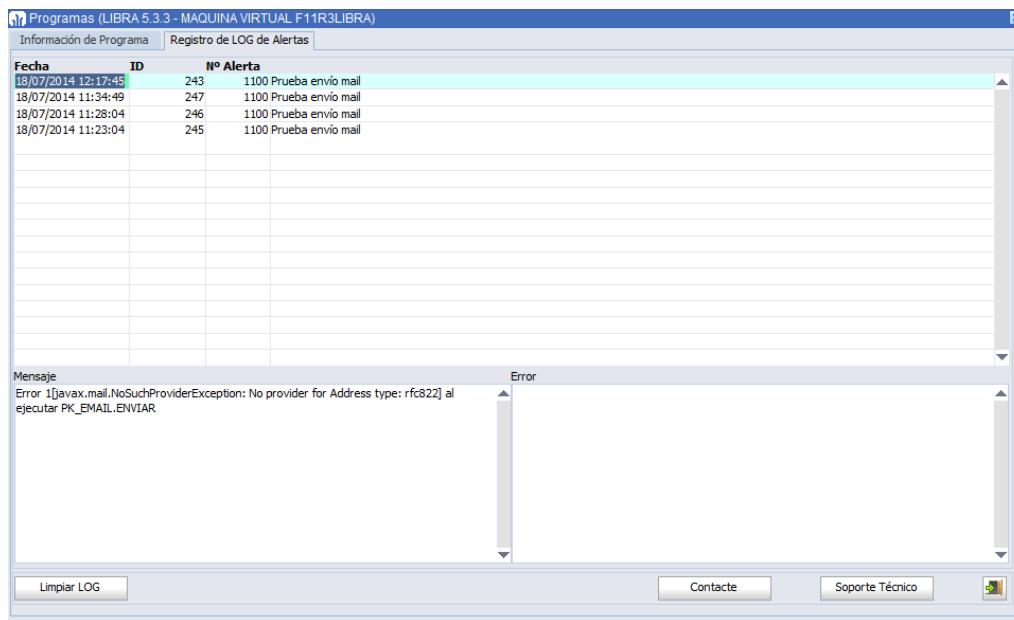
Gestión de fecha de última ejecución

Hay un plug-in para poder modificar la fecha de última ejecución de una alerta y esa forma forzar el cambio de fecha de próxima ejecución.



Consultar errores de las alertas

En la pantalla “Acerca de...” de cualquier programa aparece una pestaña “Incidencias en Alertas” que mostrará aquellas alertas que no se han ejecutado correctamente:



Invocar una alerta desde un código PL/SQL

Una alerta que no es una tarea programada, para que se ejecute debe de invocarse a través de un código PL/SQL.

Para ser invocada se realiza a través del procedimiento PKALRT.GRABAR_COLA_ALERTAS que tiene los siguientes parámetros:

- **p_numero_alerta:** Es el único parámetro obligatorio, indica el número de alerta a ejecutar.
- **p_asunto_alerta:** Texto del asunto de la alerta.
- **p_texto_alerta:** Texto plano de la alerta.
- **p_enviada_por:** Dirección de correo electrónico del remitente.
- **p_procedimiento_ejecutar:** Código pl/sql a ejecutar.
- **p_notificar_mail_to:** Lista de direcciones de correo electrónico de destino (separadas por comas) de la notificación de la alerta.
- **p_notificar_mail_cc:** Lista de direcciones de correo electrónico en copia (separadas por comas) de la notificación de la alerta.
- **p_notificar_mail_cco:** Lista de direcciones de correo electrónico en copia oculta (separadas por comas) de la notificación de la alerta.
- **p_notificar_mail_attachments:** Lista de archivos adjuntos (separados por comas). Se puede indicar mediante este mismo parámetro que se anexen al correo archivos que deben de ser descargados desde una url, para ello hay que indicar el archivo con el siguiente formato:
 - URL:nombre_archivo:dirección_URL
 - Ejemplo: URL:logo.png:<http://www.edisa.com/wp-content/uploads/2014/03/edisa1.header-logo-regular.png>
- **p_notificar_mail_acuse_recibo:** Pasar S en el caso de solicitar confirmación de lectura de la notificación.
- **p_notificar_smss:** Lista de teléfonos (separados por comas) a enviar la notificación por SMS.
- **p_notificar_usuarios_libra:** Lista de usuarios (separados por comas) de Libra a los que enviar la notificación al menú.
- **p_texto_alerta_html_clob:** Texto en formato HTML del correo electrónico.
- **p_notificar_mail_id_archivos:** Lista de archivos (separados por comas) de ARCHIVOS_ERP a ser adjuntados al correo electrónico.
- **p_texto_alerta_html_sms:** Texto del SMS.
- **p_parametro_varchar01:** Parámetro alfanumérico de uso Libre, puede ser usado en el PL/SQL de la alerta mediante PKALRTJOB.GET_PROPIEDAD_VARCHAR2('PARAMETRO_VARCHAR01');
- **p_parametro_varchar02:** Parámetro alfanumérico de uso Libre, puede ser usado en el PL/SQL de la alerta mediante PKALRTJOB.GET_PROPIEDAD_VARCHAR2('PARAMETRO_VARCHAR02');
- **p_parametro_varchar03:** Parámetro alfanumérico de uso Libre, puede ser usado en el PL/SQL de la alerta mediante PKALRTJOB.GET_PROPIEDAD_VARCHAR2('PARAMETRO_VARCHAR03');
- **p_parametro_varchar04:** Parámetro alfanumérico de uso Libre, puede ser usado en el PL/SQL de la alerta mediante PKALRTJOB.GET_PROPIEDAD_VARCHAR2('PARAMETRO_VARCHAR04');
- **p_parametro_varchar05:** Parámetro alfanumérico de uso Libre, puede ser usado en el PL/SQL de la alerta mediante PKALRTJOB.GET_PROPIEDAD_VARCHAR2('PARAMETRO_VARCHAR05');
- **p_parametro_number01:** Parámetro numérico de uso Libre, puede ser usado en el PL/SQL de la alerta mediante PKALRTJOB.GET_PROPIEDAD_NUMBER('PARAMETRO_NUMBER01');
- **p_parametro_number02:** Parámetro numérico de uso Libre, puede ser usado en el PL/SQL de la alerta mediante PKALRTJOB.GET_PROPIEDAD_NUMBER('PARAMETRO_NUMBER02');
- **p_parametro_number03:** Parámetro numérico de uso Libre, puede ser usado en el PL/SQL de la alerta mediante PKALRTJOB.GET_PROPIEDAD_NUMBER('PARAMETRO_NUMBER03');
- **p_parametro_number04:** Parámetro numérico de uso Libre, puede ser usado en el PL/SQL de la alerta mediante PKALRTJOB.GET_PROPIEDAD_NUMBER('PARAMETRO_NUMBER04');

- **p_parametro_number05:** Parámetro numérico de uso Libre, puede ser usado en el PL/SQL de la alerta mediante PKALRTJOB.GET_PROPIEDAD_NUMBER('PARAMETRO_NUMBER05');
- **p_parametro_date01:** Parámetro de tipo fecha de uso Libre, puede ser usado en el PL/SQL de la alerta mediante PKALRTJOB.GET_PROPIEDAD_DATE('PARAMETRO_DATE01');
- **p_parametro_date02:** Parámetro de tipo fecha de uso Libre, puede ser usado en el PL/SQL de la alerta mediante PKALRTJOB.GET_PROPIEDAD_DATE('PARAMETRO_DATE02');
- **p_parametro_date03:** Parámetro de tipo fecha de uso Libre, puede ser usado en el PL/SQL de la alerta mediante PKALRTJOB.GET_PROPIEDAD_DATE('PARAMETRO_DATE03');
- **p_parametro_date04:** Parámetro de tipo fecha de uso Libre, puede ser usado en el PL/SQL de la alerta mediante PKALRTJOB.GET_PROPIEDAD_DATE('PARAMETRO_DATE04');
- **p_parametro_date05:** Parámetro de tipo fecha de uso Libre, puede ser usado en el PL/SQL de la alerta mediante PKALRTJOB.GET_PROPIEDAD_DATE('PARAMETRO_DATE05');
- **p_enviar_email:** Permite modificar el parámetro "Enviar eMail" de la alerta, puede pasarse uno de los siguientes valores:
 - S: Sí.
 - N: No.
 - I: Sólo si hay informes con datos.
- **p_aplicar_parser_asunto:** Permite modificar el parámetro "Parsear" del Asunto, se puede pasar los valores S ó N.
- **p_aplicar_parser_texto:** Permite modificar el parámetro "Parsear" del cuerpo del mensaje en formato texto plano, se puede pasar los valores S ó N.
- **p_aplicar_parser_texto_html:** Permite modificar el parámetro "Parsear" del cuerpo del mensaje en formato HTML, se puede pasar los valores S ó N.
- **p_aplicar_parser_texto_sms:** Permite modificar el parámetro "Parsear" del texto SMS, se puede pasar los valores S ó N.
- **p_fecha_ejecucion:** Fecha y hora a partir de la que debe de ser ejecutada. Mediante este parámetro se puede programar la ejecución retardada.
- **p_evitar_duplicados:** Controla los valores de llamada a esta función, evitando encolar si ya existe una alerta con la misma parametrización. Valores posibles:
 - N: No controlar.
 - P: Controlar solo los parametros "p_parametro_xxx".
 - C: Controlar todos los campos de configuración [resto de variables entrada].
 - S: Controlar todos los valores de entrada.

IMPORTANTE: La alerta únicamente se ejecutará en el momento en que la transacción que llame a PKALRT.GRABAR_COLA_ALERTAS haga un COMMIT, mientras no lo haga el motor de alertas no la procesará. En el caso de querer ejecutar la alerta sin esperar a que finalice la transacción, se puede llamar a PKALRT.GRABAR_COLA_ALERTAS_AT, con los mismos parámetros, la alerta queda registrada y será ejecutada independientemente de la transacción que la lanza.

Ejemplos:

Al ser el único parámetro obligatorio el número de alerta a ejecutar, lanzar una alerta es tan simple como ejecutar lo siguiente (siendo en el ejemplo la alerta 1006 la que se quiere ejecutar):

- PKALRT.GRABAR_COLA_ALERTAS(p_numero_alerta => 1006);

Otro ejemplo modificando el asunto y el destinatario del correo.

- PKALRT.GRABAR_COLA_ALERTAS(p_numero_alerta => 1006, p_asunto_alerta => 'Asunto', p_notificar_mail_to => 'elias.fernandez@edisa.com');

Se puede poner una alerta en cola para que se ejecute en tiempo real, esto es independiente de que esté o no configurada como "Tiempo Real". Al grabar la cola se genera un scheduled job para ejecutarse en ese momento, de forma que la ejecución es asíncrona. La opción está únicamente disponible en `pkalrt.f_grabar_cola_alertas_at` y `pkalrt.grabar_cola_alertas_at`, es decir, las que ponen la alerta en cola con una transacción autónoma. El parámetro que hay que indicar es "`p_ejecutar_tiempo_real => 'S'`"

Grabar cola de alerta con archivos adjuntos en formato BLOB

Para poner en cola una alerta y a esa alerta adjuntarle archivos binarios que se encuentran en variables BLOB hay que usar la función `PKALRT.F_GRABAR_COLA_ALERTAS` o `PK_ALRT_F_GRABAR_COLA_ALERTAS_AT`. Esta función es similar a los procedimientos `PKALRT.GRABAR_COLA_ALERTAS` y `PKALRT.GRABAR_COLA_ALERTAS_AT` con las siguientes diferencias:

- La función devuelve el ID (NUMBER) que identifica el registro en la cola de la alerta.
- La función tiene el parámetro `pt_adjuntos_blob` que es un array indexado por número de archivos BLOB a adjuntar a la alerta. El tipo de ese array es `pk_email.tabla_blobs_adjuntos`;

Ejemplo para adjuntar 2 archivos en formato BLOB, uno de ellos se carga de un directorio de la base de datos y el otro ya se encuentra en la memoria de PK_BLOB2BD.

```

DECLARE
  v_id_cola      NUMBER;
  vt_adjuntos_blob pk_email.tabla_blobs_adjuntos;
BEGIN
  vt_adjuntos_blob(1).nombre_archivo_original := 'archivo_excel.xls';
  vt_adjuntos_blob(1).archivo_blob2bd := pk_blob2bd.get_fichero();
  vt_adjuntos_blob(2).nombre_archivo_original := 'ejemplo.pdf';
  vt_adjuntos_blob(2).archivo_blob2bd :=
    pk_blob2bd.lee_blob_desde_fic(p_directorio => 'BLOB_TEMP',
                                    p_nombre_archivo => 'ejemplo.pdf');
  v_id_cola := pkalrt.f_grabar_cola_alertas(p_numero_alerta => 7802,
                                              pt_adjuntos_blob => vt_adjuntos_blob);
END;

```

Gestión de la alerta a través del pl/sql.

Propiedades

Desde el PL/SQL se puede modificar el comportamiento de la alerta mediante la llamada a `PKALRTJOB.SET_PROPIEDAD('<propiedad>', '<valor_propiedad>');`

Las propiedades disponibles son las siguientes:

- **NOTIFICAR_MAIL_TO**: Lista de destinatarios del correo electrónico, separados por ; (punto y coma) o por , (coma). En el caso de haber direcciones de correo previamente en la alerta serán sobrescritas.
- **NOTIFICAR_MAIL_TO_ADD**: Añade los correos indicados sin sobrescribir los que ya tiene la alerta configurados.
- **NOTIFICAR_MAIL_TO_USU**: Igual que NOTIFICAR_MAIL_TO, pero en vez de indicar direcciones de correo se indicarán usuarios de Libra y la dirección de correo electrónico se irá a buscar a la ficha del usuario.
- **NOTIFICAR_MAIL_TO_USU_ADD**: Igual que NOTIFICAR_MAIL_TO_ADD, pero en vez de indicar direcciones de correo se indicarán usuarios de Libra y la dirección de correo electrónico se irá a buscar a la ficha del usuario.
- **NOTIFICAR_MAIL_TO_PER**: Igual que NOTIFICAR_MAIL_TO, pero en vez de indicar direcciones de correo se indicarán perfiles de menú y se incluirá la dirección de correo electrónico de todos los usuarios de Libra que no estén de baja y que tengan dirección de correo electrónico en su ficha.

- **NOTIFICAR_MAIL_TO_PER_ADD:** Igual que NOTIFICAR_MAIL_TO_ADD, pero en vez de indicar direcciones de correo se indicarán los perfiles de menú para incluir todos los usuarios de Libra que no estén de baja y tengan dirección de correo electrónico en su ficha.
- **NOTIFICAR_MAIL_TO_SU:** Igual que NOTIFICAR_MAIL_TO. Se incorpora la dirección de correo de todos los usuarios de Libra marcados como superusuarios, que tengan dirección de correo electrónico en su ficha y que no estén de baja. En '*<valor_propiedad>*' se debe de pasar '.', ya que ese valor será ignorado.
- **NOTIFICAR_MAIL_TO_SU_ADD:** Igual que NOTIFICAR_MAIL_TO_ADD. Se añaden las direcciones de correo de todos los usuarios de Libra marcados como superusuarios, que tengan dirección de correo electrónico en su ficha y que no estén de baja. En '*<valor_propiedad>*' se debe de pasar '.', ya que ese valor será ignorado.
- **NOTIFICAR_MAIL_CC:** Lista de destinatarios con copia del correo electrónico, separados por ; (punto y coma) o por , (coma). En el caso de haber direcciones de correo previamente en la alerta serán sobrescritas.
- **NOTIFICAR_MAIL_CC_ADD:** Añade correos a la lista de correos con copia sin sobrescribir los que ya tiene la alerta configurados.
- **NOTIFICAR_MAIL_CC_USU:** Igual que NOTIFICAR_MAIL_CC, pero en vez de indicar direcciones de correo se indicarán usuarios de Libra y la dirección de correo electrónico se irá a buscar a la ficha del usuario.
- **NOTIFICAR_MAIL_CC_USU_ADD:** Igual que NOTIFICAR_MAIL_CC_ADD, pero en vez de indicar direcciones de correo se indicarán usuarios de Libra y la dirección de correo electrónico se irá a buscar a la ficha del usuario.
- **NOTIFICAR_MAIL_CC_PER:** Igual que NOTIFICAR_MAIL_CC, pero en vez de indicar direcciones de correo se indicarán perfiles de menú y se incluirá la dirección de correo electrónico de todos los usuarios de Libra que no estén de baja y que tengan dirección de correo electrónico en su ficha.
- **NOTIFICAR_MAIL_CC_PER_ADD:** Igual que NOTIFICAR_MAIL_CC_ADD, pero en vez de indicar direcciones de correo se indicarán los perfiles de menú para incluir todos los usuarios de Libra que no estén de baja y tengan dirección de correo electrónico en su ficha.
- **NOTIFICAR_MAIL_CC_SU:** Igual que NOTIFICAR_MAIL_CC. Se incorpora la dirección de correo de todos los usuarios de Libra marcados como superusuarios, que tengan dirección de correo electrónico en su ficha y que no estén de baja. En '*<valor_propiedad>*' se debe de pasar '.', ya que ese valor será ignorado.
- **NOTIFICAR_MAIL_CC_SU_ADD:** Igual que NOTIFICAR_MAIL_CC_ADD. Se añaden las direcciones de correo de todos los usuarios de Libra marcados como superusuarios, que tengan dirección de correo electrónico en su ficha y que no estén de baja. En '*<valor_propiedad>*' se debe de pasar '.', ya que ese valor será ignorado.
- **NOTIFICAR_MAIL_CCO:** Lista de destinatarios con copia oculta del correo electrónico, separados por ; (punto y coma) o por , (coma). En el caso de haber direcciones de correo previamente en la alerta serán sobrescritas.
- **NOTIFICAR_MAIL_CCO_ADD:** Añade correos a la lista de correos con copia oculta sin sobrescribir los que ya tiene la alerta configurados.
- **NOTIFICAR_MAIL_CCO_USU:** Igual que NOTIFICAR_MAIL_CCO, pero en vez de indicar direcciones de correo se indicarán usuarios de Libra y la dirección de correo electrónico se irá a buscar a la ficha del usuario.
- **NOTIFICAR_MAIL_CCO_USU_ADD:** Igual que NOTIFICAR_MAIL_CCO_ADD, pero en vez de indicar direcciones de correo se indicarán usuarios de Libra y la dirección de correo electrónico se irá a buscar a la ficha del usuario.
- **NOTIFICAR_MAIL_CCO_PER:** Igual que NOTIFICAR_MAIL_CCO, pero en vez de indicar direcciones de correo se indicarán perfiles de menú y se incluirá la dirección de correo electrónico de todos los usuarios de Libra que no estén de baja y que tengan dirección de correo electrónico en su ficha.
- **NOTIFICAR_MAIL_CCO_PER_ADD:** Igual que NOTIFICAR_MAIL_CCO_ADD, pero en vez de indicar direcciones de correo se indicarán los perfiles de menú para incluir todos los usuarios de Libra que no estén de baja y tengan dirección de correo electrónico en su ficha.
- **NOTIFICAR_MAIL_CCO_SU:** Igual que NOTIFICAR_MAIL_CCO. Se incorpora la dirección de correo de todos los usuarios de Libra marcados como superusuarios, que

- tengan dirección de correo electrónico en su ficha y que no estén de baja. En '**<valor_propiedad>**' se debe de pasar '.', ya que ese valor será ignorado.
- **NOTIFICAR_MAIL_CCO_SU_ADD:** Igual que NOTIFICAR_MAIL_CCO_ADD. Se añaden las direcciones de correo de todos los usuarios de Libra marcados como superusuarios, que tengan dirección de correo electrónico en su ficha y que no estén de baja. En '**<valor_propiedad>**' se debe de pasar '.', ya que ese valor será ignorado.
 - **NOTIFICAR_SMSS:** Lista de números de destino de mensajes SMS, separados por , (coma).
 - **NOTIFICAR_USUARIOS_LIBRA:** Lista de destinatarios de Libra que recibirán el mensaje en el menú de Libra, separados por ; (punto y coma) o por , (coma).
 - **ENVIAR_EMAIL:**
 - S: Fuerza al envío de correo electrónico parametrizado.
 - N: Anula el envío del correo electrónico.
 - I: Sólo se enviará el informe si alguno de los informes del generador de informes parametrizado devuelve datos.
 - **ENVIADA_POR:** Dirección de correo electrónico del remitente del mensaje.
 - **ASUNTO_ALERTA:** Texto del asunto del mensaje.
 - **APLICAR_PARSER_ASUNTO:** Si se pasa el valor 'S', al procesar la alerta se aplicará el parseador al texto del asunto que tenga en ese momento la alerta, es decir, se procesan los valores que están entre corchetes {}. Ver sección "Parser".
 - **TEXTO_ALERTA:** Texto en formato plano de la alerta.
 - **APLICAR_PARSER_TEXTO:** Igual que APLICAR_PARSER_ASUNTO pero sobre el texto en formato plano de la alerta.
 - **TEXTO_ALERTA_HTML:** Texto en formato HTML de la alerta.
 - **APLICAR_PARSER_TEXTO_HTML:** Igual que APLICAR_PARSER_ASUNTO pero sobre el texto HTML de la alerta.
 - **TEXTO_ALERTA_SMS:** Texto SMS de la alerta.
 - **APLICAR_PARSER_TEXTO_SMS:** Igual que APLICAR_PARSER_ASUNTO pero sobre el texto para el envío de SMS de la alerta.
 - **NOTIFICAR_MAIL_ATTACHMENTS:** Permite modificar los adjuntos que se incluyen en la alerta.
 - **INICIALIZAR_ARCHIVOS_ADJUNTOS:** Se usa para indicar que se van a indicar archivos almacenados en ARCHIVOS_ERP a adjuntar en el mensaje.
 - **ADD_ADJUNTO_X_ID:** Adjuntar archivo almacenado en ARCHIVOS_ERP, hay que indicar el ID del archivo a adjuntar. Hay que invocar esta propiedad tantas veces como archivos a adjuntar.
 - **INICIALIZAR_INFORMES_ADJUNTOS:** Se usa para indicar que se van a indicar informes del generador de informes a mayores de los parametrizados en la alerta.
 - **ADD_ADJUNTO_X_URL:** Permite añadir archivos a descargar desde una URL. Esta propiedad a diferencia de las anteriores recibe dos parámetros, por lo que la llamada es de la siguiente forma: PKALRTJOB.SET_PROPIEDAD('ADD_ADJUNTO_X_URL', '<url>', '<nombre_archivo>');
 - <url>: Dirección desde la que descargar el archivo adjunto.
 - <nombre_archivo>: Nombre con el que se adjuntará el archivo al correo electrónico.

Añadir informe del generador de informes desde el PL/SQL de la alerta.

Se pueden añadir desde el PL/SQL informes a adjuntar en el mensaje mediante PKALRTJOB.ADD_ADJUNTO_GI('<informe>', '<nombre_archivo>', '<idioma>', '<empresa>', '<usuario>', '<plantilla_valores_defecto>', '<tipo_archivo>', <codigo_configuracion>);

- **<informe>:** Código del informe a añadir.
- **<nombre_archivo>:** Nombre del archivo que se adjuntará en el mensaje con el contenido del informe.

- **<idioma>**: Idioma de las etiquetas del informe.
- **<empresa>**: Empresa de Libra en la que se ejecutará el informe.
- **<usuario>**: Código del usuario con el que se ejecutará el informe.
- **<plantilla_valores_defecto>**: Puede ser NULL. Indica el código de la plantilla de valores de los filtros con los que se ejecutará el informe.
- **<tipo_archivo>**: Por defecto es Excel XML, que lo asumirá si se pasa NULL o EXCELXML, para otros formatos:
 - HTML: Renderizado en formato HTML
 - TXT: Texto plano.
 - CSV: Texto separado por comas
- **<codigo_configuracion>**: Puede ser NULL. Indica el código de configuración de columnas con la que se ejecutará el informe.

Añadir archivo binario BLOB desde el PL/SQL de la alerta.

Se pueden añadir archivos binarios desde el PL/SQL a adjuntar en el mensaje mediante PKALRTJOB.ADD_ADJUNTO_BLOB('<blob>', '<nombre_archivo>');

- **<blob>**: Archivo binario BLOB a adjuntar.
- **<nombre_archivo>**: Nombre con el que se adjuntará en el mensaje ese archivo.

Ejemplo:

```
pkalrtjob.add_adjunto_blob(p_blob => pk_blob2bd.get_fichero(),
                            p_nombre_archivo => 'prueba_blob.pdf');
```

Paso de parámetros a informes del generador de informes.

Desde el código PL/SQL de la alerta se pueden indicar parámetros a los informes que va a ejecutar la alerta, para ello hay que llamar al procedimiento **PK_B2B_GENINF.GI_SET_VALOR_PARAMETRO('PARAMETRO', 'VALOR_PARAMETRO')**

- **PARAMETRO:** Código del parámetro de en los informes.
- **VALOR_PARAMETRO:** Valor que se le quiere pasar al parámetro.

Todos los informes que tengan un filtro con ese código de parámetro asumirán el valor indicado.

Ejemplo paso del valor '01' en el parámetro **F_CODIGO**:
pk_b2b_geninf.gi_set_valor_parametro('F_CODIGO', '01');

Ejemplo paso del valor recibido en la cola de alerta en **PARAMETRO_VARCHAR01** al parámetro **F_CODIGO** de los informes:

pk_b2b_geninf.gi_set_valor_parametro('F_CODIGO', PKALRTJOB.GET_PROPIEDAD_VARCHAR2('PARAMETRO_VARCHAR01'));

Ejemplo envío de un PDF con una factura desde la alerta.

Vamos a ver como incluir en el correo electrónico un PDF con el resultado de la ejecución de un report de un formato de factura.

Requisitos: Un formato de factura que acepte por parámetro la clave primaria de la factura.

- Nombre informe: **l_factura_gen.rdf**
- Parámetros que le hay que pasar:
 - **p_numero_factura**
 - **p_numero_serie**
 - **p_ejercicio**
 - **p_empresa**

```

DECLARE
  v_numero_factura FACTURAS_VENTAS.NUMERO_FACTURA%TYPE := 1;
  v_numero_serie   FACTURAS_VENTAS.NUMERO_SERIE%TYPE := '0';
  v_ejercicio      FACTURAS_VENTAS.NUMERO_FACTURA%TYPE := 2013;
  v_empresa        FACTURAS_VENTAS.EMPRESA%TYPE := '013';
  v_rdo            VARCHAR2(500);
BEGIN
  pkpantallas.set_empresa_usuario(p_empresa => '001', p_usuario => 'EDISA');
  pk_libra_impresion.inicializar;
  pk_libra_impresion.set_informe('l_factura_gen.rdf');
  pk_libra_impresion.set_dispositivo('BD');
  pk_libra_impresion.set_formato('PDF');
  pk_libra_impresion.set_parametro(p_parametro => 'P_NUMERO_FACTURA',
                                    p_valor => v_numero_factura);
  pk_libra_impresion.set_parametro(p_parametro => 'P_NUMERO_SERIE',
                                    p_valor => v_numero_serie);
  pk_libra_impresion.set_parametro(p_parametro => 'P_EJERCICIO',
                                    p_valor => v_ejercicio);
  pk_libra_impresion.set_parametro(p_parametro => 'P_EMPRESA',
                                    p_valor => v_empresa);
  v_rdo := pk_libra_impresion.lanzar_impresion();
  pkalrtjob.add_adjunto_blob(p_blob => pk_blob2bd.get_fichero(),
                            p_nombre_archivo => 'factura-' ||
                                         v_numero_serie || '-' ||
                                         v_ejercicio || '-' ||
                                         v_numero_factura || '.pdf');
END;

```

Parser

Introducción

En el Asunto, texto y texto HTML se puede aplicar el parser, de forma que el contenido de esos campos es usado como plantilla y serán cubiertos en tiempo de ejecución de la alerta. Por ejemplo, partiendo de lo siguiente:

Estimado Sr. {clientes.nombre} le informamos que dispone de la factura {facturas_ventas.numero_serie}/{facturas_ventas.numero_factura} de fecha {facturas_ventas.fecha_factura} disponible para descarga.

Una vez aplicado el parseador sobre esa plantilla se obtendrá algo similar a esto:

Estimado Sr. José Fernández López le informamos que dispone de la factura FV/3433 de fecha 08/06/2011 disponible para descarga.

Para realizar el proceso del ejemplo hay que indicar qué factura es la que se tiene usar para reemplazar el texto, esto se hará en el código pl/sql de la alerta. El ejemplo para ejecutar el parseador para obtener el resultado del ejemplo sería el siguiente:

```
pkbdparser.set_variable('numero_factura', 3433);
pkbdparser.set_variable('numero_serie', 'FV');
pkbdparser.set_variable('ejercicio', '2011');
pkbdparser.set_variable('empresa', '013');
pkbdparser.set_propiedad_tabla('FACTURAS_VENTAS',      'WHERE_DEFECTO',      'numero_factura =
{numero_factura} AND numero_serie = {numero_serie} AND ejercicio = {ejercicio} AND empresa =
{empresa}');
pkbdparser.set_propiedad_tabla('CLIENTES',          'WHERE_DEFECTO',          'codigo_rapido      =
{facturas_ventas.cliente} AND codigo_empresa = {empresa}');
```

Tipos de etiquetas

Se contemplan los siguientes tipos de etiquetas

- **Variable:** Valor indicado previamente en una variable, tendrá el formato {variable}. Ejemplo: {numero_factura}
- **Valor de tabla:** Indica que debe sustituirse esa etiqueta por el valor del campo de una determinada tabla. Esta etiqueta tendrá el siguiente formato: {tabla.campo}. Ejemplo: {facturas_ventas.cliente}. En caso de estar en una zona de repetición, es decir, entre etiquetas de inicio de repetición y de fin de repetición se usará {alias.campo} en vez de {tabla.campo}

Se pueden usar modificadores para alterar el resultado, en el caso de usar modificadores el formato será {alias.campo:modificador1|valor modificador1|modificador2|valor_modificador2|..|modificador n| valor modificador n}, por ejemplo: {crmexpedientes_cab.fecha_alta|FM|DD/MM/HH24:MI:SS}. Modificadores posibles:

- FM: Máscara de formato.
- TYPE: Tipo de campo, Valores posibles:
 - HTML: Se le aplica al resultado la función pk_xml.codifica_texto_to_html, para reemplazar los caracteres especiales del HTML por la codificación correcta. También se reemplazan los retornos de carro por

 - ESHTML: Se supone que se el resultado ya viene codificado en HTML y por tanto no se le debe de alterar.
- **De repetición**, se dividen en otras dos etiquetas:
 - **Inicio de repetición:** Tiene el siguiente formato. {R:tabla:alias:condición:orden:R}, dentro de condición se podrán usar etiquetas de *valor de tabla* o de *variable* (las del punto anterior). Ejemplo: {R:albaran_ventas_lin:AVL:articulo = {articulos.codigo_articulo} AND empresa =

{empresa}:numero_albaran DESC:R}. En caso de no indicar una condición o una ordenación se dejará en blanco, es decir: {R:albaran_ventas_lin:AVL::R}, en ese caso se usará la condición y la ordenación que tenga asignada la tabla por defecto.

- **Fin de repetición:** Indica que todo lo que se encuentra entre “Inicio de repetición” y “Fin de repetición” se procesará tantas veces como filas devuelva la consulta sobre la tabla indicada en la etiqueta de inicio. Tendrá el siguiente formato: {E:alias:E}, ejemplo: {E:AVL:E}
- **Generador de informes:** Ejecuta un generador de informes y el resultado del informe lo incluye en el texto. El formato de la etiqueta es: {GI:<informe>:<idioma>:<empresa>:<usuario>:<plantilla de valores de filtro>:<tipo>:< código de configuración>}.
 - <informe>: Código del informe del generador de informes a ejecutar.
 - <idioma>: Idioma en el que se generarán las etiquetas de los campos.
 - <empresa>: Código de la empresa sobre la que se ejecutará el informe.
 - <usuario>: Usuario de Libra con el que se ejecutará el informe, el usuario determinará los permisos de ejecución.
 - <plantilla de valores de filtro>: Valores de filtro a aplicar al informe.
 - <tipo>: Si se indica HTML se incluirán al resultado etiquetas para maquetar el resultado en formato HTML. Si lo que se está formateando es un campo de tipo HTML lo detectará y no es necesario incluirlo. Si no se está formateando un campo de tipo HTML y no se indica nada en este campo el resultado será formateado en texto plano.
 - < código de configuración>: Si el informe tiene varias configuraciones de columnas, se puede indicar la que se debe de utilizar.

Ejemplo: {GI:DIARIOS:01:013:EDISA:245:HTML:45} (Ejecuta el informe DIARIOS, con las etiquetas en idioma 01, sobre la empresa 013, con los permisos del usuario EDISA, aplicando la plantilla de valores por defecto 245, el resultado se genera con etiquetas HTML y utilizando la configuración de columnas con código 45).

- **Función de base de datos:** Ejecuta la función de base de datos indicada que devuelva un resultado en VARCHAR2 o en CLOB, con el formato {SF:<función a ejecutar>:EF}. A la función se le pueden pasar los parámetros que sean necesarios obtenidos de variables, con {variable} o de un campo con {tabla.campo}. **Ejemplo:** {SF:PRUEBA_FUNCION(p_numero_serie => {facturas_ventas.numero_serie}, p_numero_factura => {facturas_ventas.numero_factura}):EF}
- **Imagen:** En parámetros generales de menú, en la pestaña “Imágenes Públicas” se pueden configurar enlaces a URLs que apunten a imágenes. Con el parseador se puede generar la etiqueta HTML que haga referencia a esa imagen con: {IMGHTML:CODIGO}, por ejemplo: {IMGHTML:GE5MOLW084CE8YLNQFTI4DV5TN8F4}
- **Modificación de propiedades**, se dividen en:
 - **Variables:** Permite modificar en un momento dado una variable, tiene el siguiente formato: {SVA:< código de la variable>:< valor de la variable>:SVA}, Ejemplo: {SVA:codigo_consgen:CONSGEN:SVA}
 - **Propiedades generales:** Permite cambiar en un momento dado propiedades generales del parseador, tiene el siguiente formato: {SPG:< código de la propiedad>:< valor de la propiedad>:SPG}, Ejemplo: {SP:MASCARA_FECHAS:YYYY/MM/DD:SP}
 - **Propiedades de tablas:** Permite cambiar en un momento dado propiedades generales de una tabla, tiene el siguiente formato: {SPT:< código de tabla / alias>:< código de la propiedad>:< valor de la propiedad>:SPT}, Ejemplo: {SPT:PROGRAMAS_ERP:WHERE_DEFECTO:codigo={codigo_consgen}:SPT }
 - **Añadir columnas calculadas:** Permite añadir columnas calculadas a una tabla y ser usadas dentro del parseador como si fuese una columna más de la tabla, tiene el siguiente formato: {ACT:< nombre de tabla / alias>:< nombre de

columna>:<sentencia SQL para obtener el resultado:ACT}. Ejemplo:
{ACT:CRMEXPEDIENTES_CAB:NUMERO_LINEAS:(SELECT COUNT(*)
FROM crmexpedientes_lin l WHERE l.numero_expediente =
crmexpedientes_cab.numero_expediente AND l.empresa =
crmexpedientes_cab.empresa):ACT}

Propiedades generales del proceso

Las propiedades del parser se modificarán el código pl/sql de la alerta.

Para establecer las propiedades generales del proceso se usará el procedimiento **pkbdparser.set_propiedad(<codigo_propiedad>', '<valor>');**

Las propiedades que se pueden establecer son las siguientes:

- **MASCARA_FECHAS:** Máscara a aplicar a los campos de tipo fecha, en el caso de no especificar esta propiedad se usará DD/MM/YYYY
- **MASCARA_NUMEROS:** Máscara a aplicar a los campos de tipo numérico, en el caso de no especificar esta propiedad se realizará un TO_CHAR sin indicar ninguna máscara.
- **PLANTILLA:** Texto que contiene la plantilla a usar.
- **TRAZA:** Si se pasa el valor 'S' se guardará en LIBRA_LOG una traza interna para depurar del proceso.

Variables

Las variables son valores que no se obtienen de ninguna tabla, por lo que hay que suministrárselos al proceso, luego serán usadas en las etiquetas de tipo **Variable**.

Para declarar una variable hay que llamar al procedimiento **pkbdparser.set_variable(<codigo de la variable>', <valor>);** Por ejemplo: **pkbdparser.set_variable('numero_factura', 3433);**

Propiedades de tabla

Para ejecutar el proceso hay que limitar los registros de las tablas que se usan y en el caso de bloques de repetición el orden en el que se procesarán, para ello se dispone del procedimiento **pkbdparser.set_propiedad_tabla(<alias>', '<propiedad>', '<valor>');** Por ejemplo: **pkbdparser.set_propiedad_tabla('CLIENTES', 'WHERE_DEFECTO', 'codigo_rapido = {facturas_ventas.cliente} AND codigo_empresa = {empresa}');**

- **<alias>:** Si es una tabla que se usa en un grupo repetitivo se indicará el alias del grupo repetitivo en vez del nombre de la tabla.
- **<propiedad>:** Existen las siguientes propiedades:
 - **WHERE_DEFECTO:** Condición que se aplicará al hacer la consulta de la tabla. Se pueden usar etiquetas de tipo variable o de valor de tabla. Si en <alias> se indicó el alias de un grupo repetitivo y en este grupo repetitivo se indicó una condición, esa condición prevalecerá sobre el valor pasado en esta propiedad.
 - **ORDER_BY_DEFECTO:** Condición de ordenación que se aplicará al consultar la tabla. Al igual que en WHERE_DEFECTO, si en el repetitivo se indica una condición de ordenación, esa condición de ordenación prevalecerá sobre la que se pasa en esta propiedad.

Propiedades de columna

Permite añadir columnas calculadas a una tabla y ser usadas dentro del parseador como si fuese una columna más de la tabla.

Para añadir una columna calculada hay que llamar al procedimiento `pkbdparser.add_columna_tabla(<alias>', '<nombre de columna>', '<sentencia SQL para obtener el resultado>');`

Ejemplo: `pkbdparser.add_columna_tabla('CRMEXPEDIENTES_CAB', 'NUMERO_LINEAS', '(SELECT COUNT(*) FROM crmexpedientes_lin l WHERE l.numero_expediente = crmexpedientes_cab.numero_expediente AND l.empresa = crmexpedientes_cab.empresa');`

- **<alias>:** Si es una tabla que se usa en un grupo repetitivo se indicará el alias del grupo repetitivo en vez del nombre de la tabla.
- **<nombre de columna>:** Nombre de la columna, que será luego usada en el texto como alias.columna.
- **<sentencia SQL para obtener el resultado>:** *Sentencia SQL que se debe de añadir a la SELECT para obtener el resultado de la columna .*

Ejemplo

Alerta para notificar los clientes que se han dado de alta en la semana de la fecha en que se ejecuta la alerta.

Código PL/SQL:

```
DECLARE
  v_fecha_inicio DATE;
  v_fecha_fin      DATE;
BEGIN
  v_fecha_inicio := TRUNC(SYSDATE) - TO_NUMBER(TO_CHAR(SYSDATE, 'D')) + 1;
  v_fecha_fin := v_fecha_inicio + 6;

  pkalrtjob.set_propiedad('APLICAR_PARSER_ASUNTO', 'S');
  pkalrtjob.set_propiedad('APLICAR_PARSER_TEXTO', 'S');
  pkalrtjob.set_propiedad('APLICAR_PARSER_TEXTO_HTML', 'S');
  pkbdparser.set_variable('empresa', '013');
  pkbdparser.set_variable('fecha_inicio', v_fecha_inicio);
  pkbdparser.set_variable('fecha_fin', v_fecha_fin);
  pkbdparser.set_propiedad_tabla('CL', 'WHERE_DEFECTO', 'codigo_empresa = {empresa} AND
fecha_alta BETWEEN {fecha_inicio} AND {fecha_fin}');
END;
```

- **v_fecha_inicio := TRUNC(SYSDATE) - TO_NUMBER(TO_CHAR(SYSDATE, 'D')) + 1;** Asigna en la variable la fecha del lunes de la semana en la que se ejecuta la alerta.
- **v_fecha_fin := v_fecha_inicio + 6;** Asigna en la variable la fecha del domingo de la semana en la que se ejecuta la alerta.
- **pkalrtjob.set_propiedad('APLICAR_PARSER_ASUNTO', 'S');** Le indica al motor de alertas que debe de aplicar el parser al texto que contiene el campo “Asunto” de la notificación.
- **pkalrtjob.set_propiedad('APLICAR_PARSER_TEXTO', 'S');** Le indica al motor de alertas que debe de aplicar el parser al texto que contiene el campo “Texto” de la notificación.
- **pkalrtjob.set_propiedad('APLICAR_PARSER_TEXTO_HTML', 'S');** Le indica al motor de alertas que debe de aplicar el parser al texto que contiene el campo “Texto HTML” de la notificación.
- **pkbdparser.set_variable('empresa', '013');** Define la variable “empresa” con el valor 013 en el parser.

- **pkbdparser.set_variable('fecha_inicio', v_fecha_inicio);** Define la variable "fecha_inicio" en el parser con el valor de la variable "v_fecha_inicio" que se calculó al principio.
- **pkbdparser.set_variable('fecha_fin', v_fecha_fin);** Define la variable "fecha_fin" en el parser con el valor de la variable "v_fecha_fin" que se calculó al principio.
- **pkbdparser.set_propiedad_tabla('CL', 'WHERE_DEFECTO', 'codigo_empresa = {empresa} AND fecha_alta BETWEEN {fecha_inicio} AND {fecha_fin}');** Se le indica al parser que si en la plantilla se define una tabla con alias CL la condición que debe de aplicar para obtener los registros, en este caso aplicará codigo_empresa = '013' y filtrará también por el campo FECHA_ALTA cuyos registros se encuentren entre el margen de fechas que se asignó en el pl/sql de la alerta.

Asunto

En el asunto de la alerta se pondrá lo siguiente:

Clientes creados desde el {fecha_inicio} al {fecha_fin}

Antes de enviar la notificación cambiará {fecha_inicio} y {fecha_fin} por sus valores correspondientes.

Texto

Texto que se visualizará en aquellos dispositivos que no puedan visualizar el texto en HTML. En el texto se pondrá lo siguiente:

{ACT:CL:CODIGO15ESP:RPAD(clientes.codigo_rapido, 15, ''):ACT}Nuevos clientes:

CODIGO	NOMBRE
{R:CLIENTES:CL:::R}{cl.codigo15esp}	{cl.nombre}
{E:CL:E}	

- **{ACT:CL:CODIGO15ESP:RPAD(clientes.codigo_rapido, 15, ''):ACT}**: Añade a la tabla que tenga el alias CL el campo calculado CODIGO15ESP que será el resultado de ejecutar RPAD(clientes.codigo_rapido, 15, '').
- **{R:CLIENTES:CL:::R}{cl.codigo15esp} {cl.nombre}**: Define el comienzo de un bloque repetitivo sobre la tabla CLIENTES y le asigna el alias CL, por cada registro que se obtiene de CLIENTES mostrará el campo calculado CODIGO15ESP y el NOMBRE del cliente.
- **{E:CL:E}**: Se indica donde termina el bloque repetitivo de la tabla con alias CL.

Texto HTML

Texto que será visualizado en dispositivos que puedan renderizar HTML. Se pondrá lo siguiente:

```
<table width="100%">
<tr bordercolor="#999999" bordercolordark="#999999" bordercolorlight="#999999" bgcolor="#999999">
<td width="20%"><b>CODIGO</b></td>
<td width="80%"><b>NOMBRE</b></td>
</tr>
```

```

{R:CLIENTES:CL:::R}
<tr>
<td width="20%">{cl.codigo_rapido}</td>
<td width="80%">{cl.nombre}</td>
</tr>
{E:CL:E}
</table>

```

El funcionamiento es similar al explicado en “Texto”, con la diferencia que al hacerse en base a una tabla no hace falta tener un campo calculado que ajuste el tamaño del código a 15 espacios.

Clientes creados desde el 31/10/2011 al 06/11/2011

► De:  Notificaciones CRM

Nuevos clientes:

CODIGO	NOMBRE
5002	CORTE INGLES PORRIÑO HIJO VIGO
00001	SAMSUNG ELECTRONICS, S.A
0004V4	BENITO FERNANDEZ

Resultado en formato Texto Plano.

Resultado en formato HTML

Clientes creados desde el 31/10/2011 al 06/11/2011

► De:  Notificaciones CRM

CODIGO	NOMBRE
5002	CORTE INGLES PORRIÑO HIJO VIGO
5002	SAMSUNG ELECTRONICS, S.A
5002	BENITO FERNANDEZ

Apéndices

Autorizar descargas de direcciones WEB

IMPORTANTE: Esto no es necesario en el caso de tener instalado el servicio GAL_PROXY de Galileo, que es la opción recomendada en estos casos.

Desde Oracle 11g, el acceso por defecto a conexiones externas está por defecto deshabilitado, para autorizar las URL a las que se permiten realizar conexiones hay que dar de alta el registro en el ACL (Lista de acceso permitido), de la siguiente forma (Cambiar "blog.edisa.com" por la dirección deseada a autorizar):

```

DECLARE
  acl_path  VARCHAR2(4000);
  v_existe  VARCHAR2(1);
BEGIN
  SELECT acl
  INTO acl_path
  FROM dba_network_acls
  WHERE HOST = 'blog.edisa.com'
    AND lower_port IS NULL
    AND upper_port IS NULL;

  dbms_network_acl_admin.add_privilege (acl_path, USER, TRUE, 'connect');
  COMMIT;
EXCEPTION
  WHEN NO_DATA_FOUND THEN
    BEGIN
      SELECT 'S'
      INTO v_existe
      FROM dba_network_acls
      WHERE acl LIKE '%ACL_LIBRA.xml%';
    EXCEPTION
      WHEN NO_DATA_FOUND THEN
        v_existe := 'N';
      WHEN TOO_MANY_ROWS THEN
        v_existe := 'S';
    END;
  IF v_existe = 'N' THEN
    dbms_network_acl_admin.create_acl ('ACL_LIBRA.xml', 'Permisos TCP usuario LIBRA', USER, TRUE, 'connect');
  END IF;
  dbms_network_acl_admin.assign_acl ('ACL_LIBRA.xml', 'blog.edisa.com');
  COMMIT;
END;
/

```

Generación de Reports y envío por alerta

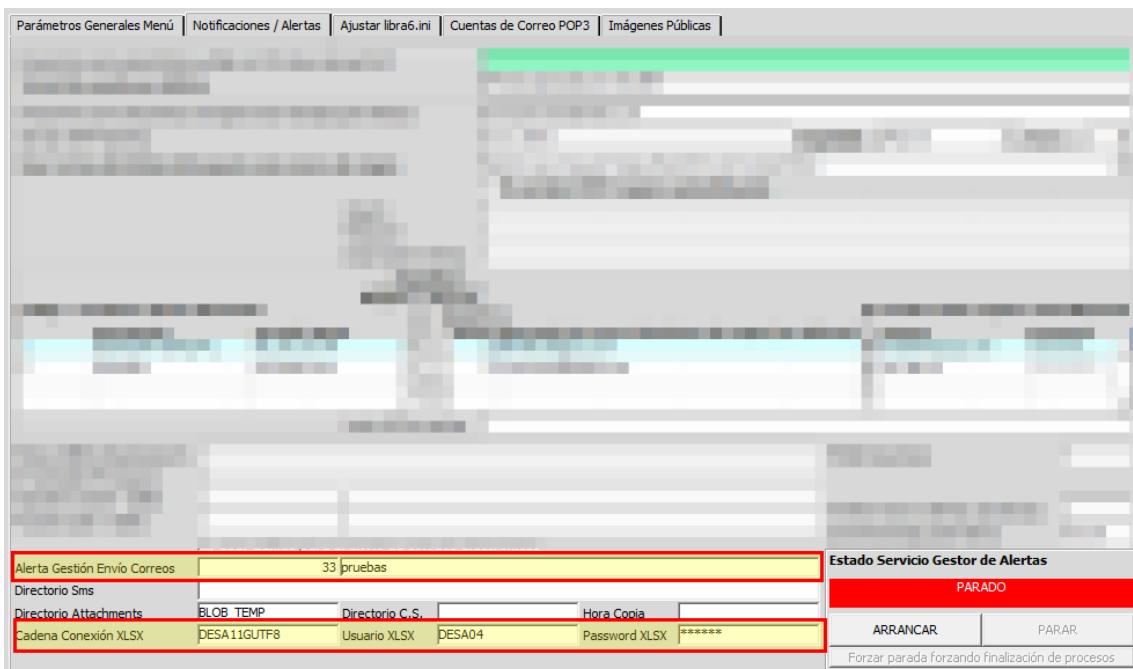
Existe la posibilidad de generar un Report y enviarlo por alertas como PDF. Para poder usar esta funcionalidad necesitamos una serie de parametrizaciones, que se detallan a continuación:

La generación de reports se hace por medio del servidor de Reports de Forms12c o BI-Publisher. Para que funcione es necesario no tener parametrizada la referencia "GAL REP GEN" en a_socket y por otro lado disponer del servidor de Reports de Forms12c y BI-Publisher correctamente configurado y parametrizado en la pestaña de integraciones de parámetros de menú (Esta parte la hace sistemas).

Parametrización en parámetros generales de Libra

A nivel de parámetros generales de Libra hay que activar obligatoriamente la "Alerta de Gestión de envío de correos". Activar esta parametrización va a afectar al comportamiento de Libra en el envío de correos, básicamente cualquier correo que envíemos ejecutando por PL/SQL una llamada a pk_email, no se enviará directamente sino que activará a la alerta que parametrizemos aquí y la encolará para enviar ese correo, este cambio es transparente para el funcionamiento de Libra, pero mejora los tiempos de respuesta a los usuarios, ya que se devuelve el control del equipo al usuario sin necesidad de que el correo electrónico esté enviado. Es necesario que esta alerta que se parametrice aquí sea específica para este cometido.

Por otro lado, tenemos que especificar la cadena de conexión de Libra en los campos reservados para ello.



IMPORTANTE: La cadena conexión XLSX tiene que ser de la forma ip:puerto:service_name o nombredns:puerto:service_name o ip:puerto/SID o nombredns:puerto/SID

Lo que no puede ser nunca es el nombre de la conexión tnsnames o una cadena que incluya el driver de la conexión con la BBDD.

Parametrización en el Generador de Informes

La idea detrás de esta funcionalidad es que podamos diseñar generadores de informes que saquen los filtros que se aplican al generador de Reports para que genere los PDF de los informes. Los informes que se usen tienen que cumplir una serie de restricciones, para evitar problemas:

Los report que se utilicen no pueden usar una tabla temporal que previamente haya que rellenar.

Datos del Informe | Configuración del Listado | Plug-in |

Informe **L_FACTURAS_GEN** Llamada a report de generación de facturas

		Imp. 1º Cabecera		24/03/2015 20:24:49	
		Orden Clasificación	Orden Impresión	Agrupar	Visible inicialmente
		Orden	Filtro	Subtotal	Total
				Valor por defecto	Ancho Máscara
Título Columna en Informe	SQL Columna				
P_EMPRESA		1			5
P_EJERCICIO		2			4
P_NUMERO_SERIE		3			3
P_NUMERO_FACTURA		4			10
L\$ESENIAPE	'HP_OFICINA_VIGO'	5			14
L\$DESTINATARIO_PARA1	'ernesto.gonzalez@edisac.com'	6			14
L\$ASUNTO	'Report de prueba'	7			14
L\$MODULE	'c:\adaptaciones\l_factura_gen.rdf'	8			14
L\$DESTINATARIO_PARA2	'desarrollodisa@gmail.com'	9			14
L\$ESTIPE	'PRINTER'	10			14

ID: 5 **Síntesis:** Vista Tabla

ID	Tabla	Columna	Tipo dato	Ancho	Ver campos	Objetos
5	FACTURAS_VENTAS	TIPO_FACTURA	VARCHAR2	5		+ TABLAS
6	FACTURAS_VENTAS	ORGANIZACION_COMERCIAL	VARCHAR2	5		+ VISTAS
7	FACTURAS_VENTAS	CLIENTE	VARCHAR2	15		+ SINONIMOS
8	FACTURAS_VENTAS	FECHA_FACTURA	DATE	10		
9	FACTURAS_VENTAS	FECHA_CONTABILIZACION	DATE	10		
10	FACTURAS_VENTAS	CODIGO_DIVISA	VARCHAR2	4		
11	FACTURAS_VENTAS	FORMA_COBRO	VARCHAR2	4		
12	FACTURAS_VENTAS	CENTRO_CONTABLE	VARCHAR2	4		
13	FACTURAS_VENTAS	DTOS_GLOBAL	NUMBER	5		
14	FACTURAS_VENTAS	IMP_DTO_GLOBAL	NUMBER	19		
15	FACTURAS_VENTAS	IMP_DTO_GLOBAL_DIV	NUMBER	19		
16	FACTURAS_VENTAS	DTO_PRONTO_PAGO	NUMBER	5		
17	FACTURAS_VENTAS	RECARGO_FINANCIERO	NUMBER	5		

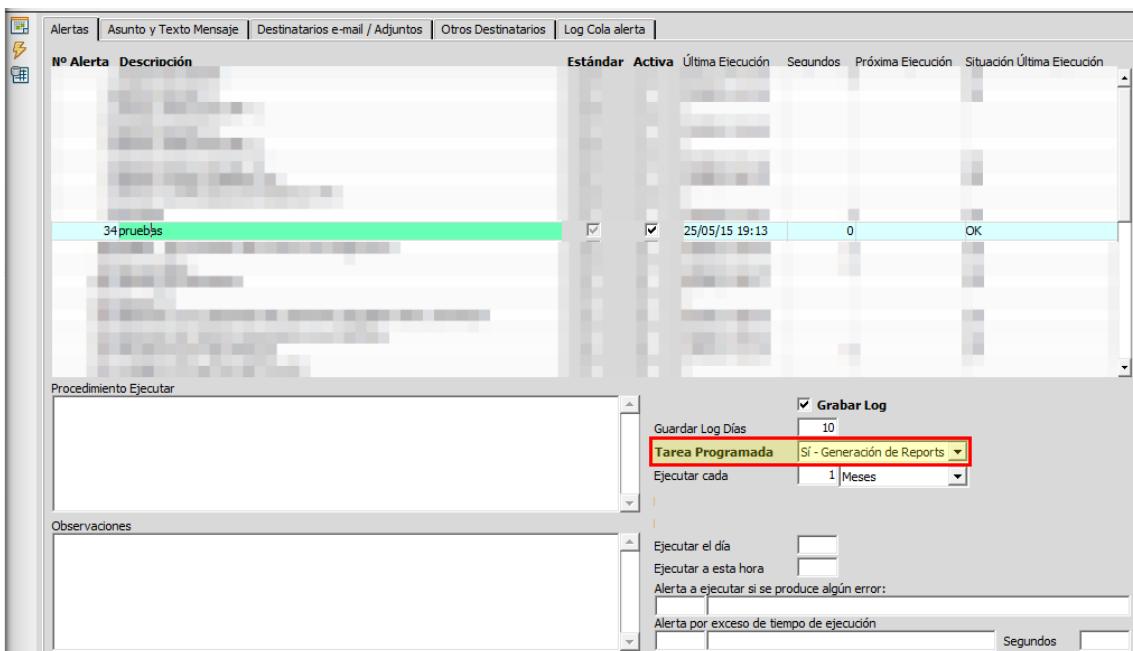
- Nombre del generador:** El generador puede tener cualquier nombre, pero si se usa el mismo nombre que el Report que queramos ejecutar, no tendremos que indicarlo con el registro L\$MODULE.
 - Filtros del Report:** Tenemos que parametrizar en los registros del generador de informes los filtros que luego se van a aplicar en el Report y cuyos valores saldrán de la consulta que parametrizaremos en el generador.
 - Parámetros L\$:** Existen algunos parámetros cuyo nombre empieza por L\$ que nos van a permitir definir acciones especiales a la hora de generar el Report, se detallan a continuación:
 - L\$DESTYPE:** Tipo de generación de Reports. FILE es el valor por defecto y el que se usa en el caso de generación por Galileo. Si queremos generar con Forms 11g obligatoriamente debemos parametrizar este registro con el valor CACHE. En el caso de querer imprimir automáticamente usaremos PRINTER, para ello deben estar parametrizadas las impresoras obligatoriamente en la máquina que genera los Reports, y debemos indicar la impresora con el parámetro L\$DESNAME.
 - L\$DESFORMAT:** Formato de salida del informe **(6.4.3+)**. Valores soportados:
 - PDF
 - XLSX (EXCEL ó ENHANCEDSPREADSHEET)
 - XLS (ó SPREADSHEET)
 - XML
 - L\$MODULE1,L\$MODULE2,....,L\$MODULEN:** Indica el nombre del formato que se va a utilizar para generar el Report. Si se indican varios parámetros L\$MODULE se van a intentar generar cada uno de esos informes y se adjuntarán todos en el mismo correo. Si no se alimenta se usa el nombre del generador de informes. Se pueden indicar informes del generador de informes de Libra indicando GI:XXXXXX, siendo XXXXXX el código de informe a ejecutar.
 - L\$DESNAME1,L\$DESNAME2,....,L\$DESNAMEN:** Nombre de cada archivo que se va a generar y adjuntar en el correo. Si se quiere imprimir por impresora, se tendrá que parametrizar el nombre de la impresora parametrizada en el servidor de Reports.
 - L\$USERID=** Conexión de base de datos que se va a usar para la generación del Report. No tendremos que configurarla si la hemos definido en parámetros de menú.
 - L\$P_COPIAS:** Número de copias que queremos generar. Por defecto es 1.

- **L\$DISPOSITIVO_GALILEO:** Indica el dispositivo del mantenimiento a_socket que se va a usar para generar el Report. Solo es obligatorio alimentarlo en el caso de que no se haya especificado el id_galileo del GAL_GEN
- **L\$PLSQL:** PLSQL que se ejecutará cuando la operación de generación del Report y envío haya sido satisfactoria. A este PL/SQL se le puede concatenar cualquiera de los campos de las tablas/vistas sobre las que consulta el Report. Ejemplo:

```
'BEGIN UPDATE ALBARAN_VENTAS SET FECHA_ENVIO_EMAIL = SYSDATE
WHERE NUMERO_SERIE='||ALBARAN_VENTAS.NUMERO_SERIE||' AND
NUMERO_ALBARAN='||ALBARAN_VENTAS.NUMERO_ALBARAN||' AND
EJERCICIO='||ALBARAN_VENTAS.EJERCICIO||' AND
EMPRESA='||ALBARAN_VENTAS.EMPRESA||'; COMMIT; END; '
```

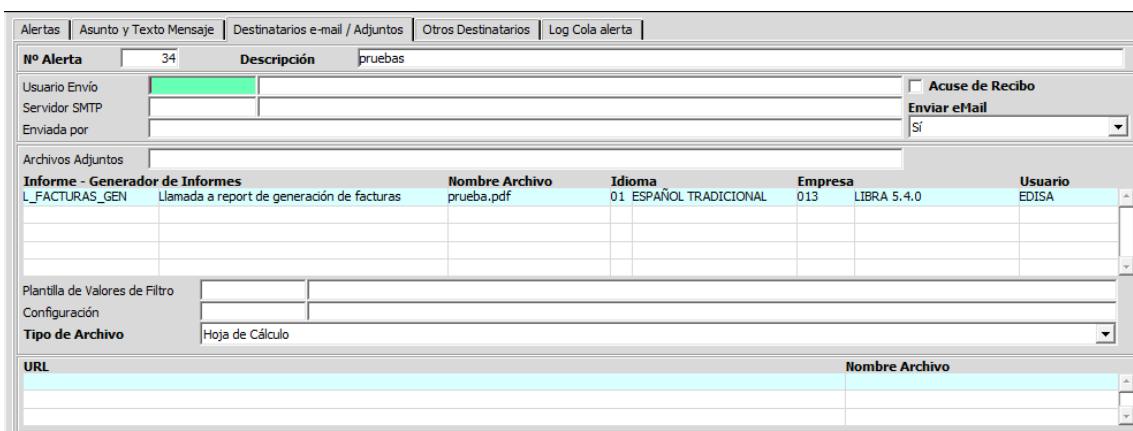
- **L\$PLSQL_FALLO:** PLSQL que se ejecutará cuando la operación de generación del Report y envío haya fallado.
- **L\$ASUNTO:** Asunto del correo que se va a enviar.
- **L\$PARSER_ASUNTO:** Si se pasa el valor 'S', al texto indicado en L\$ASUNTO se le aplicará el parser de forma que se puede reemplazar en tiempo de ejecución el contenido.
- **L\$CUERPO:** Cuerpo del correo que se va a enviar.
- **L\$PARSER CUERPO:** Si se pasa el valor 'S', al texto indicado en L\$CUERPO se le aplicará el parser de forma que se puede reemplazar en tiempo de ejecución el contenido.
- **L\$CUERPO_HTML:** Cuerpo del correo en formato HTML.
- **L\$PARSER CUERPO_HTML:** Si se pasa el valor 'S', al texto indicado en L\$CUERPO_HTML se le aplicará el parser de forma que se puede reemplazar en tiempo de ejecución el contenido. De esta forma en el cuerpo se podrían incrustar el resultado de ejecución de informes.
- **L\$EMAIL_REMITENTE:** Email que va a figurar como remitente de la alerta que se va a enviar. Si no se alimenta se va a usar el email_remitente parametrizado en la alerta desde la que se lanza el informe
- **L\$DESTINATARIO_PARA1,L\$DESTINATARIO_PARA2,...,**
L\$DESTINATARIO_PARAN: Direcciones que figurarán en el campo "PARA" del correo. Si no se especifica se usará el definido en la alerta que ejecuta el informe.
- **L\$DESTINATARIO_CC1,L\$DESTINATARIO_CC2,...,**
L\$DESTINATARIO_CCN: Direcciones que figurarán en el campo "CC" del correo. Si no se especifica se usará el definido en la alerta que ejecuta el informe.
- **L\$DESTINATARIO_BCC1,L\$DESTINATARIO_BCC2,...,**
L\$DESTINATARIO_BCCN: Direcciones que figurarán en el campo "BCC" del correo. Si no se especifica se usará el definido en la alerta que ejecuta el informe.

Alerta



The screenshot shows the configuration of an alert. At the top, there are tabs for 'Alertas', 'Asunto y Texto Mensaje', 'Destinatarios e-mail / Adjuntos', 'Otros Destinatarios', and 'Log Cola alerta'. The main area displays a table with columns: 'Nº Alerta', 'Descripción', 'Estándar', 'Activa', 'Última Ejecución', 'Segundos', 'Próxima Ejecución', and 'Situación Última Ejecución'. One row is highlighted in green, showing '34 pruebas' as the description. Below the table, there is a section titled 'Procedimiento Ejecutar' with a text input field. To the right, there are options for 'Grabar Log' (checked), 'Guardar Log Días' (set to 10), 'Tarea Programada' (set to 'Sí - Generación de Reports'), 'Ejecutar cada' (set to '1 Meses'), and other scheduling and monitoring options like 'Ejecutar el día', 'Ejecutar a esta hora', 'Alerta a ejecutar si se produce algún error', and 'Alerta por exceso de tiempo de ejecución'.

La forma de parametrizar la alerta es la misma que la que se usa para parametrizar cualquier alerta de Hoja de Cálculo. Captura de ejemplo de la parte donde se indica el informe a usar.



The screenshot shows the configuration of the 'Asunto y Texto Mensaje' tab for alert number 34. The 'Descripción' field contains 'pruebas'. The 'Archivos Adjuntos' section shows a table with columns: 'Informe - Generador de Informes', 'Nombre Archivo', 'Idioma', 'Empresa', and 'Usuario'. One row is highlighted in green, showing 'L_FACTURAS_GEN' as the report name, 'prueba.pdf' as the file name, '01 ESPAÑOL TRADICIONAL' as the language, '013 LIBRA 5.4.0' as the company, and 'EDISA' as the user. Below this, there are fields for 'Plantilla de Valores de Filtro', 'Configuración', 'Tipo de Archivo' (set to 'Hoja de Cálculo'), and a 'URL' section with a table for 'Nombre Archivo'.

Extras

Fuera de la propia lógica de alertas podemos usar estos generadores de informes para hacer ciertas cosas a nivel de PL/SQL. Un ejemplo sería, imprimir un Report en un equipo remoto por PL/SQL o generar un Report para guardar en Libra.

Enviar email por PLSQL con adjunto que sale de informe:

```

DECLARE
  p_informe      VARCHAR2 (30) := 'L_FACTURAS_GEN';
  p_idioma       VARCHAR2 (30) := '01';
  p_empresa       VARCHAR2 (30) := '013';
  p_usuario       VARCHAR2 (30) := 'EDISA';
  v_rdo          VARCHAR2 (30);
  v_urls          pk_b2b_geninf.tabla_urls;
  v_resultado     VARCHAR2 (30);
BEGIN
  v_rdo := pk_b2b_geninf.genera_urls_descarga
    (p_urls => v_urls,
     p_informe => p_informe,
     p_idioma => p_idioma,
     p_empresa => p_empresa,
     p_usuario => p_usuario,
     p_plantilla_valores_defecto => NULL,
     p_codigo_configuracion => NULL,
     p_userid => 'DESA04/DESA04@DESA11GUTF8',
     p_report => 'l_factura_gen',
     p_dispositivo_galileo => 'RWF11',
     p_destype => 'CACHE');

  FOR reg IN v_urls.FIRST .. v_urls.LAST LOOP
    pk_email.inicializar ('EDISA');
    pk_email.set_asunto ('PRUEBA');
    pk_email.set_cuerpo ('<texto plano del cuerpo del mensaje>');
    pk_email.add_destinatario ('TO', 'prueba@edisa.com');
    pk_email.add_adjunto_x_url(p_url => v_urls (reg).url,
                               p_nombre_archivo => 'archivo.pdf');
    v_resultado := pk_email.enviar ();
  END LOOP;
END;

```

Generar informe y descargarlo a blob para poder guardar lo en una tabla o hacer lo que se necesite (Este PL/SQL se usaría en el caso de querer imprimir)

```

DECLARE
  p_informe      VARCHAR2 (30) := 'L_FACTURAS_GEN';
  p_idioma       VARCHAR2 (30) := '01';
  p_empresa       VARCHAR2 (30) := '013';
  p_usuario       VARCHAR2 (30) := 'EDISA';
  v_rdo          VARCHAR2 (30);
  v_urls          pk_b2b_geninf.tabla_urls;
  v_resultado     VARCHAR2 (30);
  v_blob          BLOB;
BEGIN
  v_rdo := pk_b2b_geninf.genera_urls_descarga
    (p_urls => v_urls,
     p_informe => p_informe,
     p_idioma => p_idioma,
     p_empresa => p_empresa,
     p_usuario => p_usuario,
     p_plantilla_valores_defecto => NULL,
     p_codigo_configuracion => NULL,
     p_userid => 'DESA04/DESA04@DESA11GUTF8',
     p_report => 'l_factura_gen',
     p_dispositivo_galileo => 'RWF11',
     p_destype => 'CACHE');

  FOR reg IN v_urls.FIRST .. v_urls.LAST LOOP
    v_rdo := pk_blob2bd.carga_blob_desde_url(p_url => v_urls (reg).url,
                                              p_usuario => NULL,
                                              p_password => NULL);

    IF v_rdo = 'OK' THEN
      v_blob := pk_blob2bd.get_fichero;
    END IF;
  END LOOP;
END;

```

LATINOAMÉRICA

COLOMBIA

ECUADOR

MÉXICO

REP. DOMINICANA

ESPAÑA

MADRID

BARCELONA

VALENCIA

VIGO

OVIEDO

LAS PALMAS

OURENSE (CENTRO I+D)