

HP Service Manager

Para sistemas operativos Windows® y UNIX®

Versión de software: 9.30

Guía del Motor de documentos

Fecha de publicación del documento: 2011 de julio

Fecha de la versión del software: 2011 de julio



Avisos legales

Garantía

Las únicas garantías de los productos y servicios HP se exponen en el certificado de garantía que acompaña a dichos productos y servicios. El presente documento no debe interpretarse como una garantía adicional. HP no es responsable de omisiones, errores técnicos o de edición contenidos en el presente documento.

La información contenida en esta página está sujeta a cambios sin previo aviso.

Leyenda de derechos limitados

Software informático confidencial. Es necesario disponer de una licencia válida de HP para su posesión, uso o copia. De conformidad con FAR 12.211 y 12.212, el Gobierno estadounidense dispone de licencia de software informático de uso comercial, documentación del software informático e información técnica para elementos de uso comercial con arreglo a la licencia estándar para uso comercial de los proveedores.

Aviso de copyright

© Copyright 1994-2011 Hewlett-Packard Development Company, L.P.

Avisos de marca comercial

Adobe™ es una marca comercial de Adobe Systems Incorporated. Java™ y todas las marcas comerciales y logotipos de Java son marcas comerciales o marcas comerciales registradas de Sun Microsystems, Inc. en Estados Unidos y en otros países.

Java™ es una marca comercial de Estados Unidos de Sun Microsystems, Inc.

Microsoft® y Windows® son marcas comerciales registradas estadounidenses de Microsoft Corporation.

Oracle® es una marca comercial registrada estadounidense de Oracle Corporation, Redwood City, California.

UNIX® es una marca comercial registrada de The Open Group.

Actualizaciones de la documentación

La primera página de este documento contiene la siguiente información de identificación:

- Número de versión del software, que indica la versión del software.
- Fecha de publicación del documento, que cambia cada vez que se actualiza el documento.
- Fecha de la versión del software, que indica la fecha desde la que está disponible esta versión del software.

Para consultar las últimas actualizaciones o comprobar que está utilizando la edición más reciente de un documento, visite:

<http://h20230.www2.hp.com/selfsolve/manuals>

Este sitio requiere que esté registrado como usuario de HP Passport. Para ello, vaya a:

<http://h20229.www2.hp.com/passport-registration.html>

O pulse el enlace **New user registration** (Registro de nuevos usuarios) de la página de registro de HP Passport.

Asimismo, recibirá ediciones actualizadas o nuevas si se suscribe al servicio de soporte del producto correspondiente. Póngase en contacto con su representante de ventas de HP para obtener más información.

Soporte

Visite el sitio Web HP Software Support Online en:

<http://www.hp.com/go/hpsoftwaresupport>

Este sitio Web proporciona información de contacto y detalles sobre los productos, servicios y soporte que ofrece HP Software.

El soporte en línea de HP Software proporciona capacidades de resolución de problemas por parte de los propios clientes. Ofrece una forma rápida y eficaz de acceder a las herramientas de soporte técnico interactivo necesarias para gestionar su negocio. Puede beneficiarse de ser un cliente preferente de soporte utilizando el sitio Web de soporte para:

- Buscar documentos de interés en la base de conocimiento
- Enviar y realizar un seguimiento de los casos de soporte y las solicitudes de mejora
- Descargar parches de software
- Gestionar contratos de soporte
- Buscar contactos de soporte de HP
- Consultar la información sobre los servicios disponibles
- Participar en debates con otros clientes de software
- Investigar sobre formación de software y registrarse para recibirla

Para acceder a la mayor parte de las áreas de soporte es necesario que se registre como usuario de HP Passport. En muchos casos también será necesario disponer de un contrato de soporte. Para registrarse y obtener un ID de HP Passport, visite:

<http://h20229.www2.hp.com/passport-registration.html>

Para obtener más información sobre los niveles de acceso, visite:

http://h20230.www2.hp.com/new_access_levels.jsp

Contenido

Guía del Motor de documentos.....	1
Contenido.....	5
Visión general del Motor de documentos.....	9
¿Qué es el Motor de documentos?.....	9
Ventajas del enfoque modular.....	9
Coherencia.....	9
Tiempos de desarrollo reducidos.....	10
Flexibilidad.....	10
Acceso al Motor de documentos.....	10
Objetos.....	11
Creación y actualización de objetos.....	11
Descripciones de campos del formulario Definición del objeto.....	12
Descripciones de campos de la ficha Info. del objeto.....	13
Descripciones de campos de la ficha Bloqueo.....	15
Descripciones de campos de la ficha Revisiones.....	16
Descripciones de campos de la ficha Variables/Listas globales.....	16
Descripciones de campos de la ficha Actividades.....	16
Descripciones de campos de la ficha Alertas.....	17
Descripciones de campos de la ficha Aprobaciones.....	18
Descripciones de campos de la ficha Gestión de colas.....	21
Descripciones de campos de la ficha Vistas/Plantillas.....	22
Descripciones de campos de la ficha Notificaciones.....	22
Descripciones de campos de la ficha Configuración de búsqueda.....	23
Ficha Consultas definidas.....	23
Ficha Rangos.....	24
Estados.....	25
Búsqueda.....	25
Listas de registros.....	25

Visualización de un único registro.....	25
Exploración de un registro.....	25
Sugerencias de integración para la aplicación de visualización.....	26
Creación y actualización de estados.....	26
Descripciones de campos de Definición de estado.....	26
Procesos.....	29
Creación y modificación de un proceso.....	29
Descripciones de campos del formulario Definición del proceso.....	29
Ficha Expresiones iniciales.....	30
Ficha Javascript inicial.....	30
Ficha RAD.....	31
Ficha Expresiones finales.....	31
Ficha Javascript final.....	31
Ficha Proceso siguiente.....	32
Recursos del Motor de documentos.....	33
Objeto predeterminado (DEFAULT).....	33
Aplicaciones RAD.....	33
se.search.engine.....	34
se.list.engine.....	34
se.view.engine.....	34
Flujo de aplicaciones RAD.....	34
Sugerencias de integración para la aplicación de visualización.....	35
Funciones básicas en se.search.engine.....	36
Funciones básicas en se.view.engine.....	37
Funciones básicas en se.list.engine.....	40
Variables locales.....	40
Visión general de la solución de problemas.....	43
Investigación de la ruta de la aplicación a través del Motor de documentos.....	43
Búsqueda del dbdict u objeto utilizados.....	43
Búsqueda del estado de los registros.....	43
Búsqueda del nombre del proceso.....	44
Investigación de los errores de la aplicación.....	44

Impresión de valores de variables o resultados de expresiones.....	45
Visión general de un pedido de trabajo de muestra.....	47
Creación de una tabla.....	48
Adición de campos de claves a una tabla.....	49
Creación de un formulario.....	50
Creación de una copia de un formulario.....	51
Creación de un enlace para el formulario de pedido de trabajo.....	52
Creación de un archivo de números secuenciales.....	53
Creación de una definición de objeto.....	53
Creación de una definición de proceso de inicialización.....	55
Creación de definiciones de pantallas de visualización de la aplicación.....	55
Creación de definiciones de opciones de visualización de la aplicación.....	57
Creación de una definición de estado.....	63
Adición de un botón de cierre de pedido de trabajo.....	65
Creación de un asistente para el pedido de trabajo.....	67
Adición de un registro de definición de procesos.....	69
Creación de un formulario de entrada para el asistente.....	70
Modificación de los formularios de incidentes de cierre y actualización.....	72
Creación de un alias en la tabla probsummary para un enlace.....	74
Modificación de la definición de proceso im.set.close.....	75
Comprobación del pedido de trabajo de muestra.....	77

Capítulo 1

Visión general del Motor de documentos

El Motor de documentos es una herramienta de personalización que incluye la posibilidad de personalizar el sistema sin tener que recurrir a los cambios RAD. Proporciona un método centralizado para establecer los privilegios y el comportamiento para acciones estándar, como crear una lista, ver y buscar. Aumenta la coherencia entre módulos. Los tres componentes principales del Motor de documentos son Objetos, Estados y Procesos. Los procesos pueden reutilizarse para una mayor programación modular y la integración es perfecta, lo que permite reducir el tiempo de desarrollo.

¿Qué es el Motor de documentos?

El Motor de documentos incluye una serie de herramientas y metodologías para desarrollar y modificar los flujos de trabajo de Service Manager. En un principio, el Motor de documentos pretendía desarrollar una serie de funcionalidades básicas subyacentes que admitiesen varios módulos en Service Manager, mejorar la coherencia de la interfaz de usuario entre los módulos y reducir la cantidad de código necesario para cualquier módulo nuevo.

El Motor de documentos amplía las capacidades de la aplicación de visualización con acciones simplificadas y más extensas, especialmente aquellas que implican llamadas a varias aplicaciones. Asimismo, el Motor de documentos admite el uso de tablas join y llamadas de control de formato maestro. Se ha diseñado para satisfacer las necesidades de la mayoría de clientes de serie y a la vez conservar la flexibilidad. La relación entre objetos, estados y procesos es jerárquica.

El Motor de documentos controla el comportamiento con los objetos. Se hace referencia a un objeto cuando se abre un formulario y el objeto determina el comportamiento del estado del formulario (abrir, enumerar, buscar, etc.). Los objetos definen el comportamiento general de la tabla. En un objeto, un estado describe en qué etapa del ciclo de vida se encuentra un registro (abrir, enumerar, buscar, etc.). En el estado, se ejecutan diferentes procesos según las acciones iniciadas por un usuario en el registro. Los estados también definen cómo el sistema muestra un registro y qué opciones (acciones) están disponibles en momentos o circunstancias específicas. Por ejemplo, los estados pueden determinar una acción, como Guardar, a partir de los privilegios de acceso de un usuario.

Los procesos se llaman desde los estados en función de la acción del usuario. El proceso utiliza expresiones RAD, JavaScript y llamadas a aplicaciones RAD existentes para realizar acciones en el registro actual.

Ventajas del enfoque modular

Entre las ventajas del diseño modular se incluyen la coherencia en el desarrollo, un tiempo de desarrollo reducido y la flexibilidad.

Coherencia

El hecho que el motor permita que se ejecuten todas las aplicaciones usando las mismas aplicaciones RAD básicas aporta coherencia al conjunto de aplicaciones de Service Manager. La funcionalidad básica, como el bloqueo, las alertas, las aprobaciones y el uso de la funcionalidad de

la lista de registros, será igual para cualquier módulo, ya que todos utilizan la misma base de código.

Tiempos de desarrollo reducidos

La modularidad del Motor de documentos permite reutilizar el código existente y los procesos.

Flexibilidad

El Motor de documentos utiliza registros de proceso como mecanismo para modificar el comportamiento de los módulos dentro de las aplicaciones de Service Manager. Puede crear un nuevo proceso con un comportamiento diferente al del sistema básico, sin tener que cambiar o suprimir el proceso original del sistema. Además, los procesos básicos del sistema pueden anularse con sus propios procesos para ofrecer una mayor flexibilidad al desarrollador del sistema al personalizar Service Manager para que cumpla los requisitos específicos de una organización.

Acceso al Motor de documentos

Para acceder al Motor de documentos:

1. Inicie el Cliente de Service Manager y conéctese como administrador.
2. En el Navegador del sistema de Service Manager, pulse **Personalización**.
3. Pulse **Motor de documentos**. Desde aquí, puede acceder a las tres zonas principales para definir Objetos, Procesos o Estados. También puede configurar Alertas y Aprobaciones para que se invoquen desde Objetos o crear Registros de configuración para búsquedas, utilizados por Objetos.

Capítulo 2

Objetos

Los objetos son un conjunto básico de definiciones que determinan el comportamiento de los registros y establecen las definiciones y las normas vigentes. Los objetos corresponden uno a uno con los registros del Diccionario de base de datos (dbdict) de Service Manager. Si una tabla no tiene un registro de objeto dedicado, el Motor de documentos aplica la configuración en el objeto predeterminado DEFAULT. Todos los objetos deben tener definidos los estados predeterminados y de lista. Si no se suministran otros, los estados predeterminados son db.browse, db.list, db.search y db.view.

NOTA: no **modifique** ni **elimine** el registro de objeto predeterminado DEFAULT, ya que si lo hace podría obtener resultados impredecibles.

El registro de objeto configura las definiciones y las normas vigentes para el comportamiento de la tabla del Motor de documentos. Por ejemplo, estos incluyen:

- La aplicación utilizada para crear el perfil de los usuarios en este objeto que determina las acciones que este usuario debe llevar a cabo sobre cualquier registro de la tabla.
- Los registros de estado utilizados en circunstancias específicas (consulte la sección Estados para obtener más información).
- Los nombres de archivo de paginación, fase y categoría del objeto.
- El nombre del registro de número que debe utilizarse para este objeto.
- Cómo debe utilizar este objeto el bloqueo.
- Configuración de revisiones de registros en la tabla.
- Qué variables deben estar disponibles para los procesos que se ejecutan en este objeto.
- Qué listas globales deben estar siempre disponibles al utilizar este objeto.
- Uso de registros de actividades.
- Cómo se procesan las alertas en este objeto.
- Cómo se procesan las aprobaciones para este objeto.
- Configuración de las colas de trabajo.
- Capacidad de configuración de vistas personales o globales y plantillas predeterminadas.
- Notificaciones sobre la adición, actualización o eliminación de un registro de este objeto.
- Capacidad de configurar opciones de búsqueda adicionales para este objeto.

Para ver una lista de los objetos de serie, pulse **Buscar** en el formulario Definición del objeto.

Para ver una lista de los campos y descripciones de campos de las definiciones de objetos, consulte [Descripciones de campos del formulario Definición del objeto en la página 12](#).

Creación y actualización de objetos

Para crear un objeto:

1. Vaya al Motor de documentos. Consulte [Acceso al Motor de documentos en la página 10](#) para ver los pasos.
2. Pulse dos veces **Objetos**. Se abrirá el formulario Definición del objeto.
3. Utilice las fichas del formulario para rellenar los campos necesarios para crear un objeto que realizará las funciones que desee. Consulte las descripciones de campo para obtener más información.

Para actualizar una definición de objeto:

1. Acceda al Motor de documentos. Consulte [Acceso al Motor de documentos en la página 10](#) para ver los pasos.
2. Introduzca el nombre del objeto que desea actualizar en el campo Nombre del archivo o pulse **Buscar** para buscar el objeto.

Descripciones de campos del formulario Definición del objeto

Las descripciones de campos del formulario Definición del objeto son:

Nombre del campo	Descripción
Nombre del archivo <i>file.name</i>	Permite introducir el nombre de dbdict para el objeto; es decir, utilice el nombre de dbdict que corresponda con el nombre de este objeto. (Obligatorio)
Nombre común <i>message</i>	El sistema rellena esta información a partir de la Política de datos. Se trata de un nombre común para el objeto. El nombre común puede ser un nombre sencillo, como Pedido de trabajo.
Clave única <i>unique.field</i>	El sistema rellena esta información a partir del Diccionario de base de datos. Se trata de una clave única para el objeto.

Este formulario también incluye las siguientes fichas. Los campos de estas fichas se describen en las tablas de descripciones de campos de la ficha respectiva.

- [Descripciones de campos de la ficha Info. del objeto en la página 13](#): especifica las propiedades generales y el comportamiento del objeto.
- [Descripciones de campos de la ficha Bloqueo en la página 15](#): determina el comportamiento de bloqueo del objeto.
- [Descripciones de campos de la ficha Revisiones en la página 16](#): rastrea las revisiones del objeto.
- [Descripciones de campos de la ficha Variables/Listas globales en la página 16](#): describe las variables locales y las variables globales utilizadas por el objeto.

- [Descripciones de campos de la ficha Actividades en la página 16](#): define el registro (actividades).
- [Descripciones de campos de la ficha Alertas en la página 17](#): define dónde establecer alertas y las condiciones para generarlas.
- [Descripciones de campos de la ficha Aprobaciones en la página 18](#): establece las opciones de aprobación del objeto.
- [Descripciones de campos de la ficha Gestión de colas en la página 21](#): controla cómo se muestran las colas y los threads, y quién puede crear vistas (bandejas de entrada).
- [Descripciones de campos de la ficha Vistas/Plantillas en la página 22](#): define si un usuario puede crear plantillas y vistas globales y personales en el objeto.
- [Descripciones de campos de la ficha Notificaciones en la página 22](#): identifica las notificaciones enviadas automáticamente para una actividad Añadir, Actualizar o Eliminar de cualquier registro del objeto.
- [Descripciones de campos de la ficha Configuración de búsqueda en la página 23](#): controla las opciones disponibles en la ficha Más opciones de la pantalla de búsqueda.

Descripciones de campos de la ficha Info. del objeto

Esta ficha especifica las propiedades generales y el comportamiento del objeto.

Las descripciones de campos de la ficha Info. del objeto son:

Nombre del campo	Descripción
Campo de descripción <i>desc.field</i>	Especifica una breve descripción del objeto.
Aplicación del perfil <i>profile.appl</i>	Especifica la aplicación RAD que crea el perfil que determina si un usuario puede realizar ciertas funciones, como añadir y eliminar. Por ejemplo, db.environment. (Obligatorio)
Variable del perfil <i>profile.variable</i>	Especifica una variable a la que puede accederse siempre que se llame a este objeto sin acceder al registro de entorno. Por ejemplo, \$.env. (Obligatorio)
Nombre del registro de número <i>number.record</i>	Define una clase de número del objeto, al que puede accederse desde una llamada a la aplicación getnumb a través de una llamada de proceso, RAD o Control de formatos. Puede utilizarse para recuperar un número secuencial como clave única para el registro. Por ejemplo, EXWorkOrder.
Nombre de la tabla de categorías <i>category.file.name</i>	Especifica el nombre de la tabla que enlaza con la tabla de categorías asociada al valor de categoría de este objeto. Al mostrar un registro de este tipo, si existe un campo llamado categoría, el objeto buscará el nombre de la tabla de categorías y el registro con el nombre correspondiente. Si lo encuentra, el sistema almacenará el nombre de archivo de categoría como una variable: \$.category.

Nombre del campo	Descripción
Nombre de la tabla de fases <i>phase.file.name</i>	Especifica el nombre de la tabla que enlaza con la tabla de fases asociada a este objeto, si es aplicable. Cuando el sistema muestra un registro de este tipo, si existe un campo llamado fase, el objeto irá al nombre de la tabla de fases y seleccionará un registro con el nombre correspondiente. Si lo encuentra, el sistema almacenará el nombre de la tabla de fase como una variable: \$L.phase.
Nombre de la tabla de paginación <i>paging.file</i>	Especifica el nombre de la tabla utilizada para almacenar las páginas. Las páginas se crean como copias completas del registro actual antes de que se apliquen las últimas actualizaciones. Esto se realiza cada vez que se actualiza un registro, lo que crea un registro de auditorías detallado.
Control de formatos maestro <i>master.fc</i>	Especifica el nombre del registro Control de formatos maestro, si existe uno para el registro de objeto. Control de formatos maestro le permite definir en un registro las instrucciones de Control de formatos que se aplican a todas las fases de un área; como por ejemplo, las fases de peticiones de Gestión de cambios. Generalmente, el nombre del control de formatos maestro es el nombre del dbdict o todas las categorías de Gestión de incidentes, Gestión de problemas o Centro de servicios.
Joindef <i>joindef</i>	Especifica el nombre del registro joindef utilizado para unir varias tablas (como por ejemplo, joincomputer) o una expresión que se evalúa para un nombre joindef válido, como joindef en \$L.category.
Campo de estado <i>statusField</i>	Especifica el nombre de campo que contiene la información de estado del registro.
Asignado a campos <i>assignedToFields</i>	Especifica el nombre de campo que contiene el campo Nombre del asignatario de este objeto. Se consultará este campo cuando Autorización de carpeta verifique que el registro se ha asignado al operador conectado.
Campos del grupo de trabajo <i>workgroupFields</i>	Especifica el nombre de campo que contiene el campo Grupo de asignación de este objeto. Se consultará este campo cuando Autorización de carpeta verifique que el registro se ha asignado a cualquier grupo de trabajo del operador conectado.
Estado de la apertura <i>open.state</i>	Especifica el registro de definición de estado que se utiliza para abrir un nuevo registro.
Estado del cierre <i>close.state</i>	Especifica el registro de definición de estado que se utiliza para el procesamiento de cierre de un registro existente.
Estado de la lista <i>list.state</i>	Especifica el registro de definición de estado utilizado para mostrar una lista de registros.

Nombre del campo	Descripción
Estado predeterminado <i>default.state</i>	Especifica el nombre del estado utilizado como predeterminado para el objeto. Para editar un registro en este objeto, se utiliza el estado predeterminado.
Estado de la búsqueda <i>search.state</i>	Especifica el registro de definición de estado que se utiliza para buscar.
Estado de exploración <i>browse.state</i>	Define el registro de definición de estado que se utiliza cuando los registros utilizan el bloqueo. Básicamente, este campo define un estado de sólo lectura si otro usuario bloquea un registro actualmente.
Estados manuales <i>manual.states</i>	Especifica la matriz de estados que deben utilizarse con este objeto que no sean los estados de ciclo de vida de abrir, cerrar, enumerar, ver, buscar o examinar.

Descripciones de campos de la ficha Bloqueo

La ficha Bloqueo determina el comportamiento de bloqueo del objeto. Esto significa que cuando alguien está actualizando activamente un registro el sistema no permite que otro usuario actualice el mismo registro.

Las descripciones de campos de la ficha Bloqueo son:

Campo	Descripción
Usar bloqueo <i>use.locking</i>	Seleccione esta casilla de verificación para habilitar el bloqueo.
Bloquear al mostrar <i>lock.on.display</i>	Si es true o una condición que se evalúa como true, Service Manager intenta bloquear el registro tan pronto como se muestre. Debe habilitarse el uso del bloqueo para que este campo funcione.
Bloquear registro de nivel superior <i>lock.parent</i>	Bloquea el registro actual y el de nivel superior. Por ejemplo, si se selecciona este campo y alguien está actualizando una tarea de cambio, la petición de cambio para la tarea también se bloqueará.
Campo ID del nivel superior <i>parent.id</i>	Permite introducir un nombre de campo del registro actual que contenga el ID del nivel superior para bloquear.
Nombre del archivo/objeto de nivel superior <i>parent.object</i>	Nombre de la tabla que contiene el registro de nivel superior.
Variables de cronómetro <i>watch.variables</i>	Las variables de cronómetro se utilizan cuando el Motor de documentos comprueba si se ha cambiado un registro o no. Estas

Campo	Descripción
	variables deben ser NULL cuando se muestra el registro por primera vez.

Descripciones de campos de la ficha Revisiones

La ficha Revisiones determina el comportamiento de revisión del objeto.

Las descripciones de campos de la ficha son:

Nombre del campo	Descripción
Nombre de la tabla de revisiones <i>revision.file</i>	Especifica el nombre de la tabla en donde desea almacenar las revisiones de los registros de este objeto.
Cant. máx. de revisiones <i>max.revisions</i>	Especifica el número total de revisiones permitidas para este objeto. Si se deja en blanco, no hay ningún límite superior.

Descripciones de campos de la ficha Variables/Listas globales

La ficha Variables/Listas globales describe las variables locales y las variables globales utilizadas por el objeto.

Nota: las variables globales se crean y almacenan en la memoria.

Las descripciones de campos de la ficha Variables/Listas globales son:

Nombre del campo	Descripción
Variables locales <i>local.variables</i>	Permite introducir una lista de variables locales que pueden utilizarse en los procesos. Las variables locales se definen en la aplicación, se asignan al objeto que está creando y están disponibles para todos los procesos y estados asociados al objeto. Las variables locales no están disponibles para otros objetos.
Listas globales <i>global.lists</i>	Una vez creadas las listas globales, están disponibles para todos los procesos. Las listas globales pueden crearse en la conexión si aparecen en la lista de inicio. Las listas globales están disponibles cada vez que accede al objeto.

Descripciones de campos de la ficha Actividades

La ficha Actividades define el registro de actualizaciones (Actividad) para el Objeto.

Las descripciones de campos de la ficha Actividades son:

Nombre del campo	Descripción
Tabla de registro de actividades <i>activitylog.file.name</i>	Nombre de la tabla para guardar las entradas del registro de actividades del Objeto.
Variable de lista de selección <i>activity.selection.var</i>	Variable para utilizar en un formulario de actualización que permite mostrar los tipos de actividades que puede seleccionar el operador al efectuar una actualización en un registro para un Objeto específico.
Enlace de anotación <i>activity.post.link</i>	Nombre del enlace utilizado para anotar información.
¿Requiere actualización si NO se genera un registro de actividad? <i>activity.mandatory</i>	Casilla de verificación que si se selecciona indica que se requiere la actualización de una actividad.
Campo de actualización <i>update.field.var</i>	Campo o variable que contiene la actualización de una actividad en el formulario. Este campo sólo aparece si se incluye ¿Requiere actualización si NO se genera un registro de actividad? .
Mensaje de visualización <i>activity.mandatory.msg</i>	Mensaje que indica que se requiere la actualización de una actividad. Este campo sólo aparece si se incluye ¿Requiere actualización si NO se genera un registro de actividad? .

Descripciones de campos de la ficha Alertas

La ficha Alertas define dónde establecer alertas y las condiciones para generarlas. Cualquier objeto con una clave única puede utilizar estas alertas.

Las descripciones de campos de la ficha Alertas son:

Nombre del campo	Descripción
Ubicación de la alerta <i>alert.location</i>	<p>Ubicación de la alerta define dónde se encuentra o almacena el nombre de la definición de alerta para ejecutar. Especifica dónde se almacena la definición de alerta:</p> <p>Registro: almacena alertas en el mismo registro.</p> <p>Categoría: almacena alertas en el archivo de categoría definido en la ficha Info. del objeto.</p> <p>Fase: almacena alertas en el registro Fase definido en la ficha Info. del objeto.</p> <p>Objeto: almacena alertas en el registro Objeto. Si se selecciona,</p>

Nombre del campo	Descripción
	<p>se mostrará una tabla Alertas que le permitirá rellenar las alertas que deben utilizarse.</p> <p>Nota: la tabla Objeto debe tener una clave única para que las alertas puedan relacionarse con ella. Una tabla que contenga una clave Sin nulos en vez de una clave Únicos no puede utilizar Alertas.</p>
Condición de alerta <i>alert.condition</i>	Especifica una condición para determinar si debe procesarse la alerta o no. Por ejemplo, open in \$L.file~=false.
Nombre del campo de alerta <i>alert.field.name</i>	Especifica el nombre de campo que contiene el nombre de alerta real, como se define en Ubicación de la alerta.
Campo de estado de alerta <i>alert.status.field</i>	Especifica el campo del registro actual donde colocar el estado de la alerta, una vez procesada la alerta.
Proceso de actualización de la alerta <i>alert.update.process</i>	Especifica el nombre del registro de proceso para funciones adicionales que realiza el sistema tras ejecutarse la alerta.
¿Registrar las alertas? <i>log.alerts</i>	Si se selecciona, las alertas se mueven tras procesarse al archivo de registro de alertas para conservar un histórico de alertas.
¿Procesar alertas en el nivel superior? <i>alerts.against.parent</i>	Si ha seleccionado el campo Bloquear registro de nivel superior en la ficha Bloqueo y selecciona esta casilla de verificación, la alerta se registrará también contra el registro de nivel superior, cuando se activa.
Volver a calcular alertas si <i>alert.recalc</i>	Especifica las condiciones que determinan si deben volver a calcularse las condiciones de las alertas existentes o no.
Restablecer alertas si <i>alert.reset</i>	Determina cuándo deben eliminarse las alertas existentes y volver a calcular todas las condiciones.

Descripciones de campos de la ficha Aprobaciones

La ficha Aprobaciones establece opciones de aprobación y sus notificaciones asociadas para el objeto. Las aprobaciones se definen en el archivo ApprovalDef.

Las descripciones de campos de la ficha Aprobaciones son:

Nombre del campo	Descripción
Condición de aprobación	Si la condición de aprobación se evalúa como true, las aprobaciones se utilizan en los registros de objeto.

Nombre del campo	Descripción
<i>approval.condition</i>	
Ubicación de aprobación <i>approval.location</i>	Indica dónde se almacena la información de la aprobación: registro, fase, objeto o categoría.
Nombre de campo de aprobación <i>approval.field.name</i>	Nombre de campo que contiene el nombre de la aprobación real en la tabla que se define en Ubicación de aprobación.
Campo de estado de aprobación <i>approval.status.field</i>	Campo del registro actual donde se almacena el estado de aprobación.
Grupos de aprobación <i>approval.groups</i>	Almacena una variable que contenga los grupos al que debe pertenecer el usuario actual para emitir aprobaciones para este objeto.
Tipo de aprobación <i>appr.cond.type</i>	Existen cuatro tipos de aprobación predefinidos. Todos deben aprobar: todos los grupos/operadores definidos en la definición de aprobaciones deben emitir una aprobación antes de que el sistema establezca el estado del registro en aprobado. Si sólo uno o varios grupos/operadores (pero no todos) emite una aprobación, en este caso el estado se establece en pendiente. Uno debe aprobar: el registro se aprueba con una aprobación de cualquier miembro del grupo/operador de aprobación. Quórum: el registro se aprueba cuando la mayoría de los miembros del grupo de aprobación indica la aprobación. Todos deben aprobar - denegación inmediata: todos los grupos/operadores deben aprobar el registro. La primera denegación hace que el estado cambie a Denegar. El resto de aprobadores no deben realizar ninguna otra acción.
Notificación de aprobación <i>single.notify.approval</i>	Selecciona la notificación que se ejecuta si la petición se ha aprobado.
Notificación de denegación <i>single.notify.denial</i>	Selecciona la notificación que se ejecuta si un aprobador deniega la petición.
Notificación de retractación <i>single.notify.retraction</i>	Selecciona la notificación que se ejecuta al retractar una acción anterior.

Nombre del campo	Descripción
Notificación de aprobación final <i>final.notify.approval</i>	Selecciona la notificación para enviar una vez que se ha concedido la aprobación final.
Notificación de denegación final <i>final.notify.denial</i>	Selecciona la notificación para enviar si la petición se ha denegado.
FC de aprobación <i>appr.fc</i>	Especifica el nombre del registro de Control de formatos que se ejecuta en la aprobación.
Proceso de aprobación <i>approval.process</i>	Selecciona el proceso que se ejecuta cuando el registro se ha aprobado.
Proceso de denegación <i>denial.process</i>	Selecciona el proceso que se ejecuta cuando el registro se ha denegado.
Aprobación previa a la apertura <i>preapprove.cond</i>	Determina si el registro debe aprobarse automáticamente o no. Si la condición es true y el usuario pertenece a uno de los grupos de aprobación pendientes, la aprobación se procesa automáticamente. Pero si el usuario no pertenece a ninguno de estos grupos, la aprobación no ocurre automáticamente y debe someterse al proceso de aprobación habitual. De forma predeterminada, se establece en true.
¿Registrar aprobaciones? <i>log.approvals</i>	Si se selecciona esta casilla de verificación, podrá registrar el histórico de aprobaciones en la tabla ApprovalLog.
¿Requiere comentarios de aprobación? <i>approval.comments</i>	Si se selecciona, el aprobador solicita comentarios de aprobación.
¿Totalizar aprobaciones? <i>aggregate.approvals</i>	Si se selecciona, las aprobaciones se acumulan.
Volver a calcular las aprobaciones si <i>approval.recalc</i>	Especifica las condiciones que determinan si deben volver a calcularse las condiciones en las aprobaciones existentes o no.
Restablecer las aprobaciones si <i>approval.reset</i>	Determina cuándo deben eliminarse las aprobaciones existentes y volver a calcular todas las condiciones.

Descripciones de campos de la ficha Gestión de colas

La ficha Gestión de colas controla cómo se visualizan las colas y vistas, así como los threads. Estos mismos campos están disponibles en la Política de datos para aquellos archivos que no tienen un registro de objeto asociado o que están unidos virtualmente en el registro de datadict del registro de objeto, si existe.

Las descripciones de campos de la ficha Gestión de colas son:

Nombre del campo	Descripción
Gestionar condición <i>scm.condition</i>	Especifica una condición que sólo permite a ciertos usuarios ver las colas que muestran registros de este objeto. Por ejemplo, vaya a \$G.pm.environment.
Gestionar formulario de visualización <i>scm.manage.screen</i>	Permite seleccionar el formulario que se utilizará para mostrar la vista. La aplicación Service Manager de serie tiene un formulario de visualización predeterminado: sc.manage.generic que se utiliza si no se elige ninguna otra forma. HP recomienda no cambiar el formulario sc.manage.generic.
Gestionar vista predeterminada <i>scm.inbox</i>	Permite seleccionar la vista predeterminada de esta cola. Al especificar una vista de usuario para un usuario particular, puede configurarse una lista de vistas específica para Gestión de colas de HP Service Manager. Si un usuario no tiene una vista específica definida, se utilizará la vista de usuario predeterminada.
Gestionar consulta predeterminada <i>scm.query</i>	Especifica una consulta predeterminada para ejecutar si no se selecciona ninguna vista predeterminada.
Descripción de consulta predeterminada <i>scm.query.name</i>	Especifica un nombre para el campo anterior. Puede asociar un mensaje a este campo. Por ejemplo, scmsg(491, "us").
Vista de threads -> ¿Desea buscarlos? <i>scm.thread.list.edit</i>	Especifica true o una expresión que se evalúa como true para abrir un nuevo thread al realizar una búsqueda.
Formulario de búsqueda (si fuera necesario) <i>scm.search.format</i>	Permite seleccionar un formulario de búsqueda predeterminado.
Búsqueda de threads -> ¿Desea enumerarlos? <i>scm.thread.search.list</i>	Especifica true o una expresión que se evalúa como true para un nuevo thread cuando un usuario encuentra una lista de registros para ver.
Lista de threads -> ¿Desea editarlos? <i>scm.thread.list.edit</i>	Especifica true o una expresión que se evalúa como true para un nuevo thread cuando un usuario selecciona un registro para ver entre una lista de registros.

Nombre del campo	Descripción
Vista de threads -> ¿Desea editarlos? <i>scm.thread.inbox.edit</i>	Especifica true o una expresión que se evalúa como true para abrir un nuevo thread cuando el usuario ve un registro existente en la cola.
Permitir condición de adición <i>scm.add.condition</i>	Especifica una expresión que evalúa la capacidad de añadir un registro de un operador.
Añadir/Abrir aplicación <i>scm.add.appl</i>	Especifica el nombre de la aplicación para llamar cuando se añade o abre un registro.
Nombres de parámetros <i>scm.add.names</i>	Especifica los nombres de parámetros para pasar a la aplicación especificados en el campo Añadir/Abrir aplicación.
Valores de parámetros <i>scm.add.values</i>	Especifica los valores de parámetros para pasar a la aplicación especificados en el campo Añadir/Abrir aplicación.

Descripciones de campos de la ficha Vistas/Plantillas

La ficha Vistas/Plantillas define si un usuario puede crear vistas globales y personales, así como plantillas o no.

Las descripciones de campos de la ficha son:

Nombre del campo	Descripción
Crear vistas personales <i>personal.inbox</i>	Especifica una condición que se evalúa como true o false. True permite al usuario crear vistas personales.
Crear vistas de sistema <i>global.inbox</i>	Especifica una condición que se evalúa como true o false para determinar si el usuario puede crear vistas globales.
Plantilla predeterminada <i>default.template</i>	Especifica el nombre de la plantilla predeterminada que debe utilizarse para los registros de esta tabla.
¿Admite plantillas? <i>supportTemplates</i>	Seleccione esta casilla de verificación para habilitar la asistencia para crear plantillas para el objeto.

Descripciones de campos de la ficha Notificaciones

La ficha Notificación identifica las notificaciones enviadas automáticamente para una actividad Añadir, Actualizar o Eliminar del objeto.

Las descripciones de campos de la ficha Notificaciones son:

Nombre del campo	Descripción
Añadir <i>notification.add</i>	Permite seleccionar la notificación que se envía automáticamente cuando un registro se añade a la tabla.
Actualizar <i>notification.update</i>	Permite seleccionar la notificación que se envía automáticamente cuando un registro se actualiza en la tabla.
Eliminar <i>notification.delete</i>	Permite seleccionar la notificación que se envía automáticamente cuando un registro se elimina de la tabla.

Descripciones de campos de la ficha Configuración de búsqueda

La ficha Configuración de búsqueda controla las opciones disponibles en la ficha Más opciones de la pantalla de búsqueda.

Las descripciones de campos de la ficha Configuración de búsqueda son:

Nombre del campo	Descripción
Nombre de la tabla <i>tablename</i>	El nombre de la tabla que se consultará.
Formulario de búsqueda <i>searchFormat</i>	El nombre del subformulario utilizado para la ficha Más opciones.
Proceso de inicialización <i>init.process</i>	El nombre de un proceso que se ejecutará antes de mostrar el formulario de búsqueda.
Permitir búsqueda avanzada <i>allowAdvAccess</i>	Define las condiciones que determinan si debe admitirse o no la búsqueda avanzada.

Ficha Consultas definidas

La ficha Consultas definidas define las etiquetas e instrucciones de consulta para utilizar en la ficha Más opciones de las pantallas de búsqueda (por ejemplo, Más opciones en el formulario Búsqueda de incidentes). Los campos se definen en la tabla SearchConfig.

Las descripciones de campos de la ficha Consultas definidas son:

Nombre del campo	Descripción
ID <i>id</i>	Proporciona la ID única para la consulta y no puede incluir caracteres especiales, incluidos los espacios.
Consulta <i>query</i>	Expresión de consulta que utiliza una sintaxis del lenguaje del sistema. Por ejemplo, <code>assignee.name=operator()</code> .
Descripción <i>description</i>	Proporciona la etiqueta para la casilla de verificación en la ficha Más opciones.

Ficha Rangos

La ficha Rangos permite configurar fácilmente una búsqueda de rangos de fecha de inicio y finalización. Para ello, pulse el enlace **Modificar configuración** para definir una variable de inicio y finalización en el formulario. Introduzca esta variable en la columna Variable 1 y utilice las columnas Campo y Operador 1 para definir la consulta que se ejecutará. Modificar configuración también define cómo se han hecho las modificaciones en la ficha Consultas definidas.

Las descripciones de campos de la ficha Rangos son:

Nombre del campo	Descripción
Campo <i>fieldName</i>	Especifica el campo de la tabla que debe utilizarse en la consulta.
Operador 1 <i>operator1</i>	El operador de comparación en la consulta. (Por ejemplo: >=)
Variable 1 <i>variable1</i>	Especifica la variable utilizada en el formulario.
Tipo especial <i>specialType</i>	No se utiliza por el momento.

Capítulo 3

Estados

Los estados son llamados por los objetos y son definidos por los procesos. El registro de estado contiene información sobre cómo se visualiza y cómo actúa un registro en un período de tiempo específico.

Una definición de registro de estado puede incluir lo siguiente:

- El nombre de estado.
- La pantalla de visualización que debe utilizarse para que se muestren los registros.
- El proceso de inicialización utilizado cuando se visualiza primero una lista.
- El formato que se utilizará para mostrar los datos.
- La condición de entrada para indicar si un usuario puede modificar el registro o no.
- El proceso que debe ejecutarse cuando un usuario activa una opción de visualización específica (Acción de visualización).

El registro de estado utilizado depende del número de registros que ve el usuario. Existen registros de estado de serie para buscar registros, visualizar una lista, visualizar un único registro y explorar un registro.

Búsqueda

Si no hay registros en la variable de archivo actual, se asume que el usuario está en el modo de búsqueda. Se utiliza el estado de búsqueda definido en el registro de objeto de la tabla. El estado de búsqueda predeterminado es "db.search".

Listas de registros

Al visualizar una lista de registros, se utiliza el estado de la lista definido en el registro de objeto de la tabla. El estado de la lista predeterminado es "db.list".

Nota: en algunas versiones anteriores de Service Manager, las listas de registros se conocen como listas QBE.

Visualización de un único registro

Al visualizar un único registro, el registro se comprueba primero para ver si incluye un campo "Estado" en el dbdict. Si el campo existe y se rellena, el contenido de ese campo se utilizará como el estado actual del registro. Si este campo no existe o es NULL, se utiliza el estado predeterminado definido en el registro de objeto de la tabla. El valor predeterminado del estado predeterminado es "db.view".

Exploración de un registro

Al explorar un registro en modo de sólo lectura, se utiliza el estado de exploración definido en el registro de objeto de la tabla. Tenga en cuenta que sólo los archivos que utilizan el bloqueo necesitan un estado de exploración. Aunque la visualización de un registro sin tener derechos de

actualización parece similar a la pantalla de exploración, es utilizando el estado predeterminado, no el estado de exploración. No existe ningún estado de exploración predeterminado.

Sugerencias de integración para la aplicación de visualización

Si una pantalla de visualización no tiene ningún evento de visualización asociado a ella, el sistema utiliza automáticamente el evento de visualización `se.default` que realiza `se.lock.object` en el caso de "OnFormModified". Si un registro de opción de visualización tiene la casilla de verificación "Modifica el registro" activada, el registro se bloquea al pulsar el botón.

Creación y actualización de estados

Para crear un nuevo estado:

1. Acceda al Motor de documentos. Consulte [Acceso al Motor de documentos en la página 10](#).
2. En el formulario Definición de estado, rellene los campos necesarios para crear un estado y realizar las funciones que desee. Consulte las descripciones de campos en [Descripciones de campos de Definición de estado en la página 26](#)

Para modificar un estado existente:

1. Acceda al Motor de documentos. Consulte [Acceso al Motor de documentos en la página 10](#).
2. Introduzca el nombre del estado que desea modificar en el campo Estado o pulse **Buscar** para buscar el estado.

Descripciones de campos de Definición de estado

El registro Definición de estado define el comportamiento y las especificaciones de visualización de un registro en un período de tiempo específico.

En el registro de estado los métodos generales pueden definirse o el comportamiento de los métodos básicos puede modificarse. Los métodos básicos se describen en las secciones de funciones básicas de `se.search.engine`, de `se.view.engine` y de `se.list.engine`. Incluyen funciones como Guardar, Encontrar, Rellenar, Aceptar, Cancelar y Buscar.

Las descripciones de campos del formulario Definición de estado son:

Nombre del campo	Descripción
Estado <i>state</i>	Especifica el nombre del estado. (Obligatorio)
Pantalla de visualización <i>display.screen</i>	La pantalla de visualización que debe asociarse al estado.
Proceso de inicialización	El nombre de un proceso que debe ejecutarse antes de introducir el estado.

Nombre del campo	Descripción
<i>init.process</i>	
Formato <i>format.name</i>	Especifica el formulario en que se abre el registro de estado. Este formulario se almacena en un registro y puede ser una variable o estar codificado. El nombre del formulario puede codificarse o incluirse en una variable o recuperarse de un registro a través de una expresión de lenguaje del sistema.
Condición de entrada <i>input.condition</i>	La condición de entrada se evalúa para visualizar estados y determina si el registro es de sólo lectura o no. Especifica false si es de sólo lectura o true si puede editarse. También puede introducir una expresión que se evalúe en true o false. Sino, introduzca true o una expresión que se evalúe en true o false.
Métodos generales	
Acción de visualización <i>process.label</i>	Especifica el parámetro de acción procedente de una acción de visualización tras los datos introducidos de un usuario. La acción de visualización procede del campo de acción en la opción de visualización de un botón o elemento de menú pulsado por un usuario.
Nombre del proceso <i>valid.process</i>	Especifica el nombre del proceso ejecutado como resultado de la acción.
Condición <i>process.condition</i>	Especifica una expresión que se evalúa en true para el proceso especificado en el nombre de proceso que debe llamarse.
Guardar primero <i>run.save.before</i>	Permite ejecutar el proceso para guardar antes de ejecutar este proceso. Escriba true para guardar primero; de lo contrario, introduzca false. El valor predeterminado es false.

Capítulo 4

Procesos

Los procesos son las unidades de trabajo discretas más pequeñas disponibles en el Motor de documentos y son el nivel donde se manipulan los datos. Los usuarios pueden crear su propio proceso o utilizar uno de los más de 700 procesos que se envían con Service Manager. La definición de proceso consta de expresiones iniciales, RAD y expresiones finales, introducidas en las fichas Expresiones iniciales, Javascript inicial, RAD, Expresiones finales, Javascript final y Proceso siguiente, así como una llamada opcional a un siguiente proceso. Las expresiones se escriben mediante expresiones RAD estándar o JavaScript.

Creación y modificación de un proceso

Para crear un proceso:

1. Acceda al Motor de documentos. Consulte [Acceso al Motor de documentos en la página 10](#).
2. Mediante las fichas del panel Procesos, rellene los campos necesarios para crear un proceso que realizará las funciones que desee. Para obtener más información, consulte [Descripciones de campos del formulario Definición del proceso en la página 29](#).

Para modificar un proceso:

1. Acceda al Motor de documentos. Consulte [Acceso al Motor de documentos en la página 10](#).
2. Introduzca el nombre del proceso que desea modificar en el campo Nombre del proceso o pulse **Buscar** para buscar el proceso.

Descripciones de campos del formulario Definición del proceso

El formulario Definición del proceso permite definir nuevos procesos o editar procesos existentes. Los procesos ejecutan código o expresiones para realizar las acciones que el usuario ha seleccionado.

Las descripciones de campos del formulario Definición del proceso son:

Nombre del campo	Descripción
Nombre del proceso <i>process</i>	Especifica el nombre del proceso. (Obligatorio)
¿Guardar posición del cursor? <i>save.cursor.position</i>	Si se selecciona esta casilla indica que desea regresar a la misma posición de cursor tras la acción (por ejemplo, al rellenar).

Nombre del campo	Descripción
¿Ejecutar el proceso estándar al completar? <i>run.standard</i>	Si se selecciona, el sistema ejecuta un proceso estándar tras completar la acción o proceso actual. Por ejemplo, un proceso estándar como guardar. Si ha creado un proceso para guardar y desea ejecutar el proceso para guardar que viene con el Motor de documentos tras completar el proceso para guardar definido, marque esta casilla. En las secciones de funciones básicas de este documento encontrará la definición de un proceso estándar.
¿Ejecutar en ventana? <i>run.in.window</i>	Si se selecciona, el proceso se ejecuta en una ventana diferente.
Título de ventana <i>window.name</i>	Si la casilla ¿Ejecutar en ventana? está seleccionada, especifique un título para la ventana. Puede utilizarse una expresión scmsg para los títulos de ventanas localizados, como scmsg(1980, "us").

Además de los campos descritos arriba, también hay fichas en este formulario que le permiten definir mejor el proceso. Estas fichas son:

- [Ficha Expresiones iniciales en la página 30](#)
- [Ficha Javascript inicial en la página 30](#)
- [Ficha RAD en la página 31](#)
- [Ficha Expresiones finales en la página 31](#)
- [Ficha Javascript final en la página 31](#)
- [Ficha Proceso siguiente en la página 32](#)

Ficha Expresiones iniciales

La ficha Expresiones iniciales define las expresiones iniciales que se ejecutan antes del Javascript inicial y antes de las llamadas RAD definidos en la ficha RAD. Las expresiones iniciales se escriben mediante expresiones estándar de Service Manager.

Ficha Javascript inicial

La ficha Javascript inicial define las expresiones JavaScript iniciales que se ejecutan antes de las expresiones RAD definidas en la ficha RAD.

Ficha RAD

La ficha RAD define las expresiones previas a RAD, las llamadas RAD y las expresiones posteriores a RAD que se ejecutan como parte del proceso.

Las descripciones de campos de la ficha RAD son:

Nota: estos campos se repiten para cada aplicación RAD definida en esta ficha.

Nombre del campo	Descripción
Expresiones evaluadas antes de una llamada RAD <i>pre.rad.expressions</i>	Especifica la expresión que se ejecuta antes de la aplicación RAD definida en el campo de aplicación RAD. Todos los valores de parámetros pasados a la aplicación RAD deben ser en forma de variables o expresiones. Las variables deben asignarse con sus valores en las expresiones previas a RAD; por ejemplo, <code>\$L.value.name="Test"</code> .
Aplicación RAD <i>application</i>	Especifica el nombre de la aplicación RAD que debe ejecutarse.
Condición <i>rad.condition</i>	Especifica las circunstancias bajo las cuales la aplicación RAD debe ejecutarse (la condición se evalúa en true) u omitirse (la condición se evalúa en false).
Nombres de parámetros <i>names</i>	Especifica los nombres de parámetros pasados a la aplicación RAD.
Valores de parámetros <i>values</i>	Especifica los valores de parámetros pasados a la aplicación RAD. Estos valores deben ser en forma de variables o expresiones. Los valores de cadenas pueden pasarse cuando van entre comas ("Nombre del Asistente").
Expresiones posteriores a RAD <i>post.rad.expressions</i>	Especifica cualquier expresión RAD que se ejecutará tras finalizar la aplicación RAD.

Ficha Expresiones finales

La ficha Expresiones finales define las expresiones finales que se ejecutan tras finalizar el procesamiento de la ficha RAD. Las expresiones finales se escriben mediante expresiones estándar de Service Manager.

Ficha Javascript final

El Javascript definido en esta ficha se ejecuta tras las expresiones finales y tras las aplicaciones RAD de la ficha RAD.

Ficha Proceso siguiente

Esta ficha especifica el siguiente proceso o procesos para ejecutar una vez finalizado el proceso actual.

Las descripciones de campos de la ficha Proceso siguiente son:

Nombre del campo	Descripción
Proceso siguiente <i>next.process</i>	Especifica el nombre del siguiente proceso que debe ejecutarse.
Condición <i>process.condition</i>	Especifica una condición asociada al proceso en el campo Proceso siguiente que se evalúa en true o false. Por ejemplo, true para el proceso sm.close.

Capítulo 5

Recursos del Motor de documentos

El Motor de documentos incluye variables locales y funciones básicas que cualquier objeto puede utilizar. Las funciones básicas se definen de serie para ejecutar acciones estándar disponibles para el usuario. Por ejemplo, cuando un usuario pulsa el botón Guardar tras actualizar un registro, el Motor de documentos procesa esta petición y escribe los cambios en la base de datos. Una función básica puede sobrescribirse en el registro de estado para ejecutar un proceso diferente.

Objeto predeterminado (DEFAULT)

Cualquier archivo al que se ha accedido a través de Gestor de base de datos utiliza automáticamente las normas y procesos definidos para el objeto correspondiente. Si no se ha definido ningún objeto para el archivo, utiliza el objeto predeterminado (DEFAULT). Este objeto duplica la funcionalidad de las versiones anteriores del Gestor de base de datos.

Un Administrador del sistema también puede acceder a un archivo que tenga un objeto correspondiente a través del objeto predeterminado (DEFAULT) accediendo al archivo a través del Gestor de base de datos con la casilla de verificación "Modo de administración" seleccionada. Esta casilla de verificación sólo está disponible para Administradores del sistema. Un usuario con la palabra habilitante "AlwaysAdmin" siempre utilizará el objeto predeterminado (DEFAULT) al acceder a información en Service Manager. No se recomienda asignar la palabra habilitante a un usuario.

El Motor de documentos utiliza perfiles de entorno. En versiones anteriores del sistema, un Administrador del sistema obtenía todos los derechos (añadir, actualizar, eliminar), independientemente de los ajustes de control de formatos. Con el Motor de documentos, los derechos otorgados coinciden con aquellos definidos en el registro de control de formatos. Los administradores que prefieran el método antiguo para otorgar derechos pueden modificar simplemente el ajuste "Perfil de aplicación" en el registro de objeto predeterminado (DEFAULT) de "db.environment" a "db.environment.sysadmin".

Aplicaciones RAD

Con una aplicación basada en una base de datos, un usuario puede trabajar con cualquiera de los siguientes tres conjuntos de registros básicos:

- Sin registros: al buscar información.
- Muchos registros: al consultar una lista de información.
- Un registro: al realizar cambios en un único registro.

Desde una perspectiva de usuario al trabajar con datos:

- o bien ve un formulario en blanco, donde puede introducir información de búsqueda (o crear un nuevo registro),
- o está trabajando en la actualización/visualización de un único registro,

- o realizó una búsqueda y ahora tiene una serie de múltiples registros que se muestran como lista.

El Motor de documentos utiliza tres aplicaciones RAD principales para reflejar esta idea. La rutina RAD utilizada está determinada por el número de registros mostrados en cualquier momento.

se.search.engine

El motor de búsqueda se utiliza cuando no se muestra ningún registro. El principal objetivo de esta rutina es formular una consulta y seleccionar registros de la tabla correcta. Esta rutina también puede utilizarse para introducir información inicial en un registro en blanco con el fin de añadir un nuevo registro a la base de datos. El estado predeterminado es db.search.

se.list.engine

El motor de lista muestra varios registros. A través del motor de lista un usuario puede seleccionar un registro específico de la lista o realizar acciones en toda la lista de registros. El estado predeterminado es db.list.

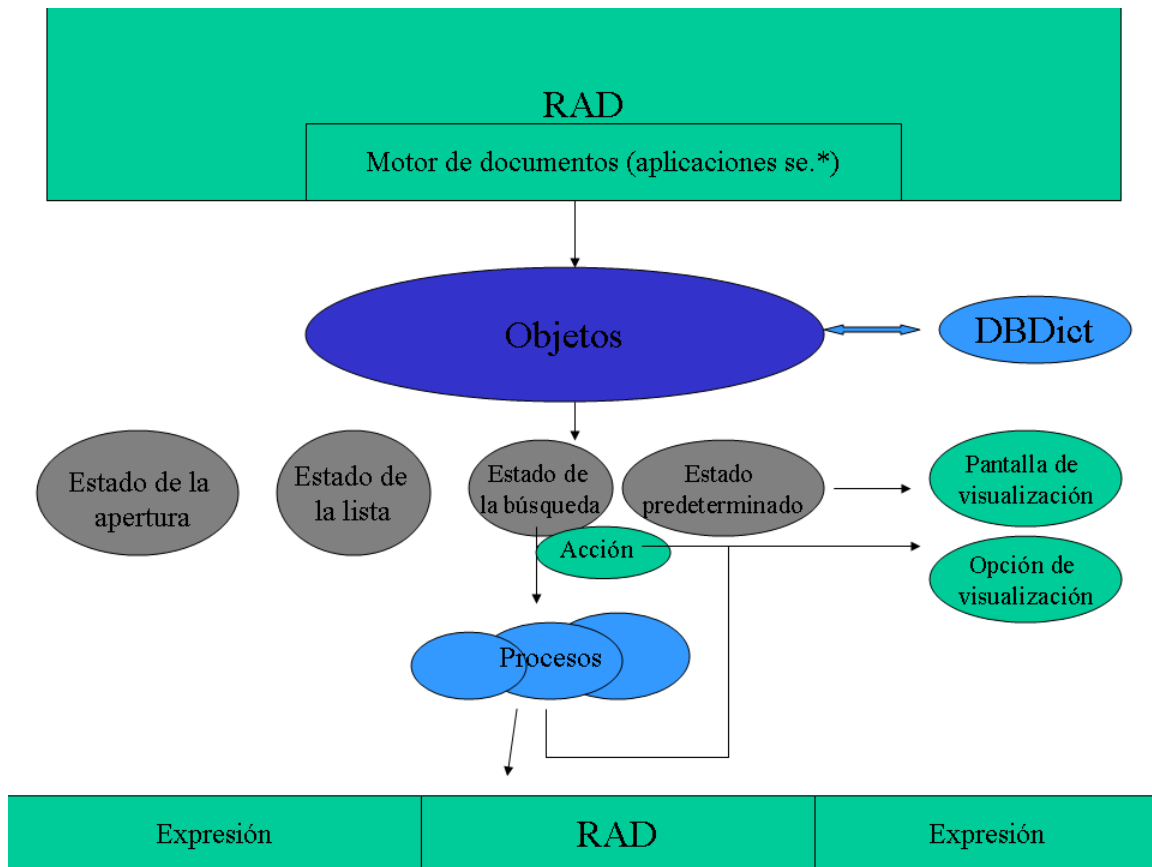
se.view.engine

El motor de visualización muestra un único registro. Esta aplicación permite realizar acciones sobre un registro específico, como actualizaciones o eliminaciones. El estado predeterminado es db.view.

Nota: al utilizar la funcionalidad de la lista de registros de Service Manager, el motor de visualización se utiliza para la información del único registro y la lista.

Flujo de aplicaciones RAD

En la siguiente figura se ofrece una visión general de cómo interactúan el resto de herramientas de personalización con el Motor de documentos.



Nota: todas las aplicaciones RAD relacionadas con el Motor de documentos empiezan por "se", como se.view.engine. El Motor de documentos trata cada registro como un documento disponible para el usuario.

Cuando un archivo se muestra mediante el Motor de documentos siempre utilizará una de las tres aplicaciones especificadas previamente en este documento. La pantalla de visualización, las opciones de visualización y el formato están determinados por el estado actual del registro. Si se activa una opción de visualización, tras efectuarse la funcionalidad de visualización estándar, el motor comprueba la "acción" de esa opción con respecto a los procesos disponibles definidos en el registro de estado actual. Si existe un proceso definido con una condición que se evalúa en true, el motor realiza ese proceso en el registro actual (o conjunto de registros). Si la acción no está definida en el registro de estado, el motor comprobará si esa acción se ha definido como un "Proceso básico" en la aplicación actual. Si es así, el sistema llevará a cabo ese proceso básico en el registro. De lo contrario, no se llevará a cabo ninguna acción.

Sugerencias de integración para la aplicación de visualización

Si una pantalla de visualización no tiene ningún evento de visualización asociado a ella, el sistema utiliza automáticamente el evento de visualización se.default que realiza se.lock.object en el caso de "OnFormModified".

Si un registro de opción de visualización tiene la casilla de verificación "Modifica el registro" activada, el registro se bloquea al pulsar el botón.

Funciones básicas en se.search.engine

Acción de visualización	Descripción
back	Sale del registro.
find	Muestra información detallada del registro enlazado al campo.
fill	Rellena información de un registro enlazado a un registro de destino. Coloca los datos de un campo de registro de origen a un campo de destino.
advanced	Inicia el proceso Búsqueda avanzada que proporciona al usuario más opciones de búsqueda. Generalmente, esta opción sólo está disponible para administradores del sistema, pero puede ofrecerse a otros usuarios.
clear	Despeja la pantalla actual.
openinbox / inbox	Indica al usuario que debe abrir una vista asociada al archivo actual. Se trata de una consulta predefinida que muestra los resultados como una vista.
search	Realiza una búsqueda estándar creando una consulta de búsqueda con la información proporcionada y, a continuación, muestra los resultados en una lista de registros.
add	Realiza una adición en Control de formatos e intenta añadir un registro a la base de datos.
restore	Restaura el contenido de la pantalla (tras "despejarla").
irquery	Ejecuta una búsqueda de texto de Consulta IR.
validitylookup	Efectúa una verificación de validez de Service Manager. La verificación de validez verifica a través de una entrada de la tabla de validaciones que un valor específico es válido para el campo definido. Con la verificación de validez, el sistema verifica que un valor introducido por el usuario cumple con las reglas de validación.
expandarray	Expandir matriz proporciona una funcionalidad adicional para editar matrices, como insertar una línea y eliminar una línea.
reset	Restablece el archivo actual. Esta acción elimina TODOS los registros de la base de datos para la tabla.
regen	Regenera los índices del archivo actual. Esta acción sólo se aplica a las tablas con una consulta IR y sólo se llevará a cabo una regeneración de IR.
export/unload	Inicia el proceso de exportar/descargar "Sin registros". Se utiliza sólo para descargar definiciones vacías de la DBDICT sin datos.

Acción de visualización	Descripción
views	Si se selecciona, el sistema presenta al usuario una lista de formularios alternativos donde mostrar el registro actual.
findrevision	Muestra revisiones de registros en este objeto.
initrevision	Crea una revisión de un registro en este objeto.
newsite	
newview	
newTable	
addFilter	El cliente activa esta opción al realizar una búsqueda de filtro avanzada en la pantalla de búsquedas estándar.
editFilter	El cliente activa esta opción al realizar una búsqueda de filtro avanzada en la pantalla de búsquedas estándar.
addCompound	El cliente activa esta opción al realizar una búsqueda de filtro avanzada en la pantalla de búsquedas estándar.
removeSelection	El cliente activa esta opción al realizar una búsqueda de filtro avanzada en la pantalla de búsquedas estándar.

Funciones básicas en se.view.engine

Acción de visualización	Descripción
save	Realiza actualizaciones en Control de formatos y actualiza el registro en la base de datos si la actualización es válida.
add	Realiza una adición en Control de formatos y añade un nuevo registro a la base de datos si el nuevo registro es válido.
ok	Si el registro ha cambiado, guarde; de lo contrario salga.
reselect	Vuelve a seleccionar el registro actual de la base de datos (en caso de haber cambiado).
fill	Rellena información de un registro enlazado a un registro de destino. Coloca la información de un campo de origen a un campo de destino.
find	Muestra información detallada del registro enlazado al campo.
next	Mueve al siguiente registro de la lista (tras verificar los cambios).

Acción de visualización	Descripción
previous	Mueve al registro anterior de la lista (tras verificar los cambios).
back	Regresa al formulario de búsqueda o la lista.
menu	Regresa al menú que invoca.
delete	Realiza una eliminación en Control de formatos e intenta eliminar un registro de la base de datos tras validar la petición.
views	Si se selecciona, el sistema presenta al usuario una lista de formularios alternativos donde mostrar el registro actual.
print	Imprime el registro actual.
printlist*	Imprime la lista actual de registros.
validitylookup	Efectúa una verificación de validez de Service Manager. La verificación de validez verifica a través de una entrada de la tabla de validaciones que un valor específico es válido para el campo definido. Con la verificación de validez, el sistema verifica que un valor introducido por el usuario cumple con las reglas de validación.
export/unload	Funcionalidad Exportar/Descargar estándar de Service Manager (varios registros).
massunload*	Proporciona la funcionalidad de descarga para una lista de registros.
massadd*	Añade una nueva serie de registros que duplica exactamente una serie de registros existentes excepto que la nueva serie de registros tendrá un valor de clave único actualizado con nuevos datos. Al final del procesamiento habrá el doble de registros en la tabla.
massupdate*	Proporciona las mismas actualizaciones a una serie de campos específicos en una lista de registros.
massdelete*	Elimina una serie de registros específicos de una tabla.
irquery	Ejecuta una búsqueda de texto de Consulta IR.
expandarray	Expandir matriz proporciona una funcionalidad adicional para editar matrices, como insertar una línea y eliminar una línea.
count*	Cuenta los registros de la lista de registros y muestra el recuento total.
audithistory	Invoca la rutina del histórico de auditorías estándar. El histórico de auditorías está determinado por los campos para realizar auditorías como se define en la tabla auditdef.

Acción de visualización	Descripción
inbox.save / inbox*	Guarda la consulta actual como una vista.
approval.log	Visualiza el histórico de aprobaciones del registro actual.
alert.log	Visualiza el histórico de alertas del registro actual.
alerts	Visualiza las alertas actuales y planificadas del registro actual.
pagelist / listpages	La lista de páginas muestra el histórico completo de un registro. Si se habilita el registro de páginas, cada vez que efectúa una actualización, una copia del registro se escribe en el archivo de páginas. La lista de páginas muestra el histórico completo de un registro según el archivo de páginas.
clocks	Muestra registros de relojes asociados a este registro.
xmlfill	Gestiona campos XML, como las opciones de usuario de Catálogo de servicios.
createTemplate	Crea un registro de plantilla a partir del registro actual.
applyTemplate	Aplica una plantilla existente al registro actual.

Nota: * estos comandos sólo se aplican si la funcionalidad de la lista de registros se está utilizando.

Funciones básicas en se.list.engine

Acción de visualización	Descripción
exit (or back)	Regresa al formulario de búsqueda (o invocación).
inbox.save / inbox	Indica al usuario que debe guardar la consulta actual como una vista nueva.
count	Realiza una función de recuento estándar y muestra el recuento total de registros.
refresh	Actualiza la lista mediante la consulta actual.
big.green	Sale totalmente del módulo actual. "Gran flecha verde"
print	Imprime la lista de registros mostrados.
views	Si se selecciona, el sistema presenta al usuario una lista de formularios alternativos donde mostrar el registro actual.
export/unload	Funcionalidad Exportar/Descargar estándar de Service Manager (varios registros).
massadd	Añade una nueva serie de registros que duplica exactamente una serie de registros existentes excepto que la nueva serie de registros tendrá un valor de clave único actualizado con nuevos datos. Al final del procesamiento habrá el doble de registros en la tabla.
massupdate	Proporciona las mismas actualizaciones a una serie de campos específicos en una lista de registros.
massdelete	Elimina una serie de registros específicos de una tabla.

Variables locales

Las variables locales empiezan por \$L. y sólo persisten en la aplicación RAD que se está ejecutando. El servidor elimina automáticamente todas las variables locales cuando cierra una aplicación RAD.

A continuación, se muestra una lista de variables estándar utilizadas con el Motor de documentos:

\$L.action: el valor de la acción de visualización de la opción de visualización.

\$L.bg: etiqueta de segundo plano.

\$L.category: el registro de categoría (si está disponible).

\$L.env: el registro de entorno actual.

\$L.exit: el parámetro de salida interna.

\$L.file: la variable de archivo actual.

\$L.file.save: una copia del registro en su estado original.

`$L.format`: nombre del formato utilizado para mostrar el registro.

`$lrsread`: determina las opciones de descubrimiento IR: 0=búsqueda superficial, 2=búsqueda profunda, 4=coincidencia completa.

`$L.mode`: el modo en que está el registro visualizado. Generalmente se añade para crear un nuevo registro, se actualiza para modificar un registro existente o se cierra para finalizar el procesamiento de un registro existente.

`$L.mult`: etiqueta que se establece como true si hay varios registros en la variable `$L.file`.

`$L.object`: el registro de objeto.

`$L.phase`: el registro de fase (si está disponible).

`$L.sql` o `$L.query`: la consulta actual.

`$L.sort`: el orden de clasificación actual.

`$L.state`: el registro de estado que utiliza el sistema (=el estado en que se encuentra el registro).

Variables que están disponibles en el modo de vista (al ver un único registro)

`$L.fc`: copia del registro de formato de control detallado.

`$L.fc.master`: copia del registro de formato de control maestro.

Capítulo 6

Visión general de la solución de problemas

Para solucionar problemas del Motor de documentos correctamente, deberá recopilar la siguiente información:

- El dbdict y objeto que se están utilizando.
- El estado en que se encuentra el registro.
- El nombre del proceso.
- Los pasos seguidos para reproducir el problema.

Investigación de la ruta de la aplicación a través del Motor de documentos

En la solución de problemas del Motor de documentos, al igual que en la solución de problemas de cualquier aplicación de Service Manager, introduzca **RTM:3** y **debugdbquery:999** en el archivo `sm.ini` de Service Manager e inicie una nueva conexión de cliente. A menos que este proceso de usuario sea el primero en invocar los procesos del Motor de documentos para depurar, no mostrará la selección de los registros de estado o proceso en el archivo `sm.log` al realizar este rastreo, pero proporcionará sugerencias útiles como qué proceso se estaba invocando.

Búsqueda del dbdict u objeto utilizados

Para determinar qué dbdicts u objetos se está utilizando, busque el archivo de registro.

Registro de muestra:

```
1320 07/18/2006 11:00:36 RADTRACE 20 [ 1] se.get.object get.object select CPU( 0 1411 )
1320 07/18/2006 11:00:36 (0x0129AC08) DBACCESS - Cache Find against file Object found 1
record, query: file.name="pcsoftware" (DBACCESS - Cache Find en el archivo Object ha
encontrado 1 registro, consulta: file.name="pcsoftware")
1320 07/18/2006 11:00:36 RADTRACE 20 [ 1] se.get.object set.access process CPU( 0 1411 )
```

Búsqueda del estado de los registros

El siguiente tema a tener en cuenta es el estado en que se encuentra el registro. Para encontrar esta información, busque lo siguiente en el archivo `sm.log`:

Registro de muestra:

```
1320 07/18/2006 11:00:48 RADTRACE 10 [ 1] se.get.state select.state select CPU( 0 1491 )
1320 07/18/2006 11:00:48 (0x01292FB0) DBACCESS - Cache Find against file States found 1
record, query: state="pcs.list" (DBACCESS - Cache Find en el archivo States ha
encontrado 1 registro, consulta: file.name="pcsoftware")
1320 07/18/2006 11:00:48 RADTRACE 10 [ 1] se.get.state exit.normal process CPU( 0 1491 )
```

Búsqueda del nombre del proceso

El nombre del proceso también puede encontrarse buscando en el archivo `sm.log` que contiene el rastreo que se muestra en el siguiente ejemplo.

Registro de muestra:

```
1320 07/18/2006 11:00:50 RADTRACE 20 [ 1] se.call.process select.process select CPU( 0
1542 )

1320 07/18/2006 11:00:50 (0x00B56810) DBACCESS - Cache Find against file Process found 1
record, query: process="upgrade.pcs" (DBACCESS - Cache Find en el archivo Process ha
encontrado 1 registro, consulta: process="upgrade.pcs")

1320 07/18/2006 11:00:50 RADTRACE 20 [ 1] se.call.process run.pre.exp process CPU( 0 1542
)
```

Investigación de los errores de la aplicación

Los procesos llaman a un número de aplicaciones RAD y ejecutan un número de expresiones, con una posibilidad de invocar más procesos posteriormente. Si alguna aplicación o expresión ha causado una salida por error debido a una sintaxis o lógica incorrectas, esta información puede encontrarse en el archivo `sm.log`.

Registro de muestra:

```
El panel de proceso run.pre.exp en RAD se.call.process ha encontrado un error en la línea 1
(se.call.process,run.pre.exp)

No se ha podido evaluar la expresión (se.call.process,run.pre.exp)
No se ha podido evaluar la expresión (se.call.process,run.pre.exp)
Arg. erróneo (2) oper. = (se.call.process,run.pre.exp)
No se ha podido evaluar la expresión (se.call.process,run.pre.exp)
No se ha podido evaluar la expresión (se.call.process,run.pre.exp)
Arg. erróneo (2) oper. = (se.call.process,run.pre.exp)
No se ha podido evaluar la expresión (se.call.process,run.pre.exp)
Arg. erróneo (2) oper. nullsub (se.call.process,run.pre.exp)
No se ha podido evaluar la expresión (se.call.process,run.pre.exp)
Arg. erróneo (2) oper. in (se.call.process,run.pre.exp)
No se ha podido evaluar la expresión (se.call.process,run.pre.exp)
Error irre recuperable en la aplicación: se.search.objects en panel call.list.engine
Error irre recuperable en la aplicación: se.list.engine en panel call.process.1
Error irre recuperable en la aplicación: se.call.process en panel run.pre.exp
```

En este ejemplo, (no de serie), el error ocurrió en `se.call.process, run.pre.exp`, o en otras palabras, mientras se evaluaba las expresiones iniciales del proceso. Para conocer qué proceso causó el error, vaya a los pasos detallados arriba y anote el proceso de la línea:

```
DBACCESS - Cache Find against file Process found 1 record, query: process="upgrade.pcs"
(DBACCESS - Cache Find en el archivo Process ha encontrado 1 registro, consulta:
process="upgrade.pcs")
```

Vaya al registro de proceso mediante el nombre de **upgrade.pcs** y compruebe las instrucciones en la ficha Expresiones iniciales. En este caso concreto, la expresión incluirá la palabra `nullsub`. Por ejemplo, la expresión en cuestión para este test puede ser

```
$L.icount=nullsub($L.icount, anynumberIwant)
```

La variable, `anynumberIwant`, no es un campo válido, literal o variable, por lo que deberá cambiarse para evitar este problema.

Impresión de valores de variables o resultados de expresiones

En el Motor de documentos la ruta a través de un flujo de trabajo está determinada a menudo por el valor asignado a un campo o variable. Para determinar el valor asignado a un campo o variable que influye en el flujo de trabajo, utilice la función JavaScript `print()` o utilice la función `rtecall("log")` (`$L.void=rtecall("log", $L.rc, "message")`) en las expresiones RAD. El mensaje puede ser una cadena concatenada como

```
$L.message="The value of $L.test is " + $L.test
```

donde `$L.test` es una variable a la que se asignó un valor de carácter.

Capítulo 7

Visión general de un pedido de trabajo de muestra

¿Qué es un pedido de trabajo? Un pedido de trabajo es una tarea específica asignada a un ingeniero que identifica una actividad necesaria para resolver un incidente o posiblemente cualquier otro módulo si el sistema de pedido de trabajo se ampliara para incluir cambios, problemas o errores conocidos. El siguiente ejemplo se basa en Service Manager versión 7.11 y se ha escrito para que los usuarios puedan crear, actualizar y cerrar pedidos de trabajo para Gestión de incidentes. Puede modificarse fácilmente para otra aplicación de Service Manager.

Este ejemplo crea un sistema de pedido de trabajo para demostrar cómo debe utilizarse el Motor de documentos. Este sistema permitirá a los usuarios crear pedidos de trabajo para un incidente y ver el estado de estos pedidos de trabajo. Asimismo, el sistema permite a los usuarios ver y actualizar un pedido de trabajo de Gestión de incidentes. Todos los pedidos de trabajo se asocian a un incidente particular en el sistema, y el incidente no puede cerrarse hasta que todos los pedidos de trabajo del incidente estén cerrados.

Este ejemplo está dirigido a aquellos usuarios que personalizan el sistema. Debe conocer a fondo las siguientes funciones de personalización:

- Diccionario de base de datos para crear una nueva tabla.
- Diseñador de formularios para modificar formularios de serie y crear nuevos formularios.
- Herramienta de creación de asistentes.

El pedido de trabajo de muestra le guía a través de los siguientes pasos:

- Creación de un nuevo diccionario de base de datos (dbdict) mediante la utilidad dbdict
- Especificación de los campos clave de la tabla
- Creación de un formulario EXWorkOrder para la tabla EXWorkOrder mediante el Diseñador de formularios
- Creación de un archivo de números secuenciales
- Modificación del formulario para incluir listas desplegables
- Creación de un enlace para el formulario EXWorkOrder
- Creación de un formulario que el asistente utilizará para recopilar información para el pedido de trabajo
- Modificación de los formularios de incidentes de cierre y actualización
- Creación de alias para enlazar pedidos de trabajo a incidentes
- Creación de definiciones de estado
- Creación de definiciones de la pantalla de visualización de la aplicación para abrir, cerrar y ver
- Creación de definiciones de la opción de visualización de la aplicación de Añadir, Cancelar, Rellenar y Encontrar para abrir, cerrar y visualizar

- Modificación del registro de definición de proceso im.set.close
- Comprobación del pedido de trabajo de muestra

Creación de una tabla

Utilice la utilidad dbdict para crear una nueva tabla. En este ejemplo creará una tabla llamada EXWorkOrder. Antes de crear esta tabla, debe saber los campos que se necesitan y los atributos de dichos campos. En este ejemplo, la tabla almacena los datos de pedidos de trabajo asociados a incidentes.

Para crear una tabla:

1. En el Navegador del sistema, pulse **Personalización > Diccionario de base de datos**. Se abre el formulario Diccionario de la base de datos.
2. Escriba EXWorkOrder en Nombre del archivo.
3. Pulse **Nuevo**.
4. En la ficha Campos, introduzca la siguiente información tras pulsar **Campo/clave nuevos**.

Campo	Descripción
ID	Se rellenará a partir del archivo numérico. Tipo: carácter
RelatedID	Se rellenará a través del ID único del registro relacionado; por ejemplo: número de incidente) Tipo: carácter
status	Permite almacenar el estado del pedido de trabajo. Tipo: carácter
initiator	Operador que abrió el pedido de trabajo. Tipo: carácter
assignee.name	Operador asignado al pedido de trabajo. Tipo: carácter
description	Permite añadir la descripción del trabajo que debe realizarse para el pedido de trabajo. Tipo:matriz de caracteres
category	Se rellenará a partir de la categoría del registro relacionado. Tipo: carácter
RelatedCIs	Se utiliza para la lista de elementos de configuración (CI) asociados al pedido de trabajo. Tipo: matriz de caracteres

impact	Tipo: carácter
urgency	Tipo: carácter
priority	Tipo: carácter
closure.code	Tipo: carácter
deadline	Fecha en que debe completarse y rellenarse el pedido de trabajo al abrirlo. Tipo: fecha/hora
est.finish	La finalización estimada del asignado. Tipo: fecha/hora
update.action	. Tipo: matriz de caracteres
closure.comments	. Tipo: matriz de caracteres

- Tras añadir los campos a la tabla debe añadir las claves. No salga de la utilidad Diccionario de base de datos hasta que lo haya hecho para la tabla **EXWorkOrder**. Para obtener información detallada, consulte [Adición de campos de claves a una tabla](#).

Adición de campos de claves a una tabla

Utilice la utilidad dbdict para añadir claves a una nueva tabla. Tras crear una tabla llamada EXWorkOrder, debe añadirle claves. Las claves permiten habilitar las búsquedas indexadas, así como garantizar la coherencia de datos.

Para añadir claves a una tabla:

- En el Diccionario de la base de datos pulse la ficha **Claves**.
- Seleccione la primera entrada disponible para un nuevo campo de clave.
- Pulse **Campo/clave nuevos**.
- Introduzca la siguiente información para cada clave que cree o edite.

Campo	Descripción
ID	Tipo: Único
RelatedID	Tipo: No hay nulos
RelatedCIs	Tipo: Nulos y duplicados
assignee.name	Tipo: Nulos y duplicados

- Pulse **Aceptar**.

Creación de un formulario

Vaya a Diseñador de formularios para crear un formulario para la tabla EXWorkOrder. Mediante el asistente para crear el formulario seleccione Detalle de un único registro y continúe. Esto crea un formulario básico con todos los campos que se pueden modificar, si fuera preciso. A continuación, se incluye un ejemplo del formulario EXWorkOrder.

Para crear un formulario:

1. En el Navegador del sistema, pulse **Personalización > Diseñador de formularios**.
Se abre el formulario Diseñador de formularios.
2. Escriba `EXWorkOrder` en Formulario.
3. Pulse **Nuevo**.
4. Pulse **Sí** para usar el Asistente para formularios.
5. Escriba `EXWorkOrder` para el nombre de la tabla para la que creará un formulario.
6. Seleccione **Detalle de un único registro** para el tipo de formulario que desee crear.
7. Pulse **Aceptar**.
8. Pulse **Continuar** para aceptar los valores predeterminados de los campos que desea incluir en el formulario.
9. Utilice la herramienta Diseñador de formularios para modificar el diseño del formulario. A continuación, se incluye un ejemplo del formulario **EXWorkOrder**.
10. Los valores de Estado son:
 - New (Nuevo) (estado predeterminado)
 - Open (Abierto)
 - Ready (Listo)
 - Closed (Cerrado)
11. Los valores de Código de cierre son:
 - Implemented (Implementado)
 - Canceled (Cancelado)
 - Rollback (Revertido)

A continuación, se incluye un formulario EXWorkOrder de muestra.

The screenshot shows a web-based form titled "WorkOrder". At the top, there is a menu bar with buttons for "Aceptar", "Cancelar", "Eliminar", and "Diseñar". Below the menu, the form is organized into several sections. The top section contains fields for "ID:", "ID relacionado:", and "Categoría:", each with a text input box. To the right of these is a dropdown menu for "Estado:". Below this is a tabbed interface with three tabs: "General", "Información de activos", and "Actualización / Cierre". The "General" tab is active and contains several fields: "Iniciador:" and "Nombre del asignado:" with dropdown menus and search icons; "Crítica:" and "Otra estim.:" with dropdown menus; "Impacto:", "Urgencia:", and "Prioridad:" with text input boxes; and a large "Descripción:" text area. The HP logo is visible in the top right corner of the browser window.

Creación de una copia de un formulario

En el Diseñador de formularios cree el formulario `sc.manage.WorkOrder` haciendo una copia de `sc.manage.problem`. Modifique las entradas de las columnas de la tabla para utilizar los campos de la tabla `EXWorkOrder`.

Para crear un nuevo formulario a partir de una copia de un formulario:

1. En el Navegador del sistema, pulse **Personalización > Diseñador de formularios**.
Se abre el formulario Diseñador de formularios.
2. Escriba `sc.manage.problem` en Formulario.
3. Pulse **Buscar**.
4. Seleccione `sc.manage.problem.g`.
5. Pulse **Copiar/Cambiar de nombre** en el menú Más acciones.
6. Escriba `sc.manage.WorkOrder` en Nuevo nombre.
7. Pulse **Aceptar**.
8. En Diseñador de formularios, actualice los campos de entrada de las columnas mediante la tabla `EXWorkOrder`.
9. ■ ID del incidente: ID

- Categoría: category
- ID relacionado: RelatedID
- Estado: status
- Asignatario: assignee.name
- Descripción: description,1
- Prioridad: priority
- Impacto: impact
- Urgencia: urgency

10. Pulse **Aceptar**.

Creación de un enlace para el formulario de pedido de trabajo

Una de las ventajas de una base de datos relacional es la eliminación de información redundante. Esto se consigue mediante el almacenamiento de información sobre un asunto concreto en un lugar o tabla que tenga enlaces hacia otros asuntos. Los enlaces son combinaciones de definiciones de datos y enlaces con conjuntos de condiciones que contienen las relaciones de la información enlazada.

Para crear un enlace para el formulario de pedido de trabajo:

1. En el Navegador del sistema, pulse **Personalización > Herramientas de personalización > Enlaces**.
Se abre el formulario Archivo de enlace.
2. Escriba `EXWorkOrder` en Nombre.
3. Añada una descripción en Descripción.
4. Pulse **Nuevo**.
5. Introduzca la siguiente información:

Nombre del campo de origen	Nombre del archivo de destino	Nombre del campo de destino
initiator	operator	name
assignee.name	operator	name
RelatedCIs	device	logical.name

6. Seleccione la fila initiator (resáltela) y elija **Seleccionar línea** del menú Más acciones.
7. En el formulario link.structure.g introduzca la siguiente información para initiator. A continuación, repita el paso 6 para **assignee.name** y **RelatedCIs** y utilice la siguiente información para aquellos campos, respectivamente.

Campo de origen (Rellenar en/Anotar desde)	Campo de destino (Rellenar en/Anotar desde)
initiator	name
assignee.name	name
RelatedCIs	logical.name

8. Pulse **Guardar**.
9. Pulse **Aceptar**.

Creación de un archivo de números secuenciales

Cree un archivo de números secuenciales para generar números secuenciales para los registros de la tabla EXWorkOrder.

Para crear un archivo de números secuenciales:

1. En el Navegador del sistema, pulse **Personalización > Herramientas de personalización > Números secuenciales**.
Se abre el formulario Archivo de números secuenciales.
2. Escriba `EXWorkOrder` en Clase.
3. Escriba `1` en Último número.
4. Escriba `Número del pedido de trabajo` en Descripción.
5. Escriba `5` en Longitud.
6. Escriba `wo` en Prefijo.
7. Pulse **Añadir**.

Creación de una definición de objeto

El objetivo de este objeto es definir las características y el comportamiento del objeto de EXWorkOrder que determina los datos que deben incluirse en un registro de pedido de trabajo y cómo procesará el sistema los pedidos de trabajo.

Nota: Consulte [Descripciones de campos del formulario Definición del objeto en la página 12](#) para obtener más información.

Para crear una definición de objeto:

1. En el Navegador del sistema, pulse **Personalización > Motor de documentos > Objetos**.
Se abre el formulario Definición del objeto.
2. Escriba `EXWorkOrder` en Nombre del archivo.
3. Pulse **Añadir** para crear el registro de objeto.
4. En la ficha Info. del objeto, introduzca la siguiente información:

Campo	Valor
Clave única	Se rellena automáticamente (ID)
Nombre común	Se rellena automáticamente (EXWorkOrder)
Campo de descripción	Tabla que incluye los pedidos de trabajo que pueden relacionarse con cualquiera de los módulos
Aplicación del perfil	db.environment
Variable del perfil	\$L.env
Nombre del registro de número	EXWorkOrder
Nombre de la tabla de categorías	category
Control de formatos maestro	EXWorkOrder
Campo de estado	status
Asignado a campos	assignee.name
Estado de la apertura	EXWorkOrder.open
Estado del cierre	EXWorkOrder.close
Estado predeterminado	EXWorkOrder.view
Estado de la búsqueda	EXWorkOrder.search

5. Pulse **Guardar**.
6. Seleccione la ficha **Gestión de colas**.
7. En la ficha Gestión de colas, introduzca la siguiente información:

Campo	Valor
Gestionar condición	true
Gestionar formulario de visualización	sc.manage.WorkOrder
Gestionar consulta predeterminada	assignee.name=operator()
Permitir condición de adición	false
Descripción de consulta predeterminada	My WorkOrders

8. Pulse **Guardar**.

9. Pulse **Aceptar**.

Creación de una definición de proceso de inicialización

Esta definición de proceso del pedido de trabajo de muestra especifica las expresiones iniciales y las aplicaciones RAD que deben utilizarse cuando un usuario abre un registro de pedido de trabajo.

Para crear una definición de proceso de inicialización:

1. En el Navegador del sistema, pulse **Personalización > Motor de documentos > Procesos**. Se abre el formulario Definición del proceso.
2. Escriba `EXWorkOrder.open.initial` en Nombre del proceso.
3. Pulse **Añadir**.
4. Escriba `$.format="EXWorkOrder"` en la ficha Expresiones iniciales.
5. Pulse **Guardar**.
6. Introduzca la siguiente información en la ficha RAD.

Campo	Valor
Expresiones evaluadas antes de una llamada RAD	<code>\$.number.record="EXWorkOrder";\$.number.type="string"</code>
Aplicación RAD	<code>getnumb</code>
Nombres de parámetros	Valores de parámetros
name	<code>\$.number.record</code>
index	ID in <code>\$.file</code>
text	<code>\$.number.type</code>

7. Pulse **Guardar**.

Creación de definiciones de pantallas de visualización de la aplicación

Como usuario crea las definiciones de pantallas de visualización de la aplicación de las pantallas abrir, cerrar y ver de los formularios de pedidos de trabajo que permiten a los usuarios abrir, cerrar y ver registros de pedidos de trabajo.

Para crear una definición de pantalla de visualización de la aplicación para abrir, cerrar y ver:

1. En el Navegador del sistema, pulse **Personalización > Herramientas de personalización > Pantallas de visualización**. Se abre el formulario Definición de la pantalla de visualización de la aplicación.
2. Escriba `EXWorkOrder.open` en ID de la pantalla.

3. Introduzca la siguiente información:

Campo	Valor
ID de la pantalla	EXWorkOrder.open
Título	Abrir nuevo pedido de trabajo
Formulario	\$L.format
E/S (si RIO)	true
Ante la opción 0:	volver a dibujar pantalla
Idioma	ES

4. Pulse **Añadir** y, a continuación, **Aceptar**.

Para este ejemplo, también debe crear definiciones de pantallas de visualización de la aplicación para la pantalla de cierre.

1. En el Navegador del sistema, pulse **Personalización > Herramientas de personalización > Pantallas de visualización**.
Se abre el formulario Definición de la pantalla de visualización de la aplicación.
2. Utilice la siguiente información:

Campo	Valor
ID de la pantalla	EXWorkOrder.close
Título	Cerrar pedido de trabajo
Formulario	\$L.format
E/S (si RIO)	true
Ante la opción 0:	volver a dibujar pantalla
Idioma	ES

3. Pulse **Añadir** y, a continuación, **Aceptar**.

Para este ejemplo, también debe crear definiciones de pantallas de visualización de la aplicación para la pantalla ver.

1. En el Navegador del sistema, pulse **Personalización > Herramientas de personalización > Pantallas de visualización**.
Se abre el formulario Definición de la pantalla de visualización de la aplicación.
2. Utilice la siguiente información:

Campo	Valor
-------	-------

ID de la pantalla	EXWorkOrder.view
Título	Ver nuevo pedido de trabajo
Formulario	\$L.format
E/S (si RIO)	true
Ante la opción 0:	volver a dibujar pantalla
Idioma	ES

3. Pulse **Añadir** y, a continuación, **Aceptar**.

Creación de definiciones de opciones de visualización de la aplicación

Este procedimiento proporciona instrucciones detalladas para crear definiciones de opciones de visualización de la aplicación (añadir, cancelar, rellenar y encontrar) para las siguientes definiciones de WorkOrder:

- abrir
- cerrar
- ver

Al hacer esto, repetirá los mismos pasos cuatro veces para cada una de las definiciones de pantalla de WorkOrder (abrir, cerrar, ver). Sin embargo, cada vez introduce diferentes valores en los campos del formulario Definición de la opción de visualización de la aplicación. Las tablas situadas después de los pasos proporcionan los valores que necesita para configurar las opciones de visualización de la aplicación.

Para obtener más información sobre las opciones de visualización de la aplicación, consulte los temas de la Ayuda en línea para la aplicación que se visualiza en el servidor de la Ayuda en línea de HP Service Manager.

Tras crear las definiciones de la pantalla de visualización de la aplicación para abrir, cerrar y ver, debe crear las opciones de visualización de la aplicación para las pantallas abrir, cerrar y ver.

Para crear una definición de opción de visualización de la aplicación:

1. En el Navegador del sistema, pulse **Personalización > Herramientas de personalización > Opciones de visualización**.
Se abre el formulario Definición de la opción de visualización de la aplicación.
2. Escriba `EXWorkOrder.open` en ID de la pantalla.
3. Introduzca la siguiente información:

Campo	Valor
-------	-------

ID único	Generado automáticamente (EXWorkOrder.open_add)
Acción	add
Opción de interfaz	4
Opción de texto	4
Banco	1
Condición	true
Etiqueta predeterminada	Añadir

4. Pulse **Añadir**.
5. Pulse **Aceptar**.
6. Repita los pasos 1 - 6 para cada una de las siguientes definiciones de opciones de visualización mediante los valores proporcionados en la tabla.

Nota: en algunos casos el valor necesario para el campo Acción no está disponible en la lista desplegable, pero puede escribir el valor aplicable en el campo.

Abrir - Cancelar

Campo	Valor
ID de la pantalla	EXWorkOrder.open
ID único	Generado por el sistema. (EXWorkOrder.open_cancel)
Acción	back
Opción de interfaz	3
Opción de texto	3
Banco	1
Condición	true
Etiqueta predeterminada	Cancelar

Abrir - Rellenar

Campo	Valor
ID de la pantalla	EXWorkOrder.open

ID único	Generado por el sistema. (EXWorkOrder.open_fill)
Acción	fill
Opción de interfaz	9
Opción de texto	9
Banco	1
Condición	true
Etiqueta predeterminada	Rellenar

Abrir - Encontrar

Campo	Valor
ID de la pantalla	EXWorkOrder.open
ID único	Generado por el sistema. (EXWorkOrder.open_find)
Acción	find
Opción de interfaz	8
Opción de texto	8
Banco	1
Condición	true
Etiqueta predeterminada	Encontrar

Cerrar - Cancelar

Campo	Valor
ID de la pantalla	EXWorkOrder.close
ID único	Generado por el sistema. (EXWorkOrder.close_cancel)
Acción	back
Opción de interfaz	3

Opción de texto	3
Banco	1
Condición	true
Etiqueta predeterminada	Cancelar

Cerrar - Cerrar

Campo	Valor
ID de la pantalla	EXWorkOrder.close
ID único	Generado por el sistema. (EXWorkOrder.close_close)
Acción	close
Opción de interfaz	5
Opción de texto	5
Banco	1
Condición	true
Etiqueta predeterminada	Cerrar

Cerrar - Rellenar

Campo	Valor
ID de la pantalla	EXWorkOrder.close
ID único	Generado por el sistema. (EXWorkOrder.close_fill)
Acción	fill
Opción de interfaz	9
Opción de texto	9
Banco	1
Condición	true
Etiqueta	Rellenar

predeterminada	
----------------	--

Cerrar - Encontrar

Campo	Valor
ID de la pantalla	EXWorkOrder.close
ID único	Generado por el sistema. (EXWorkOrder.close_find)
Acción	find
Opción de interfaz	8
Opción de texto	8
Banco	1
Condición	true
Etiqueta predeterminada	Encontrar

Ver - Cancelar

Campo	Valor
ID de la pantalla	EXWorkOrder.view
ID único	Generado por el sistema. (EXWorkOrder.view_cancel)
Acción	back
Opción de interfaz	3
Opción de texto	3
Banco	1
Condición	true
Etiqueta predeterminada	Cancelar

Ver - Rellenar

Etiqueta predeterminada	Rellenar
-------------------------	----------

ID de la pantalla	EXWorkOrder.view
ID único	Generado por el sistema. (EXWorkOrder.view_fill)
Acción	fill
Opción de interfaz	9
Opción de texto	9
Banco	1
Condición	true
Etiqueta predeterminada	Rellenar

Ver - Encontrar

Campo	Valor
ID de la pantalla	EXWorkOrder.view
ID único	Generado por el sistema. (EXWorkOrder.view_find)
Acción	find
Opción de interfaz	8
Opción de texto	8
Banco	1
Condición	true
Etiqueta predeterminada	Encontrar

Ver - Guardar

Campo	Valor
ID de la pantalla	EXWorkOrder.view
ID único	Generado por el sistema. (EXWorkOrder.view_save)

Acción	save
Opción de interfaz	4
Opción de texto	4
Banco	1
Condición	true
Etiqueta predeterminada	Guardar

Creación de una definición de estado

Estas definiciones de estado del pedido de trabajo de muestra especifican los procesos que deben utilizarse y las acciones permitidas cuando un usuario abre, cierra o visualiza un registro de pedido de trabajo.

Nota: Consulte [Descripciones de campos de Definición de estado en la página 26](#) para obtener información descripciones de los campos de Definición de estado.

Para crear una definición de estado para abrir:

1. En el Navegador del sistema, pulse **Personalización > Motor de documentos > Estados**. Se abre el formulario Definición de estado.
2. Introduzca la siguiente información:

Campo	Valor
Estado	EXWorkOrder.open
Pantalla de visualización	EXWorkOrder.open
Proceso de inicialización	EXWorkOrder.open.initial
Formato	\$L.format
Condición de entrada	true

3. Pulse **Añadir**.

Para crear una definición de estado para cerrar:

1. En el Navegador del sistema, pulse **Personalización > Motor de documentos > Estados**. Se abre el formulario Definición de estado.
2. Introduzca la siguiente información:

Campo	Valor
Estado	EXWorkOrder.close
Pantalla de visualización	EXWorkOrder.close
Proceso de inicialización	EXWorkOrder.close.initial
Formato	\$L.format
Condición de entrada	true

3. Pulse **Añadir**.

Para crear una definición de estado para visualizar:

1. En el Navegador del sistema, pulse **Personalización > Motor de documentos > Estados**. Se abre el formulario Definición de estado.
2. Introduzca la siguiente información:

Campo	Valor
Estado	EXWorkOrder.view
Pantalla de visualización	EXWorkOrder.view
Proceso de inicialización	EXWorkOrder.view.initial
Formato	\$L.format
Condición de entrada	true

3. Pulse **Añadir**.

Para crear una definición de estado para buscar:

1. En el Navegador del sistema, pulse **Personalización > Motor de documentos > Estados**. Se abre el formulario Definición de estado.
2. Introduzca la siguiente información:

Campo	Valor
Estado	EXWorkOrder.search
Pantalla de visualización	db.search
Proceso de inicialización	EXWorkOrder.view.search

Campo	Valor
Formato	\$L.format
Condición de entrada	true

3. Pulse **Aceptar**.

Adición de un botón de cierre de pedido de trabajo

Cuando un usuario termina una o varias tareas, el pedido de trabajo debe cerrarse. Este procedimiento describe cómo añadir un botón de cierre al formulario EXWorkOrder y actualizar el estado del pedido de trabajo que debe cerrarse. Este procedimiento añade una definición de la opción de visualización de la aplicación, una definición de estado para EXWorkOrder.view y una definición de proceso para EXWorkOrder.close.

Nota: este procedimiento no explica cómo debe incluirse la validación de esta actividad. Para añadir una validación, utilizaría Control de formatos.

Para crear una definición de la opción de visualización de la aplicación para EXWorkOrder.view:

1. En el Navegador del sistema, pulse **Personalización > Herramientas de personalización > Opciones de visualización**.
Se abre el formulario Definición de la opción de visualización de la aplicación.
2. Escriba `EXWorkOrder.view` en ID de la pantalla.
3. Introduzca la siguiente información:

Campo	Valor
ID único	Generado automáticamente (EXWorkOrder.view_close)
Acción	close
Opción de interfaz	5
Opción de texto	5
Banco	1
Condición	true
Etiqueta predeterminada	Cerrar

4. Pulse **Añadir**.
5. Pulse **Aceptar**.

Para actualizar la definición de estado EXWorkOrder.view:

1. En el Navegador del sistema, pulse **Personalización > Motor de documentos > Estados**.
Se abre el formulario Definición de estado.

2. En el campo Estado, escriba **EXWorkOrder.view** y pulse **Buscar**. Se abre el formulario de definición de estado EXWorkOrder.view.
3. Añada lo siguiente para actualizar el formulario de definición de estado.

Campo	Valor
Acción de visualización	close
Nombre del proceso	EXWorkOrder.close
Condición	true

4. Pulse **Guardar** y después pulse **Aceptar**.

Para añadir un registro de definición de procesos para EXWorkOrder.close:

1. En el Navegador del sistema, pulse **Personalización > Motor de documentos > Procesos**. Se abre el formulario Definición del proceso.
2. Escriba `EXWorkOrder.close` en Nombre del proceso.
3. Pulse **Añadir**.
4. En la ficha Expresiones iniciales, escriba la siguiente expresión:
 - `status in $L.file="Closed"`
 - `$L.mode="closed"`
5. Pulse **Guardar**.
6. En la ficha RAD, introduzca la siguiente información:

Campo	Valor
Expresiones evaluadas antes de una llamada RAD	<code>\$L.EXaction="update"</code>
Aplicación RAD	<code>se.base.method</code>
Condición	true
Nombres de parámetros	Valores de parámetros
file	<code>\$L.file</code>
prompt	<code>\$L.EXaction</code>
second file	<code>\$L.file.save</code>
record	<code>\$L.fc</code>
second.record	<code>\$L.object</code>
boolean1	false

7. Pulse **Guardar** y después pulse **Aceptar**.

Creación de un asistente para el pedido de trabajo

Este ejemplo emplea un asistente para crear un pedido de trabajo porque el asistente simplifica la tarea de abrir el pedido de trabajo para el usuario y la funcionalidad ya está disponible en el sistema.

En este ejemplo se utiliza la herramienta de creación de asistentes para crear un asistente que permita a los usuarios crear un pedido de trabajo desde el módulo Gestión de incidentes. Si un usuario crea un pedido de trabajo a partir de un incidente, el asistente solicitará al usuario información y rellenará automáticamente algunos campos del pedido de trabajo.

Para crear un asistente para el formulario de pedido de trabajo:

1. En el Navegador del sistema, pulse **Personalización > Asistentes**.
Se abre el formulario Info. del Asistente.
2. Escriba `Create Workorder - 1` en Nombre del Asistente.
3. Pulse **Añadir** para crear el registro Asistente.
4. En la ficha Info. del Asistente, introduzca la siguiente información:

Campo	Valor
Descripción breve	Cree un nuevo pedido de trabajo.
Título de ventana	Create Workorder
Título	Create Workorder
¿Iniciar nodo?	Seleccione la opción (establecida en true) para indicar que éste es el primer asistente de una serie de asistentes en los casos en que tenga una serie de registros de asistentes.

5. Pulse **Guardar**.
6. En la ficha Selección de archivos del formulario Info. del Asistente, seleccione la ficha **Selección de \$L.file**.
7. En la ficha Selección de \$L.file, introduzca la siguiente información:

Campo	Valor
Creación de un registro	true
del tipo	EXWorkOrder

8. Pulse **Guardar**.
9. En la ficha Uso, introduzca la siguiente información:

Campo	Valor
Uso del Asistente	Pulse Pedir entrada del usuario .
Subformulario para mostrar	Escriba createWO.assigneeAndCIs.
Pantalla de visualización	Escriba wizard.display.
Permitir la opción "Finalizar"	Seleccione la opción (establézcala en true) para tener un botón Finalizar en el formulario de asistente.

10. Pulse **Guardar**.

11. En la ficha Acciones, introduzca la siguiente información:

Campo	Valor
Realizar acciones en	Seleccione Archivo actual (\$L.file).
Acciones que se han de realizar > Expresiones	initiator in \$L.file=operator() status in \$L.file="New" RelatedID in \$L.file=number in \$relatedRec category in \$L.file=category in \$relatedRec impact in \$L.file=initial.impact in \$relatedRec urgency in \$Lfile=severity in \$relatedRec priority in \$L.file=priority.code in \$relatedRec
Mostrar los registros al finalizar	Pulse Mostrar los registros al finalizar .
Modo	Seleccione Añadir.

12. Pulse **Guardar**.

13. En la ficha Expresiones de cancelación, introduzca la siguiente información:

Campo	Valor
Expresiones ejecutadas al cancelar	cleanup(\$relatedRec)

14. Pulse **Guardar** y después pulse **Aceptar**.

Adición de un registro de definición de procesos

El registro de definición de procesos define cómo responde el sistema a una acción de usuario. La definición de procesos utiliza expresiones RAD, JavaScript y llamadas a aplicaciones RAD existentes para realizar acciones en el registro actual, en este caso, un registro de pedido de trabajo.

Para añadir un registro de definición de procesos:

1. En el Navegador del sistema, pulse **Personalización > Motor de documentos > Procesos**. Se abre el formulario Definición del proceso.
2. Escriba **create.WorkOrder** en Nombre del proceso.
3. Pulse **Añadir**.
4. En la ficha Expresiones iniciales, escriba la siguiente expresión:
 - **`$L.void=fduplicate($relatedRec, $L.file)`**
 - **`$relatedCIs={}`**
5. Pulse **Guardar**.
6. En la ficha Javascript inicial, escriba la siguiente expresión:

```
system.vars.$relatedCIs=system.library.BSGFunctions.getMembers(system.vars.$L-
_file.affected_item, false, 3)
```

Nota: esta expresión debe introducirse en una línea.

7. Pulse **Guardar**.
8. En la ficha RAD, introduzca la siguiente información:

Campo	Valor
Expresiones evaluadas antes de una llamada RAD	<p>Escriba las siguientes dos expresiones:</p> <pre>\$L.wiz.name="Create Workorder-1"</pre> <pre>if (not null(logical.name in \$L.file)) then (\$relatedCIs=insert (\$RelatedCIs, 1, 1, logical.name in \$L.file))</pre> <p>Nota: asegúrese de introducir esta expresión en una línea. Tenga en cuenta además que al introducir esta expresión, no hay ningún espacio tras la palabra insert. Por ejemplo, la expresión anterior continúa en la palabra 'insert' como se indica a continuación:</p> <pre>insert(\$RelatedCIs, 1, 1, logical.name in \$L.file))</pre>
Aplicación RAD	wizard.run
Nombres de parámetros	name
Valores de parámetros	\$L.wiz.name
Nombres de parámetros	text
Valores de parámetros	\$L.exit
Condición	true

9. Pulse **Guardar**.
10. En la ficha Expresiones finales, escriba `cleanup($relatedRec)`.
11. Pulse **Guardar** y después pulse **Aceptar**.

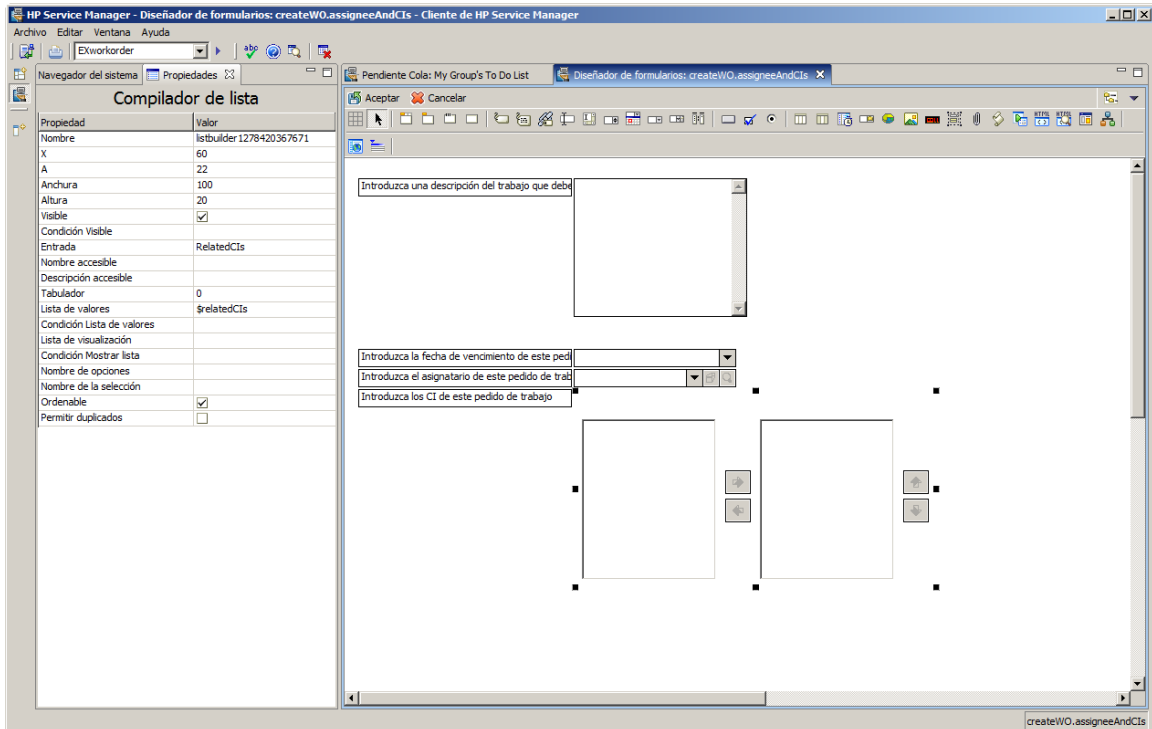
Creación de un formulario de entrada para el asistente

Este formulario es el primer formulario que se muestra en el asistente de pedidos de trabajo. El usuario introduce la información que debe proporcionar para que el asistente cree el registro de pedido de trabajo. Utilice el Diseñador de formularios para crear este formulario. Para este ejemplo, el nombre del formulario es createWO.assigneeAndCIs.

Para crear un formulario de entrada para el asistente:

1. En el Navegador del sistema, pulse **Personalización > Diseñador de formularios**. Se abre el nuevo formulario de búsqueda/creación de Diseñador de formularios.
2. Escriba **createWO.assigneeAndCIs** en Formulario.
3. Pulse **Nuevo**. No necesita utilizar el asistente del Diseñador de formularios para este formulario.
4. Cree los siguientes campos de entrada en el formulario:
 - Descripción (introduzca una descripción del trabajo que debe realizarse)
 - Fecha de vencimiento (introduzca la fecha de vencimiento de este pedido de trabajo)
 - Asignatario (introduzca el asignatario de este pedido de trabajo)
 - CI (introduzca los CI de este pedido de trabajo)
5. El campo Propiedades de este formulario debe incluir:
 - Entrada: **RelatedCIs**
 - Lista de valores: **\$relatedCIs**
 - Ordenable: verificado
6. Pulse **Guardar**.

En la siguiente figura se muestra un formulario createWO.assigneeAndCIs de muestra para el formulario de entrada del asistente EXWorkorder.



Modificación de los formularios de incidentes de cierre y actualización

En el ejemplo del pedido de trabajo, tiene que modificar los formularios de incidentes de cierre y el de actualización para poder ver los pedidos de trabajo asignados al incidente y, a continuación, pulsar dos veces y leer o editar el pedido de trabajo desde el incidente. Utilice el Diseñador de formularios para actualizar los formularios *IM.update.incident* e *IM.close.incident*. Para ello, debe añadir una ficha y luego un formulario en la ficha en los formularios *IM.update.incident* e *IM.close.incident*.

Para modificar un formulario de incidente de actualización:

Nota: utilice estos mismos procedimientos generales para actualizar el formulario de incidente de cierre.

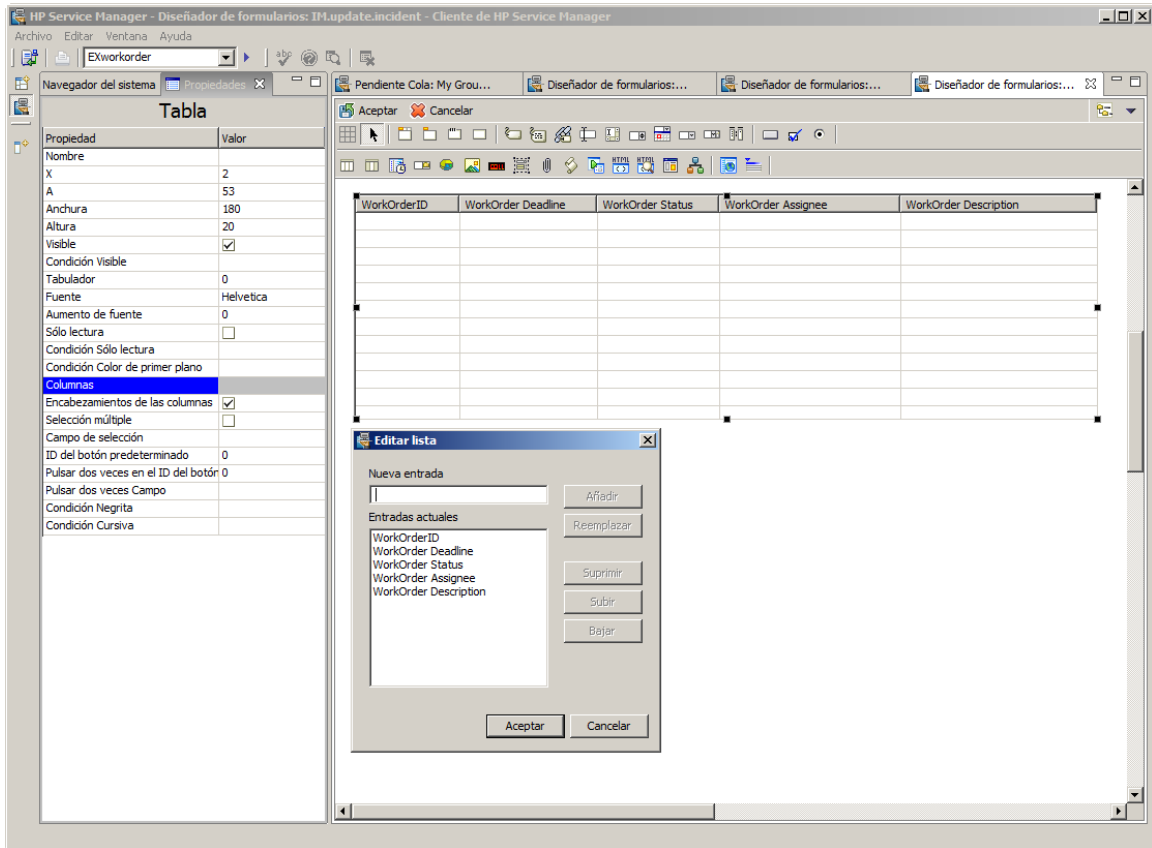
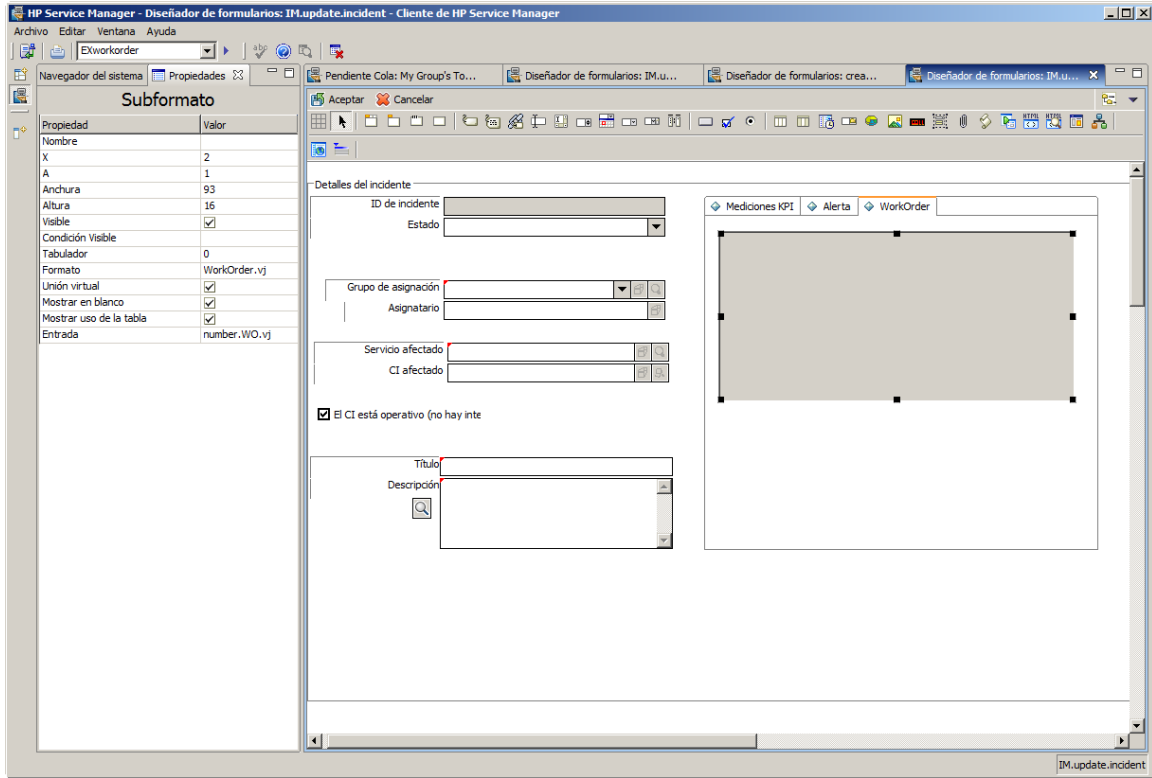
1. En el Navegador del sistema, pulse **Personalización > Diseñador de formularios**. Se abre el formulario de búsqueda del Diseñador de formularios.
2. Escriba **IM.update.incident** en Formulario.
3. Pulse **Buscar**.
4. Pulse **Diseñar**.
5. Añada una nueva ficha a la ventana de anotaciones existente.
6. Establezca el título de la ficha en WorkOrder.
7. Añada un subformulario con las siguientes propiedades:
 - Visible: verificado
 - Formato: WorkOrder.vj
 - Unión virtual: verificado
 - Mostrar en blanco: verificado
 - Mostrar uso de la tabla: verificado
 - Entrada: number.WO.vj
8. Pulse **Guardar**.
9. Utilice la herramienta Diseñador de formularios para crear el formulario WorkOrder.vj.
 - WorkOrderID (ID de pedido de trabajo): ID
 - WorkOrder Deadline (Fecha de vencimiento del pedido de trabajo): deadline
 - WorkOrder Status (Estado de pedido de trabajo): status
 - WorkOrder Assignee (Asignatario de pedido de trabajo): assign.name
 - WorkOrder Description (Descripción de pedido de trabajo): description
10. Repita los pasos 1 a 8 para **IM.close.incident**.

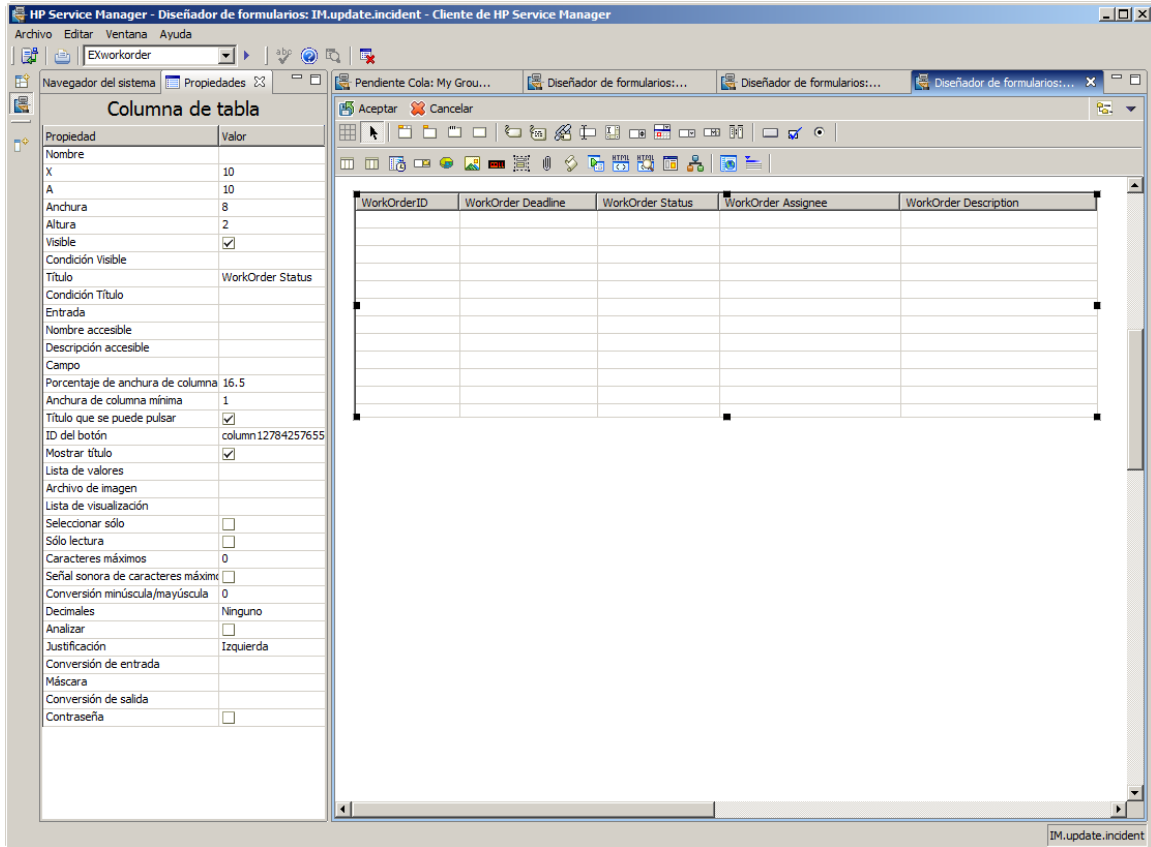
Nota: puede utilizar el formulario WorkOrder.vj creado en el paso 9 para el join virtual.

En las siguientes figuras se muestran ejemplos de las fichas para añadir.

Guía del Motor de documentos

Capítulo 7: Visión general de un pedido de trabajo de muestra





Creación de un alias en la tabla probsummary para un enlace

En el pedido de trabajo de muestra los usuarios deben poder acceder a un pedido de trabajo desde un registro de incidente cuando el incidente tiene un pedido de trabajo asociado a él. Para ello, debe crear un alias en la tabla probsummary y utilizarlo para crear un enlace entre la tabla probsummary y la tabla EXWorkOrder.

Para añadir un alias a la tabla probsummary:

1. En el Navegador del sistema, pulse **Personalización > Diccionario de base de datos**. Se abre el formulario Diccionario de la base de datos.
2. Escriba `probsummary` en Nombre del archivo.
3. Pulse **Buscar**.
4. En la ficha Campos, seleccione el campo numérico y pulse **Editar campo/clave**.
5. Pulse **Crear alias** y escriba `number.WO.vj` en Nombre y carácter en Tipo.
6. Pulse **Aceptar**.

Para enlazar la tabla EXWorkOrder con la tabla probsummary:

1. En el Navegador del sistema, pulse **Personalización > Herramientas de personalización > Enlaces**.
Se abre el formulario Archivo de enlace.
2. Escriba `probsummary` en Nombre.
3. Pulse **Buscar**.
4. Coloque el cursor tras la última entrada para crear una línea en blanco para una nueva entrada.
5. En Nombre del campo de origen, escriba `number.WO.vj`.
6. Seleccione toda la línea nueva y pulse **Seleccionar línea** del menú Más acciones.
7. Introduzca la siguiente información:

Campo	Valor
Campo (Desde/Origen)	<code>number.WO.vj</code>
Archivo (A/Destino)	<code>EXWorkOrder</code>
Campo (A/Destino)	<code>RelatedID</code>
Consulta	<code>\$query</code>
Expresiones	<code>\$query="RelatedID=\""+number in \$File+"\""</code>

8. Pulse **Guardar** y después pulse **Atrás**.
9. Repita los pasos para crear una línea de enlace para el campo ID.
10. Introduzca la siguiente información:

Campo	Valor
Campo (Desde/Origen)	<code>ID</code>
Archivo (A/Destino)	<code>EXWorkOrder</code>
Campo (A/Destino)	<code>ID</code>
Consulta	<code>\$query</code>
Expresiones	<code>\$query="ID=\""+nullsub(cursor.field.contents(), "xxx")+\""</code>

11. Pulse **Guardar** y después pulse **Atrás**.

Modificación de la definición de proceso `im.set.close`

Este proceso debe modificarse de modo que un usuario no pueda cerrar un incidente si hay pedidos de trabajo todavía abiertos para el incidente.

Para modificar un registro de definición de proceso:

1. En el Navegador del sistema, pulse **Personalización > Motor de documentos > Procesos**.
Se abre el formulario Definición del proceso.

2. Escriba **im.set.close** en Nombre del proceso.
3. Pulse **Buscar**.
4. En la ficha Javascript inicial, escriba el siguiente JavaScript:

```
var WO=new SCFile ("EXWorkOrder")

var FoundOpenWO=WO.doSelect ("RelatedID=\""+system.vars.$L_
file.number + "\"+" and status ~=\"\" + "Closed" + "\"")

if (FoundOpenWO == RC_SUCCESS)

{

system.vars.$openWO=true;

}

else

{

system .vars.$openWO=false;

}

}
```

5. Pulse **Guardar**.
6. En la ficha RAD, introduzca la siguiente información:

Campo	Valor
Nota: algunos de los campos de la ficha RAD se han rellenado previamente con valores que no necesita modificar.	
Aplicación RAD: seleccione la sección vacía bajo la sección que llama la aplicación RAD us.consume.wrapper e introduzca la siguiente información:	
Expresiones evaluadas antes de una llamada RAD	\$L.text="There are still open work orders. This incident cannot be closed yet. (Todavía hay pedidos de trabajo abiertos. Este incidente no puede cerrarse aún.)"
Aplicación RAD	apm.mb.ok
Condición	\$openWO=true
Nombres de parámetros	text
Valores de parámetros	\$L.text

7. Pulse **Guardar**.
8. En la ficha Expresiones finales, introduzca la siguiente información:

- if (\$openWO=true) then (\$L.exit="badval")
- \$L.exit="closestate"

9. Pulse **Guardar** y después pulse **Aceptar**.

Comprobación del pedido de trabajo de muestra

Tras completar las tareas para crear un sistema de pedido de trabajo, debe verificar que funciona correctamente. Para verificar la funcionalidad básica del pedido de trabajo de muestra, haga lo siguiente:

- En Gestión de incidentes, busque un incidente abierto o cree uno.
- Utilice el menú Más acciones para crear un pedido de trabajo. Create Workorder (Crear pedido de trabajo) debería mostrarse en el menú Más acciones.
- En el asistente de pedidos de trabajo, introduzca los datos para crear un pedido de trabajo para el incidente.
- Añada el pedido de trabajo al incidente y guarde los cambios en el incidente.
- Abra el incidente para editar y desde la ficha WorkOrder (Pedido de trabajo) edite el pedido de trabajo.
- Guarde los cambios realizados.
- Vuelva a abrir el incidente y esta vez edite el pedido de trabajo y ciérrelo.
- Ahora debería poder cerrar el incidente.
- Repita estos pasos, pero esta vez cree dos pedidos de trabajo para un incidente.
- Cierre sólo un pedido de trabajo y luego intente cerrar el incidente. El sistema debería generar un mensaje de error donde se indique que todavía hay pedidos de trabajo abiertos para el incidente, y por consiguiente no puede cerrarse.
- Cierre todos los pedidos de trabajo del incidente. Ahora debería poder cerrar el incidente.

