Manual de usuario

IGEO\_LCM

Llamadas a Terceros desde Bonita

Escriba el Nº del Proyecto

Versión 0.1 | Nivel de Seguridad N1 - Interno

Registro de Cambios

|  |  |  |  |
| --- | --- | --- | --- |
| **Versión** | **Fecha** | **Revisado** | **Resumen de los cambios producidos** |
| 0.1 | 21/04/2015 |  | Borrador |

Versión de la Plantilla: 2.0

Índice

1 | Introducción 4

2 | Configuración 5

3 | Utilización *llamadaTercerosIframe* 7

3.1 Primer parámetro 7

3.2 Segundo parámetro 7

3.3 Tercer parámetro 8

3.4 Cuarto parámetro 8

3.5 Quinto parámetro 8

3.6 Sexto parámetro 8

3.7 Séptimo parámetro 9

3.8 Octavo parámetro 9

3.8.1 Ampliación 9

3.9 Noveno parámetro 10

4 | Ejemplo de uso 11

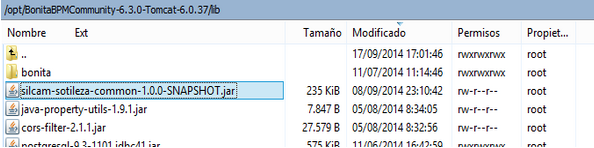
# Introducción

El presente documento tiene como finalidad describir las pautas a seguir para realizar llamadas a terceros desde formularios de Bonita BPM. Para de esta forma poder seleccionar un tercero.

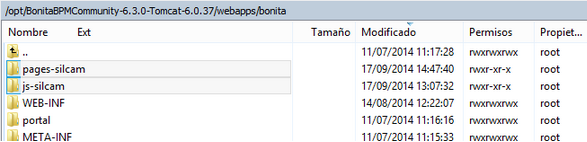
De todos los terceros existentes sólo aplica para las concesiones (también llamados Títulos o Derechos Mineros).

# Configuración

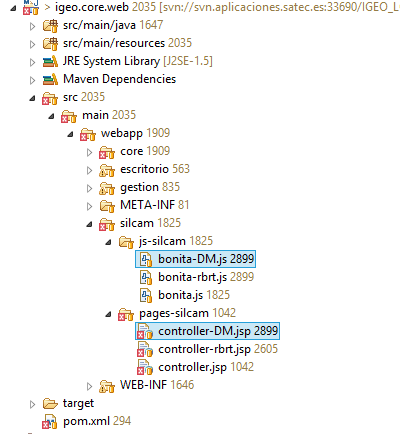
Es necesario realizar una serie de cambios en el servidor de bonita para el correcto funcionamiento. Por pasos:  
  
**Paso 1 -** Meter el SotilezaCommons.jar en el directorio lib de servidor. Es necesario para poder generar los tickets.



**Paso 2 -** Dentro del proyecto /bonita desplegado se deben crear los directorios llamados *pages-silcam* y *js-silcam*.



**Paso 3 -** Copiar dentro de pages-silcam el fichero **controller-DM.jsp** y copiar dentro de js-silcam el fichero **bonita-DM.js**. Están en el svn en el proyecto igeo.core.web aunque también podéis cogerlos de ih.



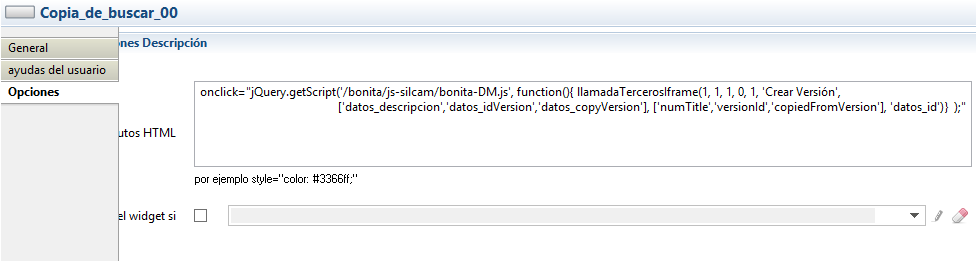
**Paso 4 -** Añadir a JAVA\_OPTS la variable ruta\_properties con la ruta al directorio donde está el fichero igeo.properties. Cómo ejemplo, en el servidor de desarrollo está puesto en el catalina.sh tal que así:

-Druta\_properties=/siva/oc4j\_data\_igeo

**Paso 5-** Una vez hecho todo esto reiniciamos el servidor y ya podríamos utilizar, en los formularios de bonita, la aplicación de terceros.

# Utilización *llamadaTercerosIframe*

Para realizar una llamada a los dialogs de terceros basta simplemente con poner un botón en el formulario de bonita y en el evento onclick realizar dicha llamada. Tal que así:



**onclick="jQuery.getScript('/bonita/js-silcam/bonita-DM.js', function(){ llamadaTercerosIframe(1, 1, 1, 0, 1, 'Crear Versión', ['datos\_descripcion','datos\_idVersion','datos\_copyVersion'], ['numTitle','versionId','copiedFromVersion'], 'datos\_id')} );"**

Es decir, en ese botón ejecutamos la función **llamadaTercerosIframe** que recibe 9 parámetros y es la que se encarga de todo. En los siguientes puntos se expone una explicación del uso de esos parámetros:

## Primer parámetro

Este parámetro es obligatorio y debe contener los valores 0 o 1.

El valor 1 indica que el derecho puede ser creado de cero. Esto significa que en el dialog aparecerá un botón para crear el derecho minero.

El valor 0 indica que el derecho no se podrá crear de cero desde el dialog.

## Segundo parámetro

Este parámetro es obligatorio y debe contener los valores 0 o 1.

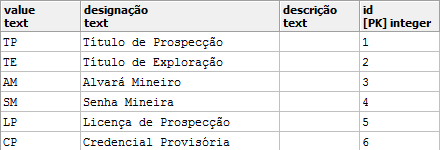
El valor 1 indica que el derecho puede ser modificado. Esto significa que, en el dialog, al presentar los datos en el formulario del derecho minero se podrán modificar antes de incorporarlo al expediente.

El valor 0 indica que el derecho no se podrá modificar en el dialog.

## Tercer parámetro

Este parámetro es obligatorio y debe contener el identificador del tipo de derecho minero a tratar. De esta forma sólo se permiten añadir derechos del tipo elegido.

Los valores posibles pueden encontrarse en la tabla tipo\_direito\_mineiro del esquema cadastro. En la actualidad existen los siguientes valores posibles:



Es decir, el parámetro debería contener uno de estos valores: 1, 2, 3, 4, 5, 6.

## Cuarto parámetro

Este parámetro es obligatorio y debe contener los valores 0 o 1.

El valor 1 indica que el derecho a incorporar deberá ser una copia.

## Quinto parámetro

Este parámetro es obligatorio y debe contener los valores 0 o 1.

El valor 1 indica que el derecho a incorporar deberá ser una versión.

## Sexto parámetro

Es obligatorio y debe contener un mensaje para denominar a la copia y/o versión que se va a crear. Si se desea puede ser una cadena vacía.

Anexo: Por defecto al crear una nueva versión se pasa a estado cancelado la versión vieja. Si se desea que la versión anterior no sea cancelada se debe añadir junto a la denominación el siguiente literal \_MANTER\_ESTADO\_ANTERIOR\_

Ej 1: Si se quiere denominar una nueva versión como “Versión de transmisión” y que no se cancele la versión anterior, este parámetro debería contener

“\_MANTER\_ESTADO\_ANTERIOR\_Versión de transmisión”

Ej 2: Si se quiere denominar una nueva versión como “Versión de transmisión” y que se cancele la versión anterior, este parámetro debería contener

“Versión de transmisión”

## Séptimo parámetro

Este parámetro es obligatorio.

Se corresponde con un array donde se colocan los nombres de los campos en el formulario donde se quiere almacenar la información que devuelve terceros.

Por ejemplo, si en el formulario has creado un campo input tipo text llamado datos1 y otro tipo hidden llamado datos2. Puedes usarlos para almacenar en ellos cualquier información que necesites que proceda del tercero que selecciones. En este caso, este tercer parámetro sería [‘datos1’,’datos2’]

## Octavo parámetro

Este parámetro es obligatorio.

Se corresponde con un array donde se colocan los nombres de los campos de terceros que quieres recuperar (los nombres de esos campos los podéis consultar ejecutando la llamada rest del servicio que se llama por debajo, p.e. <http://ih-vm-stccatastro.c.mad.interhost.com:8080/terceros/rest/personas/1> ).

Siguiendo con el ejemplo anterior (datos1 y datos2), vamos a suponer que:

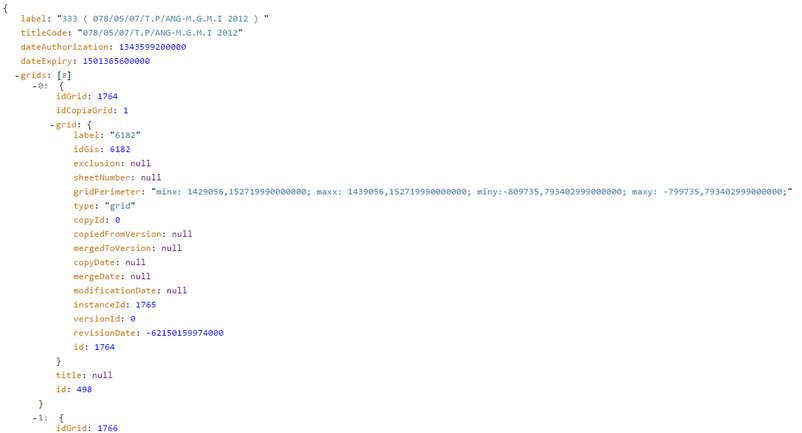
* En datos1 se va a almacenar el nombre.
* En datos2 se va a almacenar el nombre y los apellidos

En este caso, este tercer parámetro sería [‘nombre’,’ nombre+apellido1+apellido2’]

**Nota**: Como se puede ver para concatenar campos se usa el carácter +.  
   
**Nota 2**: El séptimo y octavo parámetros deben ser coherentes, es decir arrays de mismo tamaño y colocar en orden los valores para que coincidan.

### Ampliación

Dado un servicio rest.... por ejemplo el de las concesiones, nos devuelve algo tal que así:



Se puede ver que dentro de una concesión hay un array de parcelas llamado "grids", cada objeto de este array representa la relación de la concesión con una parcela. Nosotros para obtener los datos de dentro de uno de esos objetos lo vamos a indicar utilizando el string "->". Así le indicamos que es un array y lo que va hacer es devolvernos separado por comas lo que le indiquemos. De esta forma:

* Si ponemos **grids->grid.idGis:** nos devolverá un string con todos los grid.idGis de ese array separado por comas.
* Si ponemos **grids->id+grid.idGis:** nos devolverá un string con todos los valores\* de ese array separado por comas.\*Siendo "valores"los campos id  e grid.idGis concatenados con un espacio en blanco.

## Noveno parámetro

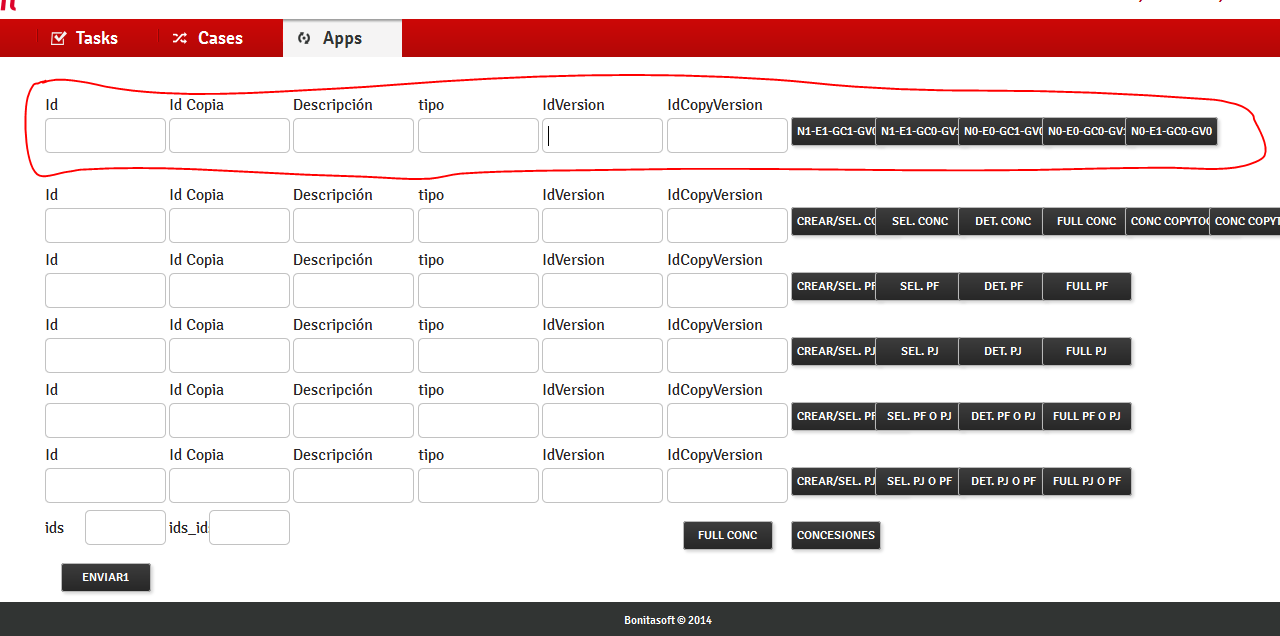
Este parámetro es obligatorio.

Se corresponde con el nombre del cuadro de texto (text o hidden) donde se desea almacenar el id del tercero seleccionado finalmente en el dialog.

**Nota**: A diferencia de como pasaba en la versión vieja (utilizada para el resto de terceros) el id se va a corresponder con la primary key del derecho minero, en lugar del campo id.

# Ejemplo de uso

En el procedimiento Terceros\_prototipo\_Form\_Creacion\_Sin\_Conector-2.1.bos se pueden probar las distintas opciones. Se deben incluir en bonita los ficheros controller-DM.jsp y bonita-DM.js ya que son los de prueba.



La primera fila, marcada de rojo, es la que permite probar estos dialogs para derechos mineros. Hay 5 ejemplos de uso, pero puede haber alguna variante más.

Los campos representados en el formulario son:

* Id: es el valor a través del que accedemos a una copia del tercero.
* Descripción: se utiliza para mostrar información de ese tercero.
* IdVersion: en caso de ser una copia siempre tendra el valor cero, se utiliza para marcar versiones.
* IdCopyVersion: en caso de ser una copia indica la versión a partir de la cual se ha creado la copia.