Application: gvSIG desktop - gvSIG bugs #3433 ApplicationLocator.getManager().getActiveComponent(Class) crea la ventana del documento como efecto colateral en ciertas circustancias

04/30/2015 01:35 PM - Cesar Martinez Izquierdo

Status:	Closed	**% Done:**	100%
Priority:	Normal	**Spent time:**	0.00 hour
Assignee:	Francisco Díaz Carsí		
Category:			
Target version:	2.2.0-2309		
Severity:	Minor	**Add-on version:**	
gvSIG version:	2.2.0	**Add-on build:**	
gvSIG build:		**Add-on resolve version:**	
Operative System:		**Add-on resolve build:**	
Keywords:		**Proyecto:**	
Has patch:		**Hito:**	
Add-on name:	Unknown		

Description

El método getActiveComponent en DefaultApplicationManager puede llegar a crear la ventana del documento en caso de que no exista, lo cual es un comportamiento indeseado (ya que getActiveComponent se llama desde muchos sitios y no debería tener "efectos colaterales"). Ver el bug relacionado para ver un código de ejemplo que muestra este comportamiento. En este código, tenemos un documento vista, sin ventana creada todavía, y abrimos la ventana de propiedades de la vista.

A continuación se llama a getActiveComponent, que llama a getActiveDocument(documentClass) para obtener el documento activo. Determina que el documento activo es la vista (ya que la ventana de propiedades de la vista tiene como modelo al mismo documento vista).
Entonces llama a getMainComponent sobre el documento vista. La implementación del método está en AbstractDocumentManager y en ella está el problema: si el documento activo es la vista se llamará a getMainWindow(doc) y por tanto se creará la vista.

Una solución podría ser distinguir los métodos de creación de la ventana de los que sólo deben obtenerla si existe (getMainWindow vs createMainWindow), pero para implementarlo habría que cambiar interfaces básicos del core.

Related issues:

Related to Application: gvSIG desktop - gvSIG bugs # 3432: ApplicationLocator...	**Closed**	**04/30/2015**
Related to Application: gvSIG desktop - gvSIG bugs # 3434: No se abren la pro...	**Closed**	**04/30/2015**

Associated revisions

Revision 42165 - 05/04/2015 10:12 AM - Francisco Díaz Carsí

fixes #3432
fixes #3433
fixes #3434

Arreglado el método getActiveDocument en DefaultProject para que no intente devolver un documento si la window no es una instancia de IDocumentWindow.

Sincronizado el método refreshControl en la clase MDIFrame para evitar que se le invoque mientras se está ejecutando.

History

#1 - 04/30/2015 01:35 PM - Cesar Martinez Izquierdo

- Related to gvSIG bugs #3432: ApplicationLocator.getManager().getActiveComponent() junto con EditingExtension pueden crear bucles infinitos

added

#2 - 04/30/2015 02:44 PM - Cesar Martinez Izquierdo

- Related to gvSIG bugs #3434: No se abren la propiedades de una vista copiada y pegada added

#3 - 05/01/2015 03:36 PM - Joaquín del Cerro Murciano

- Target version set to 2.2.0-2309

- Assignee set to Francisco Díaz Carsí

#4 - 05/01/2015 03:47 PM - Joaquín del Cerro Murciano

hasta donde he entendido de lo que cuentas el error estaria en el metodo getActiveDocument de DefaultProject. Este esta devolviendo un documento vista cuando se tropieza con la ventana de propiedades de esta cuando no deberia hacerlo. La ventana de propiedades no deberia ser relevante para saber si un documento esta activo.
La correccion de esto es simple.
Existe un interface que deben implementar todas las ventanas de los documentos, IDocumentWindow, asi que bastaria con que este metodo solo tenga en cuenta las ventanas que lo imlementen. Donde hace:

```
if (window instanceof SingletonWindow ) {
```

bastaria con añadirle que sea instancia de IDocumentWindow :

```
if (window instanceof SingletonWindow && window instanceof IDocumentWindow ) {
```

La cosaquedaria algo como:

```
public Document getActiveDocument(Class<? extends Document> documentClass) {
    ApplicationManager application = ApplicationLocator.getManager();

    Document document = null;
    IWindow[] windows = application.getUIManager().getOrderedWindows();
    IWindow window = null;
    for (int i = 0; i < windows.length; i++) {
        window = windows[i];
        if (window instanceof SingletonWindow && window instanceof IDocumentWindow ) {
    // Cogemos no la primera ventana, si no la primera
            // ventana de tipo documento (SingletonWindow).
            // Y por si las mosca no es un documento, atrapamos
            // los errores y continuamos si no puede hacer un cast
            // del Model a Document
            try {
                document = (Document) ((SingletonWindow) window).getWindowModel();
                if (documentClass == null) {
                    return document;
                }

                if (documentClass.isAssignableFrom(document.getClass())) {
                    return document;
                }
                if( document instanceof DocumentsContainer ) {
                    Document subdoc = ((DocumentsContainer)document).getActiveDocument(documentClass);
```

```
                return subdoc;
            }

        } catch (ClassCastException e) {
            // Do nothing, skip this window
        }
    }
  }
  return null;
}
```

#5 - 05/04/2015 10:13 AM - Francisco Díaz Carsí

- % Done changed from 0 to 100

- Status changed from New to Fixed

Applied in changeset svn-gvsig-desktop|r42165.

#6 - 05/07/2015 05:50 PM - Álvaro Anguix

- Status changed from Fixed to Closed