

# OpenJUMP Communication Plugin

## Inhalt

Einführung.....	1
Funktionsweise.....	1
OpenJUMP nach Fremdanwendung.....	1
Fremdanwendung nach OpenJUMP.....	2
Installation.....	2
Konfiguration.....	2
Ansteuerung der externen Anwendung.....	3
Ansteuerung OpenJUMP.....	3
Anwendung.....	4
Übernahme von Objektschlüsseln in OpenJUMP.....	5

## Einführung

Durch das CommnucationPlugin (CommPlugin) ist es jetzt möglich externe Anwendungen von OpenJUMP aus zu steuern oder OpenJUMP von externen Anwendungen zu steuern. Da in den meisten Fällen nur wenige Sachdaten in den Sachdatentabellen zu den Geometrien vorhanden sind, ist eine solche wechselseitige Ansteuerung sehr nützlich.

Entstanden ist das CommPlugin zur Verwendung mit Archikart, einer raumbezogenen Software die hauptsächlich im kommunalen Bereich verwendet wird. Mit OpenJUMP stand ein recht leistungsfähiges Open Source GIS zur Verfügung, was für viele Aufgaben einer Verwaltung völlig ausreicht. Einzigstes Manko war die Fehlende Verbindung zu den Sachdaten.

Die folgende Anleitung bezieht sich auf die Zusammenarbeit mit Archikart 3. Grundsätzlich ist es aber möglich auch mit anderen Anwendungen zu kommunizieren.

Das Plugin ist mit der OpenJUMP Version ab 1.4.2 lauffähig und getestet. Es würde auch mit früheren 1.4er Versionen laufen, allerdings wird seit Version 0.2 ein spezielle Funktion genutzt, die es erst seit der Version 1.4.2 gibt. Das Plugin funktioniert unter Linux und Windows.

## Funktionsweise

Der Austausch der Daten zwischen OpenJUMP und der Fremdanwendung erfolgt über zwei einfache Textdateien. Für jede Richtung jeweils eine.

### ***OpenJUMP nach Fremdanwendung***

Es wird eine Datei mit den in der Fremdanwendung anzuzeigenden Objekten erstellt. Diese enthält pro Objekt eine Zeile mit dem Wert der Attributspalte. Wenn die Datei komplett ist, wird eine Kommunikationsanwendung gestartet, die dann im Zielsystem die Objekte aus der Datei lesen und anzeigen muss.

## Fremdanwendung nach OpenJUMP

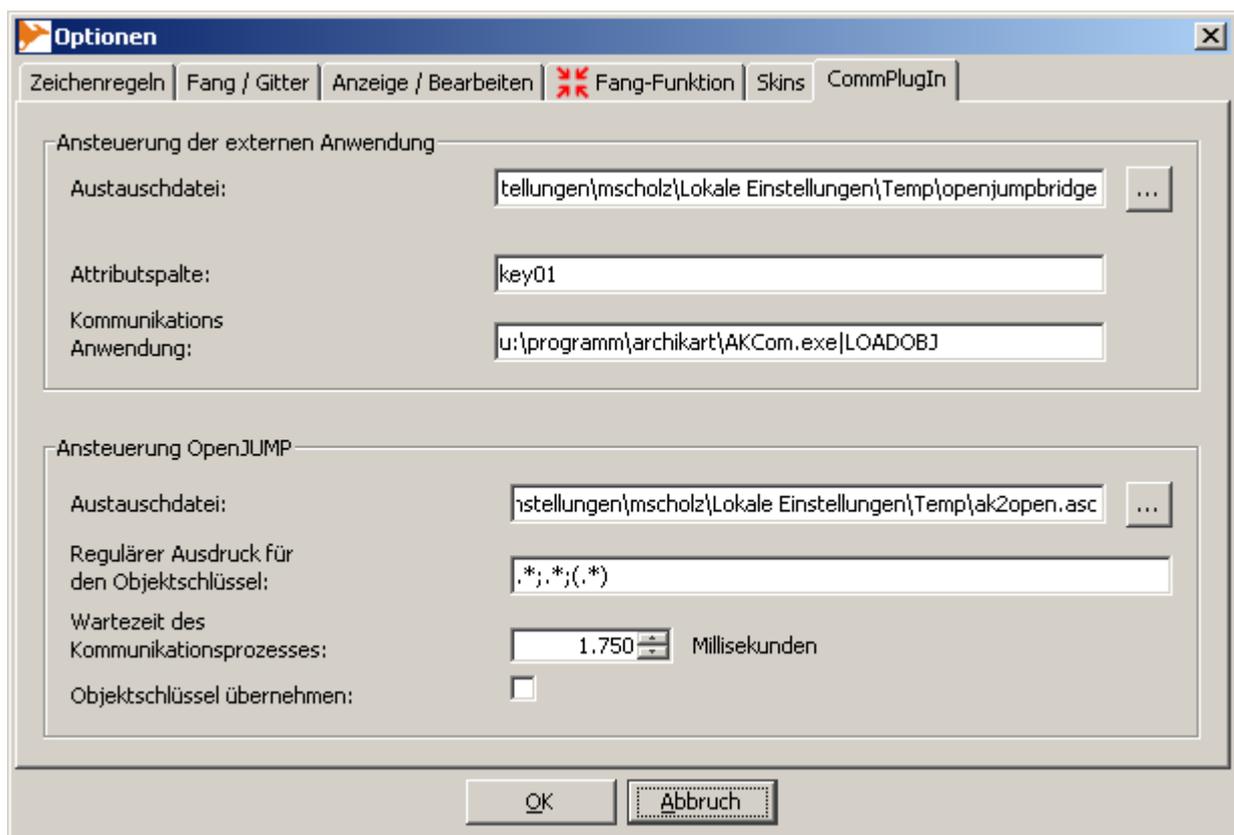
Von der Fremdanwendung wird eine Datei mit den anzuzeigenden Objekten erstellt. Diese Datei enthält eine Zeile pro Objekt. Aus einer Zeile wird über einen regulären Ausdruck der Suchschlüssel ermittelt, mit dem das Geometrieobjekt gesucht wird. Im Hintergrund läuft ein Prozess, der in einstellbaren Zeitintervallen prüft ob eine Austauschdatei vorhanden ist.

## Installation

Das CommPlugin ist ein ganz normales OpenJUMP Plugin und läßt sich somit einfach durch Kopieren der JAR Datei in das OpenJUMP lib/ext Verzeichnis installieren. Nach dem Start sollte das Plugin verfügbar sein.

## Konfiguration

Die Parameter werden über den Optionen Dialog(Menü Anpassung->Optionen->CommPlugIn) im OpenJUMP eingestellt. Der Dialog ist zweigeteilt für die Einstellungen in die jeweilige Richtung der Ansteuerung.



## **Ansteuerung der externen Anwendung**

Der vollständige Dateiname der Austauschdatei kann entweder über das Eingabefeld eingegeben, kopiert werden oder über den Button mit den drei Punkten ausgewählt werden. In die angegebene Datei schreibt OpenJUMP dann die markierten Objekte.

In das Feld „Attributspalte“ trägt man den Namen der Spalte ein, die einen eindeutigen Schlüssel für die externe Anwendung enthält. Der Wert der Spalte wird dann für das oder die markierten Objekt(e) in die Austauschdatei geschrieben. Die Spalten kann man sich im OpenJUMP über das Kontextmenü der jeweiligen Ebene (Schema anzeigen/bearbeiten oder Attribute anzeigen/bearbeiten) anzeigen lassen.

Zuletzt ist die Kommunikationsanwendung einzutragen, mit deren Hilfe der externen Anwendung signalisiert wird, dass sie neue Daten anzeigen soll. Es ist der vollständige Pfad zum Programm anzugeben. Wenn das Programm noch Parameter benötigt, dann sind diese durch ein „|“ Zeichen anstatt einem Leerzeichen getrennt anzugeben. Das ist sehr wichtig, da auch ein Dateiname Leerzeichen enthalten kann und somit eine eindeutige Zuordnung von Programmdatei und Parametern sonst nicht möglich ist. Das Plugin übergibt dann als letzten Parameter noch den Dateinamen.

Für die konkrete Verwendung mit Archikart sind die folgenden Einstellungen gebräuchlich.

- Austauschdatei
  - `C:\Dokumente und Einstellungen\mscholz\Lokale Einstellungen\Temp\openjumpbridge`
- Attributspalte
  - `key01`
- Kommunikationsanwendung
  - `u:\Pfad\zu\archikart\AKCom.exe|LOADOBJ`

Die Groß- und Kleinschreibung bei der Attributspalte spielt seit Version 0.2 keine Rolle mehr.

## **Ansteuerung OpenJUMP**

Genau wie oben kann vollständige Dateiname der Austauschdatei entweder über das Eingabefeld eingegeben, kopiert werden oder über den Button mit den drei Punkten ausgewählt werden.

Der reguläre Ausdruck für den Objektschlüssel beschreibt wie aus der gelesenen Zeile der Austauschdatei der Objektschlüssel extrahiert wird. Notwendig ist das, weil pro Zeile außer dem Objektschlüssel noch andere Daten stehen können. Der reguläre Ausdruck wird mit der Methode `java.lang.String.replaceAll()` auf die Zeile angewendet. Ersetzt wird dann durch die erste gefundene Gruppe (\$1). Das ist in der Regel der Ausdruck in der ersten runden Klammer. Somit muss der reguläre Ausdruck mindestens einen Ausdruck in runden Klammern enthalten. Siehe dazu [„\[http://java.sun.com/javase/6/docs/api/java/lang/String.html#replaceAll\\(java.lang.String,%20java.lang.String\\)\]\(http://java.sun.com/javase/6/docs/api/java/lang/String.html#replaceAll\(java.lang.String,%20java.lang.String\)\)“](http://java.sun.com/javase/6/docs/api/java/lang/String.html#replaceAll(java.lang.String,%20java.lang.String)).

Eine Zeile kann z.B. so aussehen `„DBS;Flurstück;FS1520980080009908800“`. Gebraucht wird die Zeichenkette `„FS1520980080009908800“` als Objektschlüssel. Der passende reguläre Ausdruck wäre dann `„*;*:(.*)“`. Damit wird beschrieben, dass mehrere beliebige Zeichen, dann ein Semikolon, dann wieder mehrere beliebige Zeichen, dann wieder ein Semikolon und schließlich der Objektschlüssel folgt.

Die Wartezeit für den Kommunikationsprozess legt fest, aller wie viel Millisekunden der Kommunikationsprozess nach einer Austauschdatei nachsehen soll. Da es passieren kann, dass das Erstellen der Datei länger als die Wartezeit benötigt, ist es sinnvoll die Datei an einem Ort zu erstellen und dann zu verschieben.

Für die konkrete Verwendung mit Archikart sind die folgenden Einstellungen gebräuchlich.

- Austauschdatei
  - C:\Dokumente und Einstellungen\mscholz\Lokale Einstellungen\Temp\ak2openasc
- Regulärer Ausdruck für den Objektschlüssel
  - \*.\*;.\*;(.\*)
- Wartezeit des Kommunikationsprozesses
  - 1750

Archikart schreibt die Austauschdatei deren Name „ak2openasc“ ist, in das TEMP Verzeichnis des Benutzers. Dieses Verzeichnis kann man über Arbeitsplatz->Eigenschaften->Erweitert->Umgebungsvariablen->Benutzervariablen in Erfahrung bringen.

Im Archikart ist dann für den jeweiligen Benutzer unter Einstellungen->GIS->GIS-Version „ProOpen“ auszuwählen. Wichtig ist noch, dass der Benutzer die Rechte für den „GIS Viewer-Treiber ProOpen Plus“ hat.

Der oder die übergebenen Objektschlüssel werden dann auf allen Ebenen gesucht und markiert, wenn sie gefunden werden.

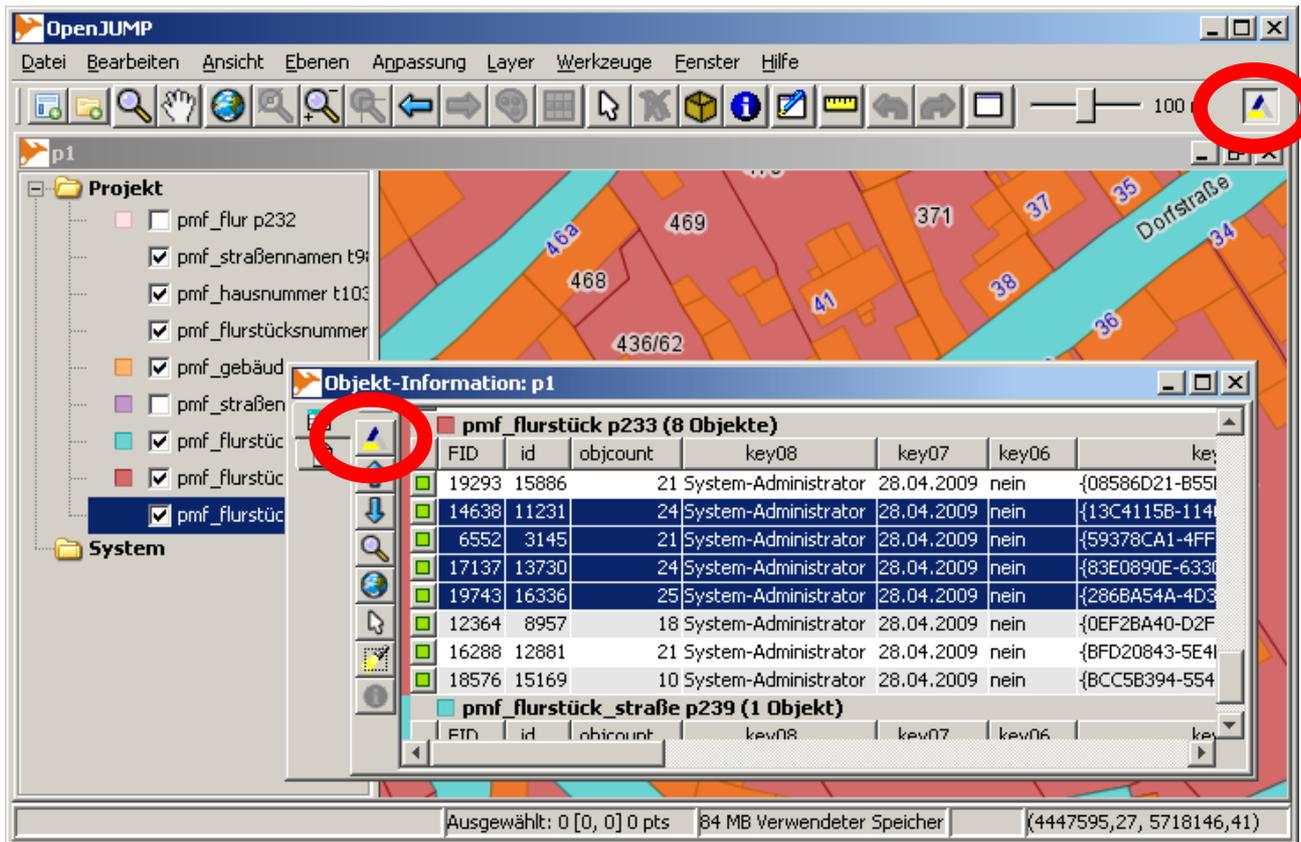
Seit Version 0.2 ist es außerdem möglich, im Dateinamen für die Austauschdatei Umgebungsvariablen zu nutzen. Hintergrund ist, dass Archikart die Datei im TEMP Verzeichnis ablegt. Leider ist es z.B. unter Windows 2003 Terminalserver so, dass der Pfad zum TEMP Verzeichnis bei jeder Anmeldung anders ist. Daher ist es besser, man gibt „%TEMP%\ak2open.asc“ an. „%TEMP%“ wird dann durch den Inhalt der Umgebungsvariablen ersetzt. Der reale Pfad sieht dann z.B. so aus „C:\Dokumente und Einstellungen\mscholz\Lokale Einstellungen\Temp\5\ak2open.asc“.

## Anwendung

Möchte man zu einem Objekt die Daten in Archikart anzeigen, clickt man in der Werkzeugleiste auf den Archikart Button, welches sich ganz rechts befinden sollte. Bei dem sich öffnenden Dialog handelt es sich im wesentlichen um einen Objektinfo Dialog, der um einen Button erweitert wurde. Dort befindet sich ebenfalls ein Archikart Button. Der Button wird aktiv, wenn man mindestens ein Objekt markiert hat. Durch den Button wird dann die Austauschdatei erzeugt und die Kommunikationsanwendung aufgerufen. Es sollte dann im Archikart das entsprechende Modul geöffnet werden. In der korrespondierenden Findenliste stehen dann die zuvor im OpenJUMP ausgewählten Objekte.

Will man vom Archikart aus ein Objekt im OpenJUMP anzeigen, geht man den üblichen Weg über die Findenliste->GIS->wechseln zur GIS Kopplung. Zu beachten ist noch, dass vorher sämtliche Objektmarkierungen im OpenJUMP aufgehoben werden, bevor die übergebenen Objekte markiert und herangezoomt werden.

Im Menü Werkzeuge gibt es noch einen neuen Menüpunkt. Je nach Status des Kommunikationsprozesses heißt der Menüpunkt „CommThread Starten“, wenn er gerade nicht läuft oder „CommThread Stoppen“, wenn er gerade läuft. Wenn er gestoppt ist, wird nicht mehr auf eingehende Daten für OpenJUMP geachtet. Der Kommunikationsprozess wird standardmäßig beim Start von OpenJUMP gestartet.



## Übernahme von Objektschlüsseln in OpenJUMP

Möchte man für neu gezeichnete Objekte den Objektschlüssel aus Archikart übernehmen, dann ist in der Konfiguration für das CommPlugIn die Option „Objektschlüssel übernehmen“ zu aktivieren. Dadurch verändert sich das Verhalten bei der Übergabe von Objektschlüsseln an das Plugin. Wenn die Übernahme des Objektschlüssels aktiv ist, dann wird geprüft, ob genau eine Geometrie markiert ist, genau ein Objektschlüssel übergeben wurde und die Attributspalte für die Geometrie vorhanden ist. Sind diese Bedingungen erfüllt, dann kann der Objektschlüssel für die Geometrie übernommen werden. Vor der Übernahme wird noch abgefragt, ob der Schlüssel tatsächlich übernommen werden soll. Es wird der Vorhandene überschrieben. Ebenso wird nicht geprüft, ob der Schlüssel schon für eine andere Geometrie vergeben ist. Das dürfte man aber sehr schnell mit einer neuen Suche bemerken, die dann mehr als eine Geometrie findet.