OSGeo Journal

The Journal of the Open Source Geospatial Foundation

Volume 3 / December 2007

2007 FREE AND OPEN SOURCE SOFTWARE
FOR GEOSPATIAL (FOSS4G) CONFERENCE
VICTORIA CANADA SEPTEMBER 24 TO 27, 2007

Proceedings of FOSS4G 2007

Integration & Development

Topical Interest

Case Studies

GeoSIPAM

Free and Open Source Software Applied to the Protection of the Brazilian Amazon

Luis Fernando Bueno, Weberson Gabriel, Pablo Filetti Moreira, Artur Henrique Villa Real F. Ramos, Fabio Augusto das Dores Silva, Marcelo Campos Brito

Keywords: *Spatial Data, Brazilian Amazon, SIPAM, Infrastructure, GeoSIPAM*

Abstract

This paper presents the infrastructure implemented and maintained by the System of Protection of Amazon - SIPAM for storage and distribution of spatial data called GeoSIPAM. The use of free and open source applications for the storage and distribution of the spatial data was chosen. The relational database PostgreSQL was chosen, with the PostGIS extension for spatial data storage and manipulation. Several applications (GeoServer, GeoNetwork and InterMap) have been customized for use by SIPAM. The implemented customizations involved procedure development which enhanced the tools used in daily tasks played by the collaborators. The application interface of the GeoNetwork and InterMap were altered aiming at making them accessible and adding features. Care was taken that the infrastructure was adjusted to spatial data storage, manipulation and distribution. The customizations allowed the applications to be merged into SIPAMś organizational environment. It was also verified that the use of OGC standards were complied with for interoperability on diverse systems.

Introduction

The database of the System of Protection of Amazon - SIPAM integrates information brought up to date on the Brazilian Legal Amazon. The use of these information in projects developed for the SIPAM and agency partners provides the generation of knowledge that assists the planning and coordination of global actions of government, aiming at the protection, social inclusion and sustainable development of the region.

This work presents the infrastructure implemented and maintained for SIPAM for storage and distribution of spatial data called GeoSIPAM. GeoSIPAM aims at providing integration and evaluation of data to aid the planning and the coordination of the actions of the developed public politics in the Brazilian Legal Amazon. Specifically, the objectives of GeoSIPAM are: to display through Internet or Intranet the metadata referring to the projects carried through for SIPAM; to make available Internet maps, geocoded images and related literal information; to visualize maps, images and relevant information for other institutions, through the use of standard compliant Open Geospatial Consortium - OGC (1) protocols

Free and Open Source Software in GeoSIPAM

The development and use of free software for geoinformatics has made available an increasing number of software tools. In the designing of GeoSIPAM it was opted to use free and open source software for the storage and distribution of the spatial data. The tools chosen include:

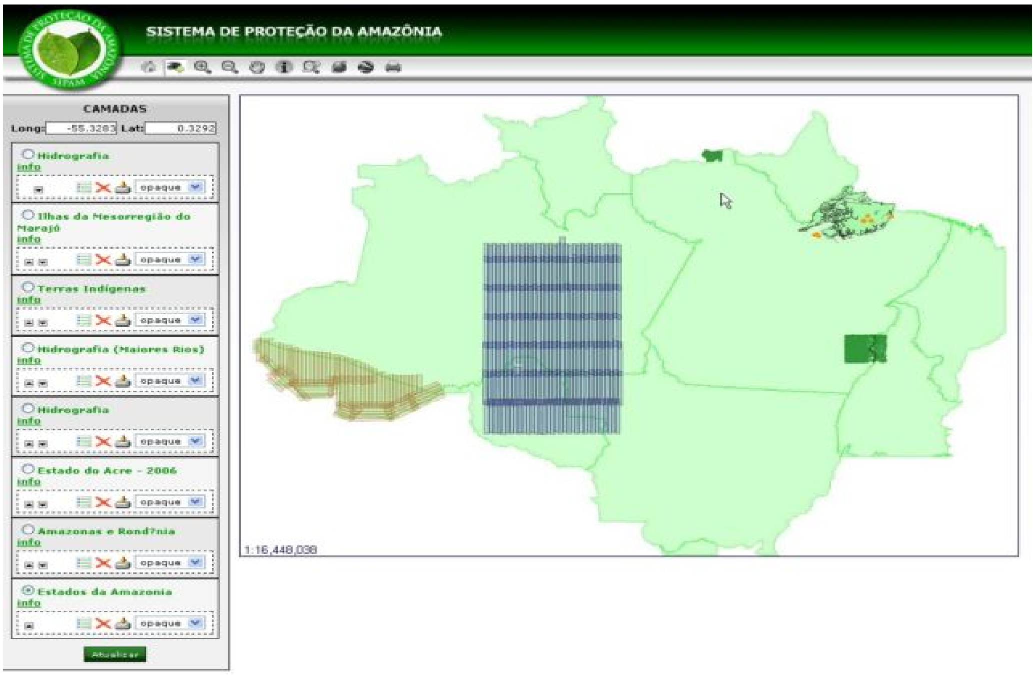
- PostgreSQL Object-relational database management system
- PostGIS Spatial database extension for PostgreSQL
- GeoServer OpenGIS Transactional Web Feature Server
- GeoNetwork Catalog application to manage spatially referenced resources through the web
- InterMap Map viewer, generally configured to operate of form integrated to the GeoNetwork

Infrastructure for Spatial Analysis

The chosen software architecture was organized in layers involving

- a Database Management System (DBMS) for data storage, management and manipulation
- maps and catalogues servers for data distribution and information
- as well as interfaces for data access based on the standards established for the OGC.

The PostgreSQL DBMS was chosen, with the PostGIS extension for storage and manipulation of



Long: -55.3283 Lat: 0.3292

Figure 1: Interactive map visualization, usage of the customized InterMap software interface

spatial data. The GeoNetwork and InterMap applications interface had undergone alterations aiming to make them accessible and unique.

Figure 1 presents the visualization of an interactive map in the customized interface of InterMap software.

The GeoServer, GeoNetwork and InterMap, applications are tools developed in Java and implement the Web Map Services (WMS), Web Feature Services (WFS) and Catalogue Service Web (CSW) standards. These were customized for use at SIPAM. The implemented customizations involved development of procedures that enabled collaborators to carry out daily played tasks. Following are some customizations implemented by the GeoSIPAM team.

Originally InterMap Open Source used to attend the requirements of the standard WMS / OGC, but did not attend to the standard WFS / OGC, both offered by GeoServer. In InterMap Open Source a service was created to carry through WFS service solicitations to GeoServer, making it possible to download raster files. In this way, an option for the WFS GetFeature service call was developed, in the SHAPE - ZIP format and the service services.map.DownloadService was implemented. Also a corresponding button in the layer toolbar was created, as demonstrated in figure 2. This functionality was implemented by source code changes, involving JavaScript, XSL and Java programming.

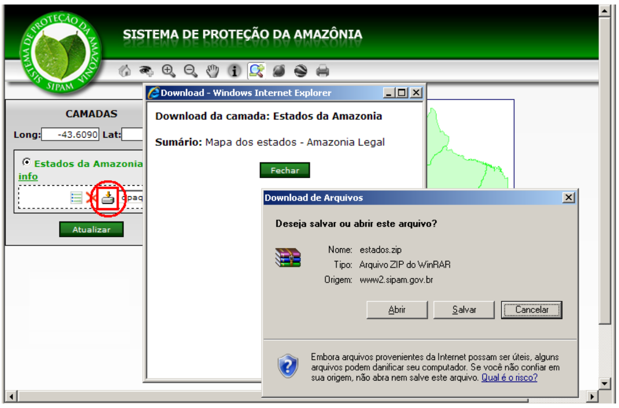
```
public Element exec(Element params,
      ServiceContext context) throws Exception
{
// Get request parameters
int id = Integer.parseInt(
    params.getChildText(
        Constants.MAP_SERVICE_ID
    ));

// Get the MapMerger object from the user session
MapMerger mm = MapUtil.getMapMerger(context);

// Get the layer name
String nome = mm.getService(id).getName();
// Get the WMS server url
String serverUrl = mm.getService(id).getServerURL();

// Declare and set the WFS server url
int pos = serverUrl.indexOf("/wms")+1;
String WfsFeatureInfoUrl = serverUrl.substring(0, pos) +
Element el = WmsGetCapClient.getCapabilities(serverUrl);

// Return the necessary information for
// the service execution
return new Element("response")
.addContent(WmsGetCapClient.getCapabilities(serverUrl))
.addContent(new Element("serviceName").setText(nome))
.addContent(
  new Element(
    Constants.URL).setText(
      WfsFeatureInfoUrl));
}
```



Download da camada: Estados da Amazonia

Sumário: Mapa dos estados - Amazonia Legal

Deseja salvar ou abrir este arquivo?

Nome: estados.zip
Tipo: Arquivo ZIP do WinRAR
Origem: www2.sipam.gov.br

Embora arquivos provenientes da Internet possam ser úteis, alguns arquivos podem danificar seu computador. Se você não confiar em sua origem, não abra nem salve este arquivo. Qual é o risco?

Figure 2: Customized InterMap interface for the download of layers

The option to offer the layer for download is configured through service WFS. The WFS GetFeature service allows download indiscriminately all the available layers in the server. The option was created to allow/deny the layer download in the cadastre screen of the FeatureType in the GeoServer. This information is also sent to WMS customer through the service getCapabilities. The Download button will be shown in the InterMap whenever the corresponding option is qualified in the GeoServer, as demonstrated in figure 3. The functionality was implemented through changes in the GeoServer source code, which is written in Java.

Another implemented functionality allows the visualization of the data together with images provided by Google Earth API. An inserted button in the InterMap toolbar sets in motion a floating window that shows the spatial data, synchronizing the GeoServerś raster data with the Googleś API images. The position and zoom are kept synchronized even after changes in the Google API window or InterMap maps‘ window. The InterMap layers are displayed in the Googleś API window in transparent shades, allowing simultaneous viewing of Googleś images. Figure 4 presents the Google API window implemented in InterMap. This functionality was implemented by the inclusion of JavaScript code in the file im_main.xsl. The code line below shows the reference to the Google API, in the file im_main.xsl:

```
<script
 src="http://maps.google.com/maps? \
    file=api&v=2&key=GOOGLE_API_KEY"
 type="text/javascript">
</script>
```

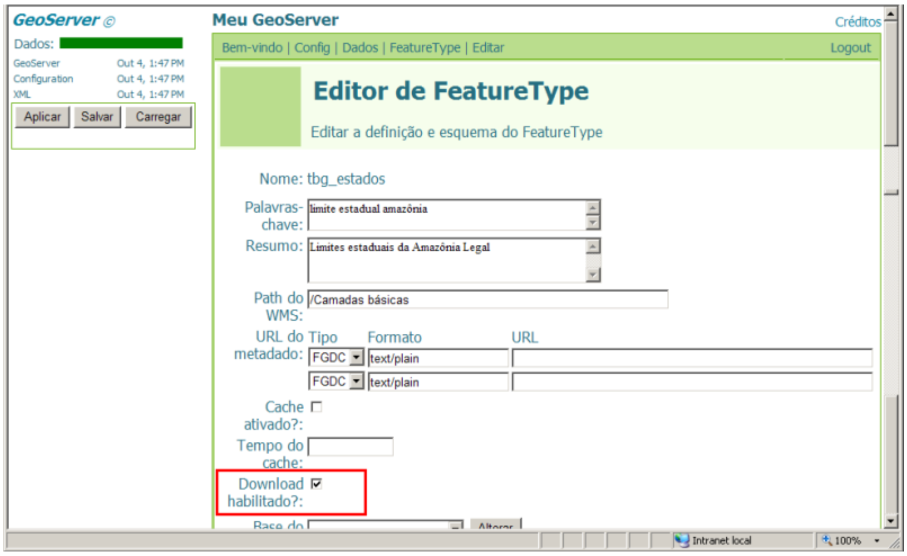
The term GOOGLE_API_KEY should be replaced by the key obtained on the Google API site[24]. To obtain the key you should access the site and provide the server URL where InterMap is installed. The key is sent to the email address of the applicant, who must have a GMail account. The functions “addWMS” and “montaMapa” were coded and added to the file im_main.xsl. The function “addWMS” is used to rebuild the list of layers to show the Google API, always when a new layer is added in InterMap. The function “montaMapa”, is used to construct the Google map. Below is shown the source code of the functions “addWMS” and “montaMapa”.

```
// Add WMS servers and layers types
function addWMS(servidor, camada){
 achou = false;
//Check if the server exists.
// If it does, add a new layer.
// If it does not register a new
// server and layer
for (contador = 0; wmsurl.length>contador; contador++){
if( wmsurl[contador] == servidor) {
// server found.. add layer
layersG[contador] = layersG[contador] + ',' + camada
achou = true;
}
}
if(!achou) {
// server not found, register in the vector
pos = wmsurl.length;
wmsurl[pos] = servidor;
layersG[pos] = camada;
}
listalayers = listalayers + camada
}

// Set up the google map on the screen
function montaMapa() {
var camadaWMS = new Array();
if( visivelG ) {
mapG = new GMap2(document.getElementById("mapaG"));

// A handler is called when a
// map movement break event is called
GEvent.addListener(mapG, "moveend", function() {
// If the movement was done at the Google map,
// the Intermap map changes too
if( actOrigem == '' ){
var center = mapG.getCenter();
// Obtain the center
main.map.zMapCenLat = center.lat();
// in LatLong
```

[24] Google API site: http://www.google.com/apis/maps/



Editor de FeatureType

Editar a definição e esquema do FeatureType

Nome: tbg_estados

Palavras-chave: limite estadual amazônia

Resumo: Limites estaduais da Amazônia Legal

Path do WMS: /Camadas básicas

URL do metadado:	Tipo	Formato	URL
	FGDC	text/plain	
	FGDC	text/plain	

Cache ativado?: □

Tempo do cache:

Download habilitado?: ☑

Base do … Alterar

Figure 3: Customized GeoServer interface for enabling download

```
main.map.zMapCenLng = center.lng();
main.map.moveIntermap();
// moves the Intermap map
}
actOrigem = '';
if(navm){setTimeout("pngfix()",1000); }
});
// Handler is called
// when a drag action event is called
GEvent.addListener(mapG, "dragstart", function() {
actOrigem=''
});

// Handler is called
// when a zoom event is called
GEvent.addListener(mapG, "zoomend",  \
    function(zoomant, zoomatu) {
if( actOrigem == '') {
var bbox = mapG.getBounds();
// Obtain the center
main.map.zmapminlat = bbox.getSouthWest().lat();
// em latitude e longitude
main.map.zmapminlng = bbox.getSouthWest().lng();
// e move o mapa
main.map.zmapmaxlat = bbox.getNorthEast().lat();
main.map.zmapmaxlng = bbox.getNorthEast().lng();
main.map.zoomIntermap(zoomant, zoomatu);
// no intermap
}
if (actOrigem != 'X'){actOrigem = '';}
});

// Handler is called
// when a drag action event is called
```

```
var layer1=[G_SATELLITE_MAP.getTileLayers()[0], \
    G_HYBRID_MAP.getTileLayers()[1]];
var custommap1 = new GMapType(layer1, \
    G_SATELLITE_MAP.getProjection(), \
    "Google", G_SATELLITE_MAP);

var layer3=[G_SATELLITE_MAP.getTileLayers()[0], \
    G_HYBRID_MAP.getTileLayers()[1]];
for( contador=0; wmsurl.length > contador; \
contador++){
// create tile layers
camadaWMS[contador]= new GTileLayer( \
    new GCopyrightCollection(""),1,17);
if( wmsurl[contador] != '' )
{ if( layersG[contador] != '') {
camadaWMS[contador].myLayers=layersG[contador];
camadaWMS[contador].myBaseURL=wmsurl[contador];
camadaWMS[contador].getTileUrl=CustomGetTileUrl;
if(navm) {camadaWMS[contador].myFormat='image/png'}
} }
layer3[contador+2] = camadaWMS[contador];
}

var custommap3 = new GMapType(layer3, \
    G_SATELLITE_MAP.getProjection(), \
    "Intermap", G_SATELLITE_MAP);

mapG.getMapTypes().length = 0;
mapG.addMapType(custommap3);
mapG.addMapType(custommap1);
SWLatLng = new GLatLng(SWLat, SWLng)
NELatLng = new GLatLng(NELat, NELng)
bbox     = new GLatLngBounds(SWLatLng, NELatLng)
zoom     = mapG.getBoundsZoomLevel(bbox)
gmapt = mapG.getMapTypes()
```

```
zoomMax = \
    gmapt[1].getMaximumResolution(mapG.getCenter())
if( zoom > zoomMax) {zoom = zoomMax;}
actOrigem = 'X'
mapG.setCenter(bbox.getCenter(),zoom);
mapG.addControl(new GLargeMapControl());
mapG.addControl(new GMapTypeControl());
with(document.getElementById( \
    "google").style){top = "113px";left = "239px";}
actOrigem = '' } }
```

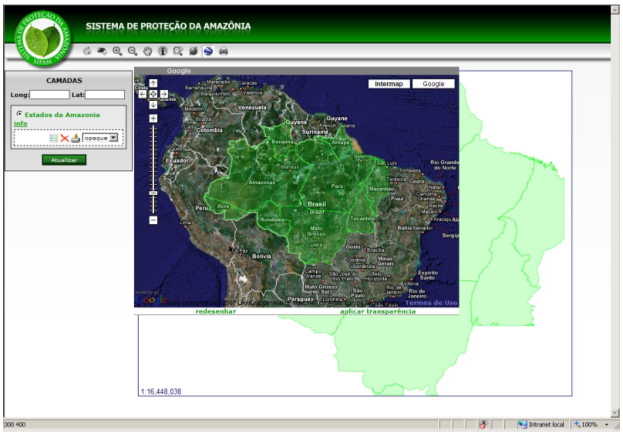


Figure 4: Customized InterMap software interface with Google API

Because of the great number of layers available in WMS servers, the list layers visualization, when shown as a flat list, becomes inadequate. The display of layers was modified to be loaded in tree form with options of "collapse" and "expand". The functionality was implemented by the inclusion of JavaScript code in the file im_get_services.xsl. It employed the dTree API, a free JavaScript tree menu. dTree JavaScript can be obtained online.[25]. In this case, the strategy adopted was entering the WMS server layers in an array. Thus, the layers tree is mounted by dTree from this array. Figure 5 demonstrates the layers visualization organized in tree form.

Results

An infrastructure was chosen and implemented that insured the capabilities for spatial data storage, manipulation and distribution. The database was populated with spatial data regarding the Legal Brazilian Amazonia, including topics such as hydrography, geology, use and occupation of land, mineral resources, administrative boundaries, roads, soil, etc. Spatial data from SIPAMś other partners, such as the National Agency of Water (ANA) and Brazilian Geological Service (CPRM), Brazilian Institute of Geography and Statistics, IBGE was also stored in GeoSIPAM. The volume of data stored grows every day. Local customizations to available FOSS GIS packages allowed implementing some features into SIPAM which were particular to organizational environment. It was also verified that the use of OGC standards ensures interoperability with diverse systems. The spatial data and products elaborated for the operational team of the SIPAM are being registered in a cadastre and stored in the GeoSIPAM system. This information is being distributed in the internal net and also for the public in general, through the SIPAMś web portal over the Internet. Among the main partners which constantly access the site are the Environment Secretaries of the State Governments of Amazonia, the Environmental Police, regional universities and others. Implemented statistics controls have logged more than 100,000 hits to GeoSIPAM.

Bibliography

[1] Bueno, L.F., P.G. Zuza e W. Gabriel. GeoSIPAM: Manual do Usuário. Presidência da República, CENSIPAM, CTO/Pv, 2007, 52 páginas.

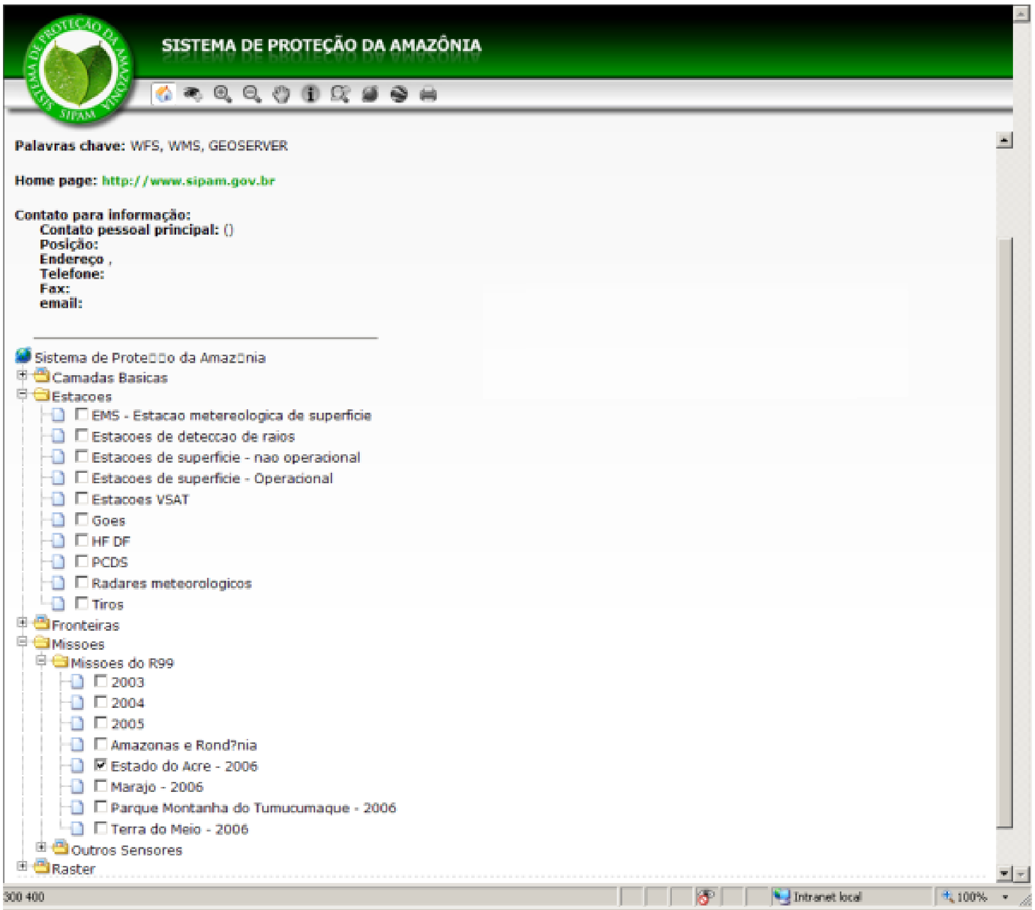
[2] Câmara G.; C. Davis, e A.M.V. Monteiro. Introdução à Ciência da Geoinformação. (online). 2001. http://www.dpi.inpe.br/gilberto/livro/introd/. 14 April 2007

[3] Garnett, J. e C. Holmes. What is Geoserver. (online). 2007. http://docs.codehaus.org/display/GEOS/Home 10 May 2007.

[4] GeoNetwork opensource Community website. (online). 2007. http://geonetwork-opensource.org/ 19 March 2007.

[5] Landrö, Geir. dTree. (online). 2003. http://www.destroydrop.com/javascripts/tree/ 05 Setembro 2007

[25] dTree JavaScript API: http://www.destroydrop.com/javascripts/tree/


SISTEMA DE PROTEÇÃO DA AMAZÔNIA
SISTEMA DE PROTEÇÃO DA AMAZÔNIA SIPAM
300 400
Intranet local
100%

Palavras chave: WFS, WMS, GEOSERVER

Home page: http://www.sipam.gov.br

Contato para informação:
Contato pessoal principal: ()
Posição:
Endereço ,
Telefone:
Fax:
email:

Sistema de Prote□□o da Amaz□nia
⊞ Camadas Basicas
⊟ Estacoes
- □ EMS - Estacao metereologica de superficie
- □ Estacoes de deteccao de raios
- □ Estacoes de superficie - nao operacional
- □ Estacoes de superficie - Operacional
- □ Estacoes VSAT
- □ Goes
- □ HF DF
- □ PCDS
- □ Radares meteorologicos
- □ Tiros

⊞ Fronteiras
⊟ Missoes
⊟ Missoes do R99
- □ 2003
- □ 2004
- □ 2005
- □ Amazonas e Rond?nia
- ☑ Estado do Acre - 2006
- □ Marajo - 2006
- □ Parque Montanha do Tumucumaque - 2006
- □ Terra do Meio - 2006

⊞ Outros Sensores
⊞ Raster

Figure 5: Customized InterMap interface with list layers vizualization

[6] PostgreSQL Global Development Group. PostgreSQL 8.1.0 Documentation. (online). 2005. `http://www.postgresql.org/docs/manuals/` 05 May 2007

[7] Refractions Research. PostGIS Manual. (online). 2005. `http://postgis.refractions.net/documentation/` 27 March 2007

[8] Wheeler, D.A. Why Open Source Software/Free Software (OSS/FS, FLOSS, or FOSS)? Look at the Numbers! (on line) `http://www.dwheeler.com/oss_fs_why.html` 03 June 2007

Luis Fernando Bueno
System of Protection of Amazon - SIPAM
SPO Área 05, Quadra 03, Bloco "K" Brasília - DF
CEP 70610 - 200
`luis.bueno AT sipam.gov.br`

Weberson Gabriel
`weberson.gabriel AT sipam.gov.br`

Pablo Filetti Moreira
`pablo.moreira AT sipam.gov.br`

Artur Henrique Villa Real F. Ramos
`artur.ramos AT sipam.gov.br`

Fabio Augusto das Dores Silva
`fabio.silva AT sipam.gov.br`

Marcelo Campos Brito
`marcelo.brito AT sipam.gov.br`

The Open Source Geospatial Foundation, or OSGeo, is a not-for-profit organization whose mission is to support and promote the collaborative development of open geospatial technologies and data. The foundation provides financial, organizational and legal support to the broader open source geospatial community. It also serves as an independent legal entity to which community members can contribute code, funding and other resources, secure in the knowledge that their contributions will be maintained for public benefit. OSGeo also serves as an outreach and advocacy organization for the open source geospatial community, and provides a common forum and shared infrastructure for improving cross-project collaboration.

Published by OSGeo, the OSGeo Journal is focused on presenting discussion papers, case studies and introductions and concepts relating to open source and geospatial software topics.

Acknowledgements
Various reviewers & the GRASS News Project

The *OSGeo Journal* is a publication of the *OSGeo Foundation*. The base of this journal, the $\LaTeX\,2_\varepsilon$style source has been kindly provided by the GRASS and R News editorial board.

OSGeo
Your Open Source Compass

Journal online: `http://www.osgeo.org/journal`
OSGeo Homepage: `http://www.osgeo.org`
Postal mail: OSGeo
PO Box 4844, Williams Lake,
British Columbia, Canada, V2G 2V8
Telephone: +1-250-277-1621

9 771994 189002 00

ISSN 1994-1897