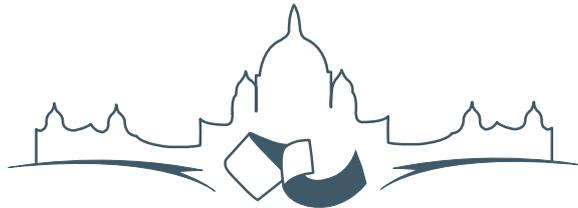

Journal de l'OSGeo

Le journal de la Fondation Open Source Geospatial

Volume 3 / Décembre 2007



2007 FREE AND OPEN SOURCE SOFTWARE
FOR GEOSPATIAL (FOS4G) CONFERENCE
VICTORIA CANADA  SEPTEMBER 24 TO 27, 2007

Compte rendu du FOS4G 2007

Intégration & Développement

- Portable GIS : SIG sur une clé USB
- Génération Automatique d'Applications SIG / Base de données sur Internet
- db4o2D - Extension de Base de données Orientée Objet pour les types géospatiaux 2D
- Google Summer of Code pour la géomatique

Intérêt thématique

- Approche générique pour la gestion de standards de métadonnées
- Vers des services web dédiés à la cartographie thématique
- Interopérabilité pour les données géospatiales 3D
- Un Service Web orienté modèle pour une interopérabilité sémantique améliorée
- Spatial-Yap : un système de base de données spatialement déductif

Études de cas

- Le Projet DIVERT : Développement de Télématiques Inter-Véhicules Fiables
 - GRASS et la Modélisation des Risques Naturels
 - Une Base de Données Spatiales pour l'Intégration des Données du Projet de Gestion des Ressources Naturelles du Rondonia
 - GeoSIPAM : Logiciel libre et Open Source appliqué à la protection de l'Amazonie brésilienne
 - Le Système de Suivi de la Déforestation Amazonienne
-

GeoSIPAM

Logiciel libre et Open Source appliqué à la protection de l'Amazonie brésilienne

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

Keywords : *Données spatiales, Amazonie brésilienne, SIPAM, Infrastructure, GeoSIPAM*

Résumé

Cet article présente l'infrastructure GeoSIPAM implémentée et maintenue par le Système de Protection de l'Amazonie - SIPAM pour le stockage et la diffusion de données spatiales. Des applications open source libres (FOSS) ont été utilisées pour le stockage et la diffusion des données spatiales. Le système de gestion de base de données relationnelle PostgreSQL a été choisi, avec son extension PostGIS pour le stockage et la manipulation des données spatiales. Plusieurs applications (GeoServer, GeoNetwork, InterMap) ont été adaptées pour le SIPAM. Les adaptations effectuées ont concerné des procédures de développement améliorant ces outils utilisés quotidiennement par les collaborateurs. Les interfaces applicatives de GeoNetwork et d'InterMap ont été modifiées afin de les rendre accessibles et d'y ajouter des éléments. Un soin particulier a été apporté pour que l'infrastructure soit adaptée au stockage, à la manipulation et à la diffusion des données spatiales. La personnalisation a permis l'intégration des applications à l'environnement organisationnel du SIPAM. Il a aussi été vérifié que l'utilisation des standards OGC était correcte afin que les systèmes soient interopérables.

Introduction

La base de données du Système de Protection de l'Amazonie - SIPAM intègre des informations de l'Amazonie Brésilienne Légale (BLA) à jour. L'utilisation de ces informations dans les projets développés par le SIPAM et par les agences partenaires fournit de la connaissance qui facilite la planification et la coordination des actions gouvernementales globales, visant à la protection, l'intégration sociale et le développement durable de la région.

Ce travail présente l'infrastructure GeoSIPAM implémentée et maintenue par le SIPAM pour le stockage et la diffusion des données spatiales. GeoSIPAM vise à offrir l'intégration et l'évaluation des données afin d'aider à la planification et à la coordination des actions des politiques publiques de développement en Amazonie Brésilienne Légale. En particulier, les objectifs de GeoSIPAM sont les suivants : afficher sur Internet ou Intranet les métadonnées des projets portés par le SIPAM ; mettre à disposition des cartes, des images géocodées et de l'information textuelle associée sur Internet ; visualiser des cartes, des images, et des informations pertinentes pour d'autres institutions, à travers l'utilisation des protocoles standards OGC (1).

Les FOSS dans GeoSIPAM

Le développement et l'utilisation de logiciels libres en géomatique ont mis à disposition un nombre croissant d'outils. Lors de la conception de GeoSIPAM, il a été décidé d'utiliser des logiciels open source libres (FOSS) pour le stockage et la diffusion des données spatiales. Les outils choisis sont les suivants :

- PostgreSQL Système de gestion de base de données relationnelle
- PostGIS Extension spatiale pour PostgreSQL
- GeoServer OpenGIS Web Feature Server Transactionnel
- GeoNetwork Catalogue pour gérer les ressources géoréférencées sur le web
- InterMap Map viewer, généralement configuré comme étant intégré à GeoNetwork

Infrastructure pour l'Analyse Spatiale

L'architecture logicielle choisie est organisée en couches impliquant

- un système de gestion de base de données (SGBD) pour le stockage, la gestion et la manipulation des données
- des cartes et des serveurs de catalogues pour la diffusion et l'information sur les données
- autant que des interfaces pour l'accès aux données basé sur les standards établis par l'OGC.

Le SGBD PostgreSQL a été choisi, ainsi que son extension PostGIS pour le stockage et la manipulation des données spatiales. Les interfaces des applications GeoNetwork et InterMap ont subi des modifications pour les rendre accessibles et uniques.

La figure 1 présente la visualisation de la carte interactive au sein de l'interface personnalisée d'InterMap.

Les applications GeoServer, GeoNetwork et InterMap sont des outils développés en Java qui implémentent les standards Web Map Services (WMS), Web Feature Services (WFS) et Catalogue Service Web (CSW). Elles ont aussi été personnalisées pour une utilisation au SIPAM. Les personnalisations effectuées ont concerné des procédures de développement qui ont permis aux collaborateurs de réaliser leurs tâches quotidiennes. Quelques personnalisations réalisées par l'équipe GeoSIPAM sont présentées par la suite.

Initialement, InterMap répondait aux exigences du standard WMS, mais pas au standard WFS, les deux étant proposés par GeoServer. Dans InterMap un service a été créé pour transporter des demandes vers GeoServer à travers un service WFS, rendant ainsi possible le téléchargement de fichiers raster. Ainsi une option pour l'appel du service GetFeature de WFS a été développée dans le format SHAPE - ZIP et le service services.map.DownloadService a été implémenté. Un bouton correspondant a été ajouté dans la barre d'outils de la couche, ainsi que la figure 2 le montre. Cette fonctionnalité a été implémentée par des modifications du code source, impliquant de la programmation en JavaScript, en XSL et en Java.

```
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)
        + "wfs?request=GetFeature&service=WFS&version=1.0";
}
```

```
+ nome;
Element e1 = 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));
}
```

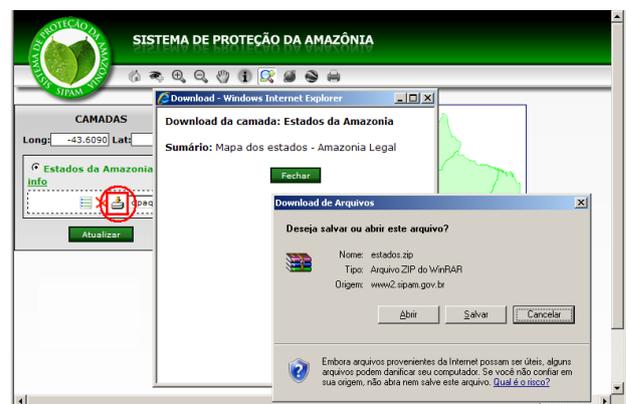


FIG. 2: Interface InterMap Personnalisée pour le téléchargement de couches

L'option qui permet de proposer une couche à télécharger est réalisée par un service WFS. Le GetFeature de WFS permet de télécharger indifféremment toutes les couches disponibles sur le serveur. L'option a été créée pour permettre/empêcher le téléchargement dans l'écran du FeatureType de GeoServer. Cette information est aussi envoyée au demandeur WMS par le service getCapabilities. Le bouton Téléchargement sera montré dans InterMap dès que l'option correspondante sera qualifiée dans GeoServer, comme présenté à la figure 3. La fonctionnalité a été implémentée grâce à des modifications du code source de GeoServer écrit en Java.

Une autre fonctionnalité implémentée permet la visualisation de données en même temps que données fournies par l'API Google Earth. Un bouton inséré dans la barre d'outils d'InterMap crée une fenêtre flottante en mouvement qui montre les données spatiales, synchronisant les données raster de GeoServer avec les images de l'API de Google Earth. La position et le zoom sont conservés même après des modifications dans la visualisation de l'API Google Earth. La figure 4 présente la fenêtre Google Earth implémentée dans InterMap. Cette

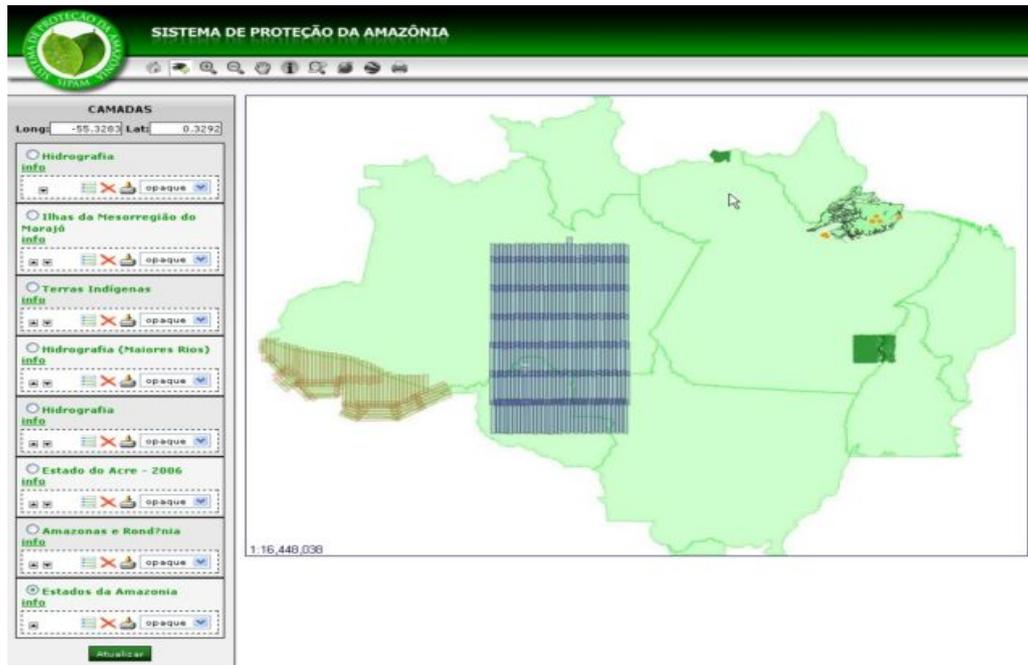


FIG. 1: Visualisation de la carte interactive, utilisation de l'interface personnalisée d'InterMap

fonctionnalité a été implémentée grâce à l'inclusion de code Javascript dans le fichier `im_main.xml`. La ligne de code ci-dessous montre la référence à l'API Google, dans le fichier `im_main.xml`.

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

L'expression `GOOGLE_API_KEY` doit être remplacée par la clé obtenue sur le site de l'API Google²². Pour obtenir la clé, vous devez accéder au site et fournir l'URL du serveur sur lequel InterMap est installé. La clé est envoyée sur l'adresse email du demandeur, qui doit avoir un compte GMail. Les fonctions "addWMS" et "montaMapa" ont été codées et ajoutées au fichier `im_main.xml`. La fonction "addWMS" est utilisée pour reconstruire la liste des couches à montrer à l'API Google, à chaque fois qu'une nouvelle couche est ajoutée dans InterMap. La fonction "montaMapa" est utilisée pour construire la carte Google. Le code source des fonctions "addWMS" et "montaMapa" est présenté ci-dessous.

```
// Add WMS servers and layers types
```

²²Google API site : <http://www.google.com/apis/maps/>

```
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
```

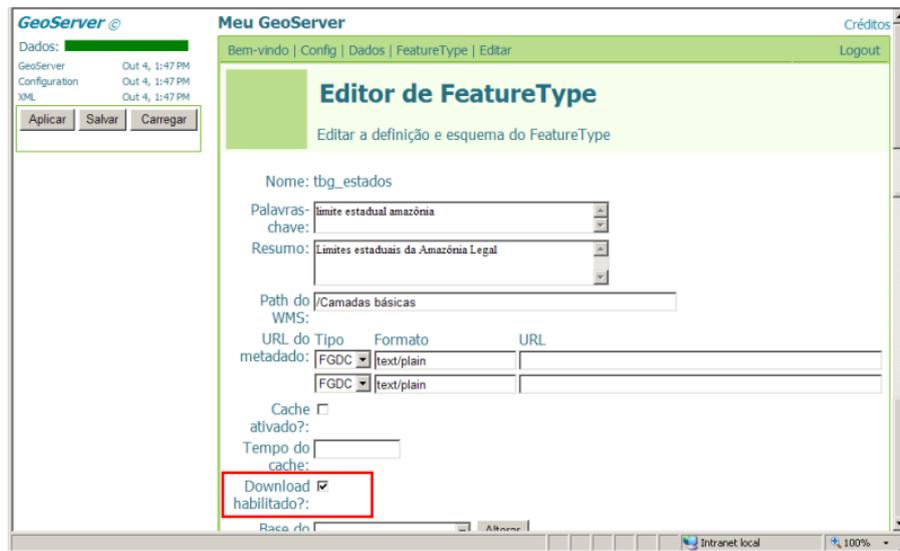


FIG. 3: Interface de Geoserver personnalisée permettant le téléchargement

```
// 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
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 = ' ' } }

```

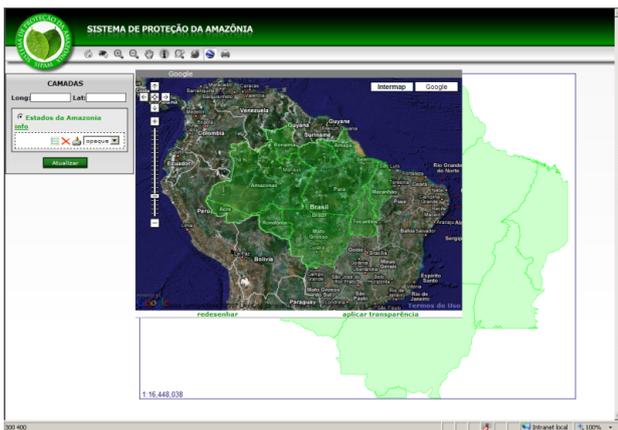


FIG. 4: Interface InterMap personnalisée avec l'API Google

A cause du grand nombre de couches disponibles sur les serveurs WMS, la visualisation de la liste des couches sous forme d'une liste déroulante devient inadéquate. L'affichage des couches a été modifié afin d'être chargé sous forme d'un arbre avec les options "refermer" et "explorer". La fonctionnalité a été implémentée par l'ajout de code Javascript dans le fichier `im_get_services.xml`. L'API dTree, un arbre Javascript libre, a été utilisée. dTree peut être récupéré en ligne.²³ Dans ce cas, la stratégie adoptée a été de mettre les couches des serveurs WMS dans un tableau. dTree crée ensuite l'arbre des couches à partir de ce tableau. Figure 5 présente la visualisation des couches sous forme d'un arbre.

²³dTree JavaScript API : <http://www.destroydrop.com/javascripts/tree/>

Résultats

Une infrastructure ayant les capacités de stockage, de manipulation et de diffusion des données spatiales a été choisie et implémentée. La base de données a été remplie avec des données spatiales concernant l'Amazonie Brésilienne Légale, incluant des sujets tels que l'hydrographie, la géologie, l'utilisation et l'occupation du sol, les ressources minérales, les limites administratives, les routes, les sols, etc. Des données spatiales venant des autres partenaires du SIPAM, comme l'Agence Nationale de l'Eau (ANA), le Service Géologique Brésilien (CPRM), l'Institut Brésilien de Géographie et de Statistiques, IBGE ont aussi été intégrées dans GeoSIPAM. Le volume des données stockées grandit chaque jour. Les personnalisations locales des packages SIG FOSS disponibles ont permis d'implémenter des éléments du SIPAM qui sont spécifiques à l'environnement organisationnel. Il a aussi été vérifié que l'utilisation des normes OGC respecte l'interopérabilité des divers systèmes. Les données spatiales et les produits élaborés pour l'équipe opérationnelle du SIPAM ont été enregistrés dans le cadastre et stockés dans le système GeoSIPAM. Cette information est en cours de diffusion sur le réseau interne mais aussi de façon public en général, à travers le portail web du SIPAM sur Internet. En plus des principaux partenaires qui accèdent constamment au site, on peut trouver les Secrétariats à l'Environnement des Gouvernements d'Etat de l'Amazonie, la Police Environnementale, des universités régionales et d'autres encore. Des contrôles statistiques

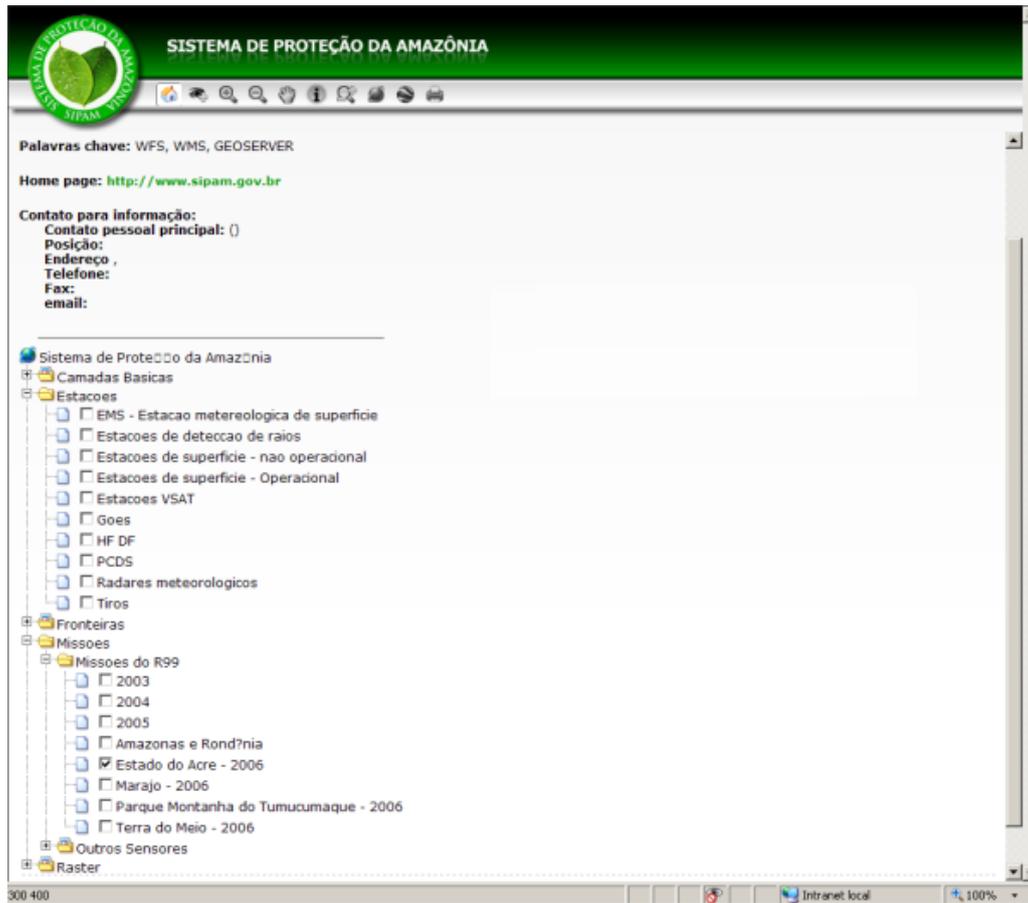


FIG. 5: Interface InterMap personnalisée avec la visualisation de la liste des couches

ont montré plus de 100,000 connexions à GeoSIPAM.

Bibliographie

[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.destrodrop.com/javascripts/tree/> 05 Setembro 2007
- [6] PostgreSQL Global Development Group. PostgreSQL 8.1.0 Documentation. (online). 2005. <http://www.postgresql.org/docs/manuals/> 05 May 2007
- [7] Refrations Research. PostGIS Manual. (online). 2005. <http://postgis.refrations.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](mailto:luis.bueno@sipam.gov.br)

Weberson Gabriel
[weberson.gabriel AT sipam.gov.br](mailto:weberson.gabriel@sipam.gov.br)

Pablo Filetti Moreira
[pablo.moreira AT sipam.gov.br](mailto:pablo.moreira@sipam.gov.br)

Artur Henrique Villa Real F. Ramos
[artur.amos AT sipam.gov.br](mailto:artur.amos@sipam.gov.br)

Fabio Augusto das Dores Silva
[fabio.silva AT sipam.gov.br](mailto:fabio.silva@sipam.gov.br)

Marcelo Campos Brito
[marcelo.brito AT sipam.gov.br](mailto:marcelo.brito@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.

Publié par l'OSGeo, le Journal de l'OSGeo a pour objectif de publier les résumés des conférences, étude de cas et introduction, et les concepts liés à l'open source et aux logiciels en géomatique.

Équipe éditorial :

- Angus Carr
- Mark Leslie
- Scott Mitchell
- Venkatesh Raghavan
- Micha Silver
- Martin Wegmann

Rédacteur en Chef :

Tyler Mitchell - [tmitchell AT osgeo.org](mailto:tmitchell@osgeo.org)

Remerciements

Divers relecteurs & le project GRASS News

Le *Journal de l'OSGeo* est une publication de la *Fondation OSGeo*. La base de ce journal, le source des styles L^AT_EX 2_ε a été généreusement fournie par le bureau d'édition de GRASS et R.



Ce travail est sous licence Creative Commons Attribution-No Derivative Works 3.0 License. Pour lire une copie de cette licence, visitez : creativecommons.org.



All articles are copyrighted by the respective authors — contact authors directly to request permission to re-use their material. See the OSGeo Journal URL, below, for more information about submitting new articles.

Journal en line : <http://www.osgeo.org/journal>

OSGeo Homepage : <http://www.osgeo.org>

Courrier postal : OSGeo

PO Box 4844, Williams Lake,
British Columbia, Canada, V2G 2V8

Téléphone : +1-250-277-1621



ISSN 1994-1897

This PDF article file is a sub-set from the larger
OSGeo Journal. For a complete set of articles
please the Journal web-site at:

<http://osgeo.org/journal>