

# Uma arquitetura híbrida para aplicações geográficas em dispositivos móveis

Claudio Henrique Bogossian

Instituto Nacional de Pesquisas Espaciais - INPE  
Caixa Postal 515 - 12227-010 - São José dos Campos - SP, Brasil  
[claudio.bogossian@gmail.com](mailto:claudio.bogossian@gmail.com)

**Resumo:** Nos dias de hoje com a tecnologia atual a coleta de informações em campo não precisam mais ser feitas utilizando meios tão manuais devido a evolução de nossos dispositivos móveis que permitem o uso de aplicativos que reúnem funções como câmera, GPS, visualização de dados e preenchimento de informações em forma de formulários. Isso permite a visualização e o levantamento dos dados em campo usando apenas um dispositivo, mas ainda existe uma carência por parte dos aplicativos para trabalhar de uma maneira off-line por isso esse trabalho propõem uma análise de um padrão OGC para armazenamento de dados em dispositivos móveis de maneira a não necessitar de acesso a redes de dados em campo.

## 1. Introdução

Na área de geoprocessamento existem várias aplicações para levantamento de desmatamento e de uso da terra a partir de imagens de sensoriamento remoto como LANDSAT e MODIS permitindo a geração de dados para os projetos PRODES, DETER, TerraClass, entre outros. Geralmente esses processos exigem um processo de validação dos objetos geográficos demarcados, pois a mesma imagem e localização pode ter interpretações diferentes para pessoas distintas. Para resolver esse tipo de problema de validação a melhor maneira é fazer visitas de campos e levantar dados em *loco*, incluindo mídias digitais e localização usando diversos equipamentos para tal.

O uso de equipamentos em campo cria uma nova necessidade que é o uso de um dispositivo integrado com os recursos de Câmera, GPS e tela para preenchimento digital de conteúdo, pois o uso de vários equipamentos encarece e dificulta o processo de levantamento de dados em campo pois exige um levantamento e pós-processamento desses dados para a integração dos mesmos. O mercado de dispositivos móveis está em alta, existem muitas alternativas de dispositivos móveis com sistemas operacionais diferentes que permitem o uso desses recursos, além disso existem muitos aplicativos que já atendem a necessidade de levantar esses tipos de informações geográficas com os recursos citados acima. No INPE já existe uma Spatial Data Infrastructure (SDI) que contém o dados dos projetos PRODES, DETER, TerraClass, entre outros, que qualquer aplicativo móvel que consuma serviço OGC – Web Map Service (WMS, 2014) pode acessar os dados citados e visualizá-los.

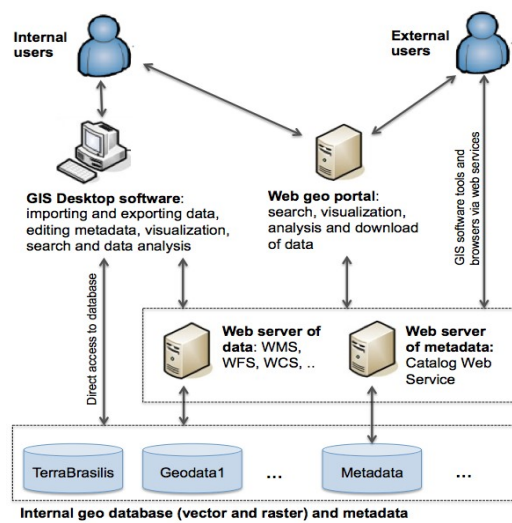


Figura 1: Spatial Data Infrastructure Fonte: SDI Spec – INPE-BOEING

O cenário brasileiro das telecomunicações acaba criando uma nova necessidade que é o uso desses aplicativos de uma maneira desconectada de qualquer rede ou servidor de armazenamento de dados, fazendo com que o aplicativo tenha que guardar o dados localmente no dispositivo e sincronizando com o servidor ao estar em contato com alguma rede que tenha acesso ao servidor de armazenamento. Isso não inclui somente o dados que são adicionados, modificados ou apagados e sim também a visualização em forma geográfica ou tabular dos dados do local que está sendo feito o levantamento, na qual geralmente não são dados pequenos por se tratar de dados vetoriais e matriciais. Portanto isso exige uma pré geração de um conjunto de dados que o validador de campo irá necessitar em sua visita, na qual deverá ser sincronizado na volta para a base dados originário. Existem aplicativos para gerar *tiles* que nada mais são que retalhos de imagens geralmente quadradas com vários níveis de escala, com o desenho de geometrias e imagens para serem visualizados em dispositivos móveis de maneira *off-line*, além disso existem ainda suítes completas para levantamento de dados de formulários dinâmicos com recursos de fotos e posicionamento e com todo o mecanismo para sincronização desses dados *on-line* e *off-line*. Apesar dessas tecnologias já existentes, não existe nada integrado para resolver esse tipo de problema incluindo mapa para navegação, ferramenta de informações e ainda usando um padrão de armazenamento, cada um tem sua própria maneira de armazenar esses dados. No inicio do ano de 2014 a *OGC Open Geospatial Consortium* lançou um novo padrão *OGC – GeoPackage* para armazenamento de dados em formato de arquivo que permite o uso em dispositivos móveis (GeoPackage, 2014).

## 1.1. Objetivo

Desenvolver um arquitetura para aplicações geográficas em dispositivos móveis capaz de acessar, visualizar e processar dados geográficos de modo on-line/off-line que permita a utilização da ferramenta TerraView 5 (TerraLib 5, 2014) para exportação de um conjunto de dados geoespaciais provenientes da Spatial Data Infraestructure (SDI) onde estão armazenados os dados do PRODES, DETER e TerraClass para trabalho em campo sem cobertura de sinal de internet ou qualquer rede, para visualização, modificação e validação desses dados e ainda permitir a volta desses dados atualizados para as bases de dados da SDI.

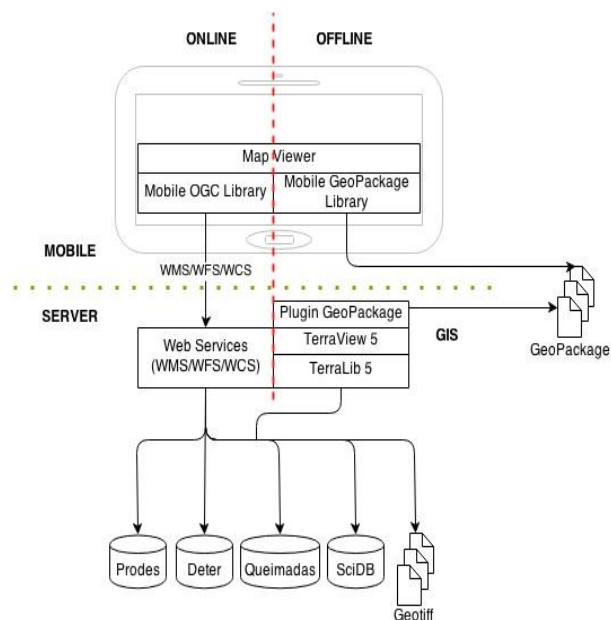


Figura 2: Arquitetura da Solução

## 1.2. Objetivo da Disciplina

Desenvolver a parte off-line da arquitetura que consiste em testar e avaliar o novo padrão de armazenamento de dados geoespaciais *OGC GeoPackage* no sentido de desempenho e tamanho do pacote com relação a outras tecnologias de armazenamento de dados geoespaciais, podendo assim validar a viabilidade de uso do padrão na arquitetura proposta.

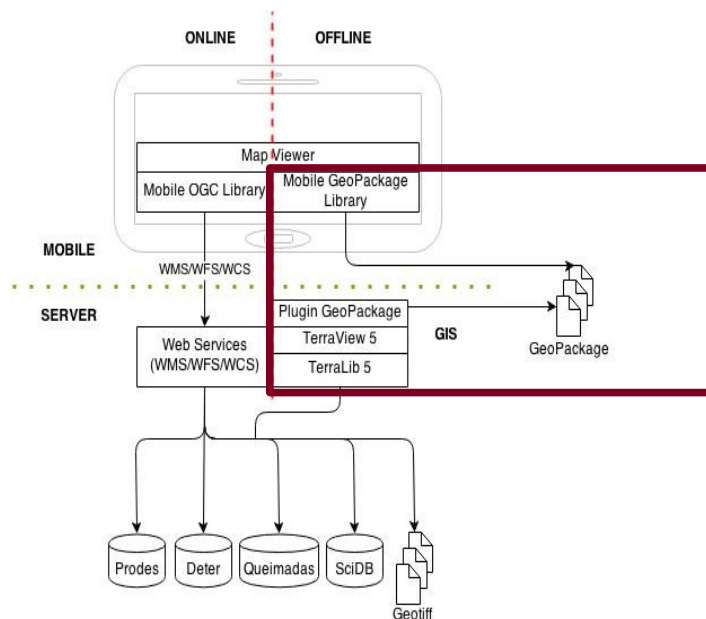


Figura 3: Arquitetura da Solução, escopo do trabalho

### 1.3. Padrão OGC GeoPackage

Em fevereiro de 2014 a *OGC Open Geospatial Consortium* definiu o novo padrão de armazenamento de dados para interoperabilidade de dados geoespaciais na qual armazena as geometrias no padrão *OGC SFS Simple Feature Specification (SFS, 2014)* que é usado pelos maiores bancos de dados com extensões espaciais do mercado. O padrão é baseado no SQLite (SQLite, 2014) que nada mais é que um banco de dados que suporta linguagem SQL em formato de arquivo sem a necessidade de um Sistema Gerenciador de Banco de Dados SGBD.

O GeoPackage ainda permite o armazenamento de tiles de imagens no formato de piramide permitindo a visualização de dados geográficos em forma de imagens em várias escalas. Para um dispositivo carente de recursos de processamento e renderização gráfica, isso é essencial, pois o tempo de desenho de geometrias complexas em um dispositivo com baixos níveis de recurso seria imensa, impedindo o uso do aplicativo em sua forma de visualização geográfica.

Os sistemas operacionais dos dispositivos móveis atuais suportam o formato SQLite nativamente facilitando o uso do mesmo nos seu aplicativos.

## 2. Metodologia de Trabalho

Para verificar a viabilidade da arquitetura, foram criados alguns ambiente de desenvolvimento para executar testes e estudar algumas bibliotecas gratuitas e disponíveis no mercado com suporte ao padrão GeoPackage (GeoPackage, 2014). Além disso foi criado uma

massa de dados vetoriais originárias dos dados do PRODES e DETER das bases de dados do SDI do INPE, para ser executado os testes de desempenho com dados controlados.

## **2.1. Ambientes**

Criamos quatro ambientes de desenvolvimento para quatro bibliotecas distintas.

### **2.1.1. Biblioteca GDAL/OGR**

Essa biblioteca tem o poder de ler e gerar os pacotes no padrão GeoPackage, mas com a limitação de somente dados vetoriais. Ela está sendo usada para gerar os pacotes de teste GeoPackage a partir das tabelas do banco de dados PostgreSQL (PostgreSQL, 2014) com a extensão espacial PostGIS (PostGIS, 2014) da SDI do INPE e para a implementação dos algoritmos para cada cenário de teste. (GDAL, 2014)

### **2.1.2. Biblioteca LUCIAD – libgpkg**

Essa biblioteca é uma extensão para o banco de dados SQLite para suportar os formatos espaciais do padrão GeoPackage. Portanto não tem utilidade para nós hoje pois precisamos de um aplicativo que consiga ler e gravar no pacote GeoPackage sem a necessidade de extensões no driver SQLite. (Luciad, 2014)

### **2.1.3. Biblioteca GeoTools**

Essa biblioteca tem uma grande quantidade de funções espaciais e a última versão já suporta gravar em banco SQLite no formato GeoPackage, ela faz leitura e gravação de dados vetoriais e matriciais e está sendo usada como referência para o desenvolvimento do *driver* para ler dados geoespaciais em um dispositivo móvel. (GeoTools, 2014)

### **2.1.4. Biblioteca OSMDroid**

Essa biblioteca permite a visualização de dados geoespaciais em dispositivos móveis com sistema operacional Android. Ela está sendo estudada para adicionar o recurso de leitura de

pacotes no padrão GeoPackage. (OSMDroid, 2014)

## **2.2. Massa de Dados Vetoriais de Teste**

Para os teste separamos dois dados distintos de tamanhos diferentes, para isso usamos os dados do DETER e do PRODES que consistem nos dados abaixo.

### **2.2.1. Dados do PRODES**

Foi usado os dados do PRODES do ano de 2001 até 2012, somente a classe Desflorestamento resultando cerca de 1.350.652 registros.

### **2.2.2. Dados do DETER**

Foi usado os dados do DETER do ano de 2004 até 2012, somente a classe Alerta resultando cerca de 439.596 registros.

## **2.3. Gerando os pacotes GeoPackage para os testes**

Para a geração da massa de dados para testes, temos que criar 3 formas de armazenamento para cada dados, PostGIS, ShapeFile (ShapeFile, 2014) e GeoPackage. Para isso geramos uma backup das tabelas *prodes\_desflorestamento* e *deter* dos dados da SDI, para podermos restaurar localmente no computador que irá fazer os testes. Após restaurar esses backups usamos a GDAL/OGR para exportar a partir da tabela, para os formatos de ShapeFile e GeoPackage no mesmo computador.

## **2.4. Testes**

Os testes serão realizados em cima dos dois dados (PRODES e DETER) com três cenários de testes para cada (Leitura completa, Filtro de Atributos e Filtro Espacial) e para cada um dos três formatos (PostGIS, ShapeFile e GeoPackage).

### 2.4.1. Tamanho dos formatos de armazenamento

Com relação ao armazenamento dos dados geoespaciais, verificamos o tamanho que cada um dos formatos apresentava para cada dado e concluímos que o GeoPackage teve um resultado muito bom e próximo ao valor de armazenamento que o PostGIS pratica, destacando o formato ShapeFile que chega a um tamanho duas vezes maior que o GeoPackage como mostra o gráfico a seguir.

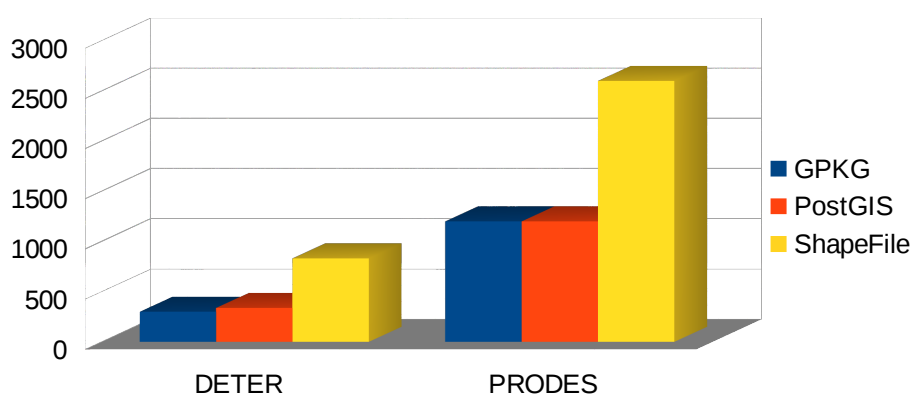


Figura 4: Tamanho dos arquivos

Tamanho (MB)	DETER	PRODES
GPKG	300MB	1200MB
PostGIS	338MB	1200MB
ShapeFile	830MB	2600MB

Tabela 1: Tamanho dos arquivos

### 2.4.2. Filtros aplicados

Para os testes utilizamos um filtro de atributos e um filtro espacial para cada dado, como mostra a tabela abaixo.

PRODES	
Filtro	Qtd Registros
Temporal (data = '2003-01-01')	138.955
Espacial (Box de Amazonia Legal)	1.300.552

Tabela 2: Filtros para os dados do PRODES

DETER	
Filtro	Qtd Registros
Temporal (data = '2005-10-30')	241.439
Espacial (Box de Amazonia Legal)	430.044

Tabela 3: Filtros para os dados do DETER

### 3. Resultados

Nos resultados dos testes tivemos um ótimo desempenho do GeoPackage superando na maioria das vezes o banco de dados PostGIS, mesmo utilizando filtros. Como usamos a mesma infraestrutura para executar o cenários de teste, os resultados foram bem próximos.

Para cada cenário executamos três vezes cada teste para evitarmos problemas com relação a concorrência de uso de recursos da infraestrutura para outro fim, por isso fazemos os gráficos em cima das médias.

#### 3.1. DETER

##### 3.1.1. Leitura Completa dos Dados

DETER (439.596)				
Processo	1 <sup>a</sup>	2 <sup>a</sup>	3 <sup>a</sup>	Media
<b>GPKG Full</b>				
Load Time	3323	6182	3624	4376
Execute Time	4457	4506	4275	4413
<b>Total</b>	<b>7780</b>	<b>10688</b>	<b>7899</b>	<b>8789</b>
<b>PG Full</b>				
Load Time	339	543	477	453
Execute Time	11360	12807	12401	12189
<b>Total</b>	<b>11699</b>	<b>13350</b>	<b>12878</b>	<b>12642</b>
<b>SHP Full</b>				
Load Time	141	99	88	109
Execute Time	11266	10730	10642	10879
<b>Total</b>	<b>11407</b>	<b>10829</b>	<b>10730</b>	<b>10989</b>

Tabela 4: Resultados Leitura Completa DETER



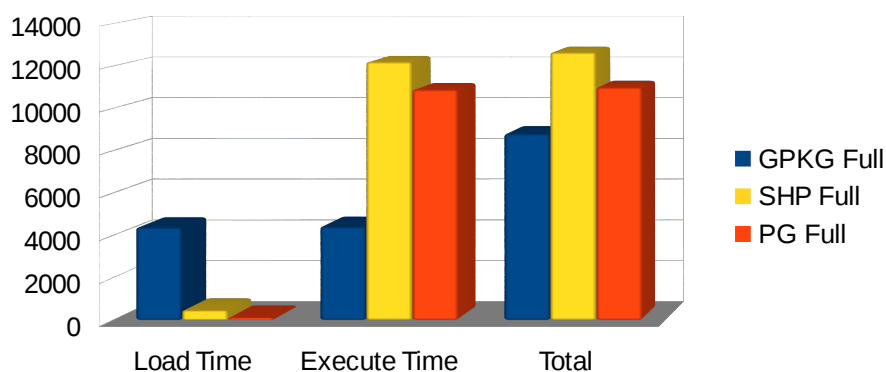


Figura 5: Resultados Leitura Completa DETER

### 3.1.2. Filtro de Atributos/Temporal

DETER (439.596)				
Processo	1 <sup>a</sup>	2 <sup>a</sup>	3 <sup>a</sup>	Media
<b>GPKG Attribute Filter</b>				
Load Time	3887	3582	3682	3717
Execute Time	3733	3261	3114	3369
<b>Total</b>	<b>7620</b>	<b>6843</b>	<b>6796</b>	<b>7086</b>
<b>PG Attribute Filter</b>				
Load Time	146	55	57	86
Execute Time	7297	5281	5289	5956
<b>Total</b>	<b>7443</b>	<b>5336</b>	<b>5346</b>	<b>6042</b>
<b>SHP Attribute Filter</b>				
Load Time	187	183	205	192
Execute Time	10680	10431	10672	10594
<b>Total</b>	<b>10867</b>	<b>10614</b>	<b>10877</b>	<b>10786</b>

Tabela 5: Resultados Filtro de Atributos DETER

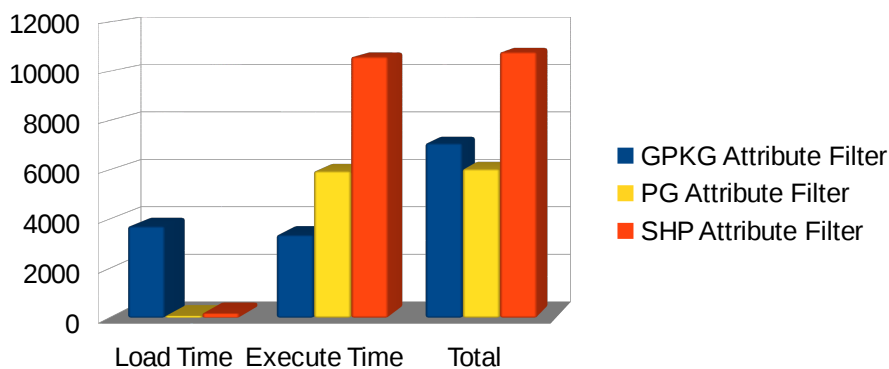


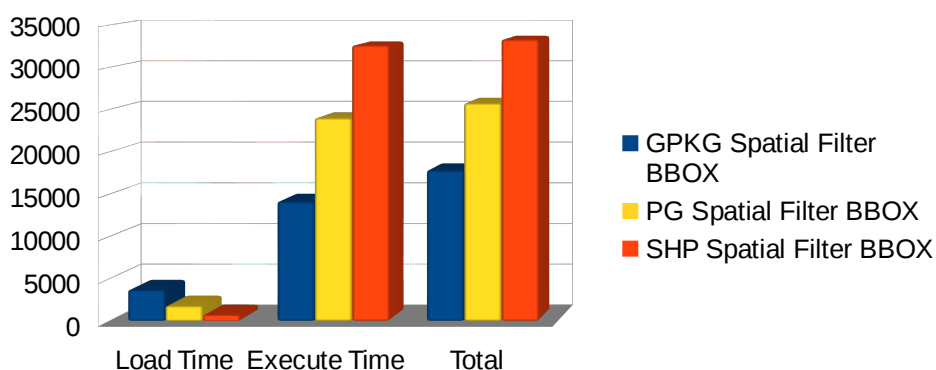
Figura 6: Resultados Filtro de Atributos DETER

### 3.1.3. Filtro Espacial

DETER (439.596)				
Processo	1 <sup>a</sup>	2 <sup>a</sup>	3 <sup>a</sup>	Media
<b>GPKG Spatial Filter BBOX</b>				
Load Time	4318	3344	3390	3684
Execute Time	15961	16543	9898	14134
<b>Total</b>	<b>20279</b>	<b>19887</b>	<b>13288</b>	<b>17818</b>
<b>PG Spatial Filter BBOX</b>				
Load Time	1064	3272	944	1760
Execute Time	24757	24860	22356	23991
<b>Total</b>	<b>25821</b>	<b>28132</b>	<b>23300</b>	<b>25751</b>
<b>SHP Spatial Filter BBOX</b>				
Load Time	613	595	881	696
Execute Time	51306	21049	25440	32598
<b>Total</b>	<b>51919</b>	<b>21644</b>	<b>26321</b>	<b>33295</b>

Tabela 6: Resultados Filtro Espacial DETER

Figura 7: Resultados Filtro Espacial DETER



## 3.2. PRODES

### 3.2.1. Leitura Completa

PRODES_DESFLORESTAMENTO (1.350.652)				
Processo	1 <sup>a</sup>	2 <sup>a</sup>	3 <sup>a</sup>	Media
<b>GPKG Full</b>				
Load Time	15208	16924	17044	16392
Execute Time	15588	16152	15663	15801
<b>Total</b>	<b>30796</b>	<b>33076</b>	<b>32707</b>	<b>32193</b>
<b>PG Full</b>				
Load Time	427	553	454	478
Execute Time	54911	50424	50341	51892
<b>Total</b>	<b>55338</b>	<b>50977</b>	<b>50795</b>	<b>52370</b>
<b>SHP Full</b>				
Load Time	280	850	1170	767
Execute Time	118360	68047	98756	95054
<b>Total</b>	<b>118640</b>	<b>68897</b>	<b>99926</b>	<b>95821</b>

Tabela 7: Resultados Leitura Completa PRODES

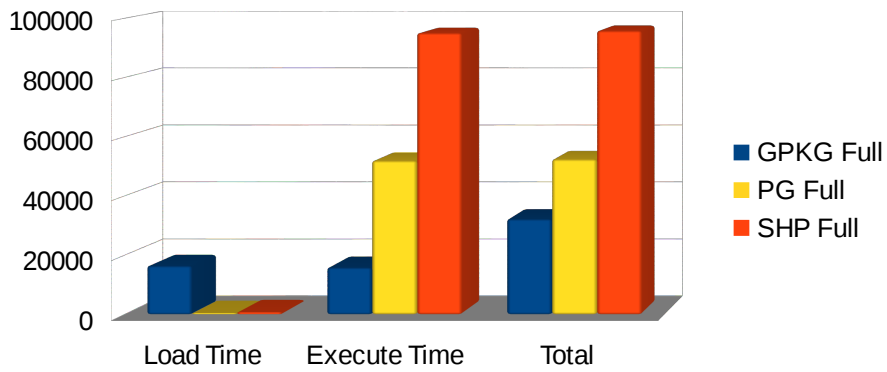


Figura 8: Resultados Leitura Completa PRODES

### 3.2.2. Filtro de Atributos/Temporal

PRODES_DESFLORESTAMENTO (1.350.652)				
Processo	1 <sup>a</sup>	2 <sup>a</sup>	3 <sup>a</sup>	Media
<b>GPKG Attribute Filter</b>				
Load Time	15612	8160	6241	10004
Execute Time	2238	2145	2301	2228
<b>Total</b>	<b>17850</b>	<b>10305</b>	<b>8542</b>	<b>12232</b>
<b>PG Attribute Filter</b>				
Load Time	472	557	422	484
Execute Time	11695	12098	11521	11771
<b>Total</b>	<b>12167</b>	<b>12655</b>	<b>11943</b>	<b>12255</b>
<b>SHP Attribute Filter</b>				
Load Time	224	379	211	271
Execute Time	30455	28288	28812	29185
<b>Total</b>	<b>30679</b>	<b>28667</b>	<b>29023</b>	<b>29456</b>

Tabela 8: Resultados Filtro de Atributos PRODES

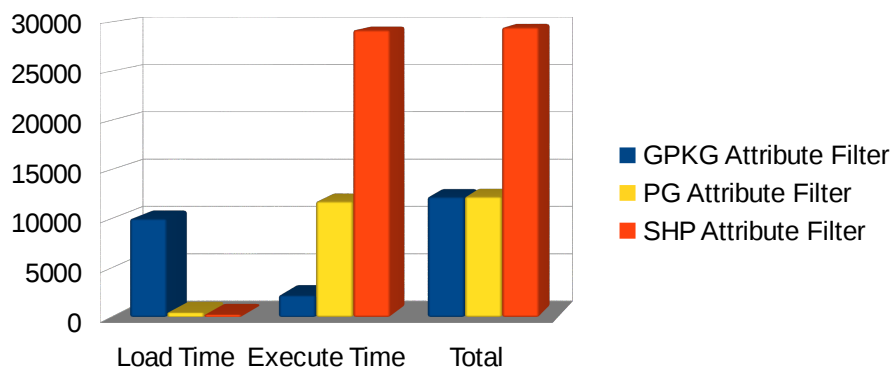


Figura 9: Resultados Filtro de Atributos PRODES

### 3.2.3. Filtro Espacial

PRODES_DESFLORESTAMENTO (1.350.652)				
Processo	1ª	2ª	3ª	Media
<b>GPKG Spatial Filter BBOX</b>				
Load Time	5341	15710	16672	12574
Execute Time	16836	16041	16466	16448
<b>Total</b>	<b>22177</b>	<b>31751</b>	<b>33138</b>	<b>29022</b>
<b>PG Spatial Filter BBOX</b>				
Load Time	467	487	503	486
Execute Time	58437	53944	52767	55049
<b>Total</b>	<b>58904</b>	<b>54431</b>	<b>53270</b>	<b>55535</b>
<b>SHP Spatial Filter BBOX</b>				
Load Time	424	822	910	719
Execute Time	56508	86130	78134	73591
<b>Total</b>	<b>56932</b>	<b>86952</b>	<b>79044</b>	<b>74309</b>

Tabela 9: Resultados Filtro Espacial PRODES

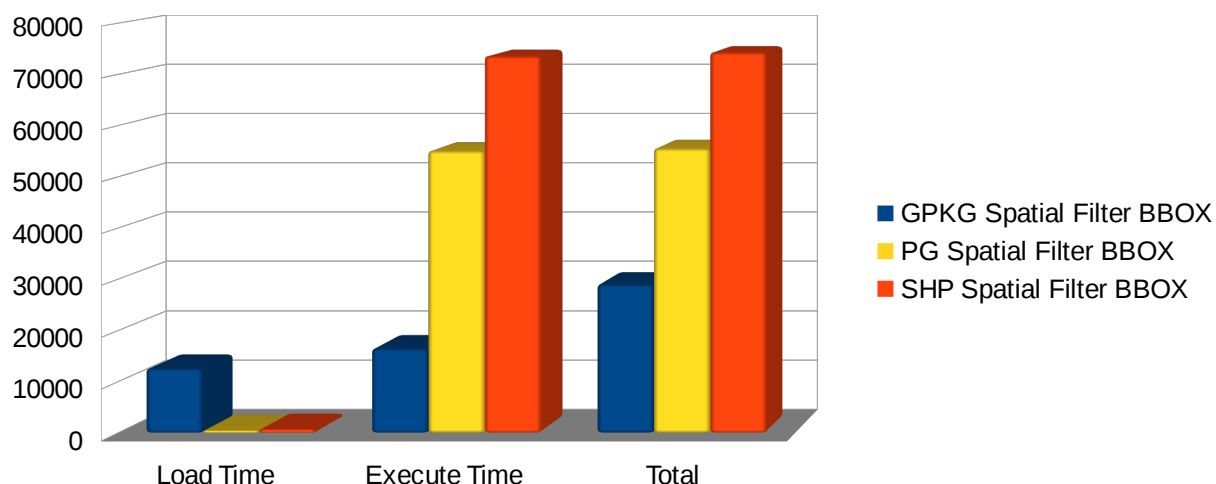


Figura 10: Resultados Filtro Espacial PRODES

### 3.3. Dados Matriciais do GeoPackage

Os dados vetoriais são adicionados aos pacotes estritamente para consultas e modificações pois a renderização desses dados exigiria um uso muito grande de recursos de processamento do dispositivo móvel. Para isso existem os tiles, são as geometrias/raster pré renderizados em forma de imagens com vários níveis de escala na forma de plano de fundo. O padrão GeoPackage permite o armazenamento desses dados no formato de piramide para a visualização em várias escalas sem uso excessivo de recurso lendo imagens com alto nível de

resolução em escalas mais baixas.

#### **4. Dificuldades Enfrentadas**

O padrão OGC GeoPackage é relativamente recente portanto existem poucas bibliotecas que o atendem conforme sua definição, além do que não existem muitas discussões sobre o assunto, isso faz com que tenhamos que testar as bibliotecas para ver o estado delas e em muitos casos complementar funcionalidade que faltam, ou usar duas ou mais para resolver um só problema. Ainda tendo as poucas bibliotecas que implementam o padrão, essas ainda não apresentam maturidade o suficiente para um ambiente de produção.

Não existe um gerador de tiles integrado com saída para GeoPackage, isso dificulta até a implementação de qualquer leitor do pacote em dispositivo móvel, que por sinal também não existe nenhum implementado.

#### **5. Conclusões**

O padrão OGC GeoPackage atende totalmente a arquitetura da solução proposta para o uso na interoperabilidade tanto em ambiente desktop como em dispositivos móveis de maneira on-line e off-line, permitindo assim a implementação de aplicativos de geoprocessamento para uso em campo em locais sem a cobertura de sinal de operadoras de telecomunicações.

O tamanho do arquivo GeoPackage ficou relativamente próximo ao tamanho do PostGIS e muito menor que o ShapeFile, concluímos então que o pacote tem potencial e não está fora da realidade dos meios atuais de armazenamento de dados geoespaciais. O tempo de acesso

ao dados no pacote GeoPackage foi mais rápido em todos os cenários de teste, nos dando mais certeza ainda da qualidade do padrão.

#### **6. Próximos Passos**

Para dar continuidade ao desenvolvimento da arquitetura proposta, atualmente estamos trabalhando no desenvolvimento do Plugin que permitirá a exportação/importação de dados vetoriais e matriciais no padrão GeoPackage dentro do TerraView 5. Estamos trabalhando em paralelo na implementação do aplicativo móvel para sistemas operacionais Android que permitirá a a visualização de dados matriciais e a leitura/escrita dos dados vetoriais dentro de pacotes.

O próximo passo, seria o desenvolvimento de um ferramental para a sincronização dos

pacotes alterados de maneira off-line, para devolvê-los para a base de dados da SDI.

## 7. Referências Bibliográficas

SQLite. SQLite Home Page. Junho/2014. Disponível em: <<http://www.sqlite.org/>> . Acesso em: 30 Junho 2014.

PostgreSQL. PostgreSQL: the world's most advanced open source database. Junho/2014. Disponível em: <<http://www.postgresql.org/>>. Acesso em: 30 Junho 2014.

PostGIS. PostGIS: Spatial and Geographic Objects for PostgreSQL. Junho/2014. Disponível em: <<http://www.postgis.net/>>. Acesso em: 30 Junho 2014.

ShapeFile. ESRI Shapefile Technical Description. Junho/2014. Disponível em: <<http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf>>. Acesso em: 30 Junho 2014.

GeoPackage. OGC GeoPackage. Junho/2014. Disponível em: <<http://www.geopackage.org/>>. Acesso em: 30 Junho 2014.

WMS. Web Map Service | OGC(R). Junho/2014. Disponível em: <<http://www.opengeospatial.org/standards/wms>>. Acesso em: 30 Junho 2014.

TerraLib 5. start – TerraLib5. Junho/2014. Disponível em: <<http://www.dpi.inpe.br/terralib5/wiki/doku.php>>. Acesso em: 30 Junho 2014.

SFS. Simple Features for OLE/COM | OGC(R). Junho/2014. Disponível em: <<http://www.opengeospatial.org/standards/sfo>>. Acesso em: 30 Junho 2014.

SDI Spec – INPE-BOEING. Spatial Data Infrastructure Specification Report - INPE-Boeing Project. Março/2014. p 4

Luciad. Luciad GeoPackage Viewer. Junho/2014. Disponível em: <<http://demo.luciad.com/GeoPackage/>>. Acesso em: 30 Junho 2014.

GeoTools. GeoTools: GeoTools 11-RC1 Released. Junho/2014. Disponível em: <<http://geotoolsnews.blogspot.com.br/2014/02/geotools-11-rc1-released.html>>. Acesso em: 30 Junho 2014.

GDAL. GeoPackage . Junho/2014. Disponível em: <[http://www.gdal.org/drv\\_geopackage.html](http://www.gdal.org/drv_geopackage.html)>. Acesso em: 30 Junho 2014.

OSMDroid. Osmdroid/osmdroid . Junho/2014. Disponível em: <<https://github.com/osmdroid/osmdroid>>. Acesso em: 30 Junho 2014.