# Catálogo Integrado Imagens – INPE

Versão 23/05/2024

1	Intro	odução	2
	1.1	STAC	2
2	Rote	eiro para utilização do repositório de dados no servidor STAC do INPE	4
	2.1	Utilizando o STAC Browser	4
	2.2	Acessando o serviço STAC através do plugin STAC API Browser do QGis	6
	2.3	Acessando o catálogo via STAC API REST 1	0
	2.4	Acessando o catálogo usando a linguagem Python 1	2
3	O se	erviço de visualização Tile Mapping Service1	3
	3.1	Acessando o serviço de visualização TMS no QGis 1	3
	3.2	Acessando o serviço de visualização TMS em aplicações web 1	6

#### 1 Introdução

Como parte do aperfeiçoamento dos produtos e serviços disponibilizados pelo INPE à sociedade, um catálogo integrado de dados geoespaciais está sendo implementado no endereço <a href="https://data.inpe.br">https://data.inpe.br</a>. Essa iniciativa faz parte da modernização da infraestrutura de serviços para acesso às imagens de satélites do acervo do instituto. Esse portal foi criado com o intuito de facilitar a pesquisa e obtenção das imagens disponibilizadas gratuitamente. Esse novo serviço tem como base protocolos modernos de compartilhamento de dados, adequados ao uso em ambientes de ciência de dados e computação em nuvem.

Essa iniciativa é ancorada dentro do Programa Base de Informações Georreferenciadas (BIG), em colaboração com o projeto Brazil Data Cube (BDC) e o Laboratório de Geração de Imagens (LGI) da Divisão de Geoinformática e Observação da Terra (DIOTG). Utilizamos a infraestutura de hardware hospedada no centro de dados na infraestrutura computacional da Coordenação de Infraestrutura de Dados e Supercomputação (COIDS).

Esse é um trabalho em andamento, não tem todas as imagens do INPE catalogadas e algumas das coleções que apresenta também estão disponíveis em outros pontos de acesso, como o portal de visualização de dados do Brazil Data Cube (<u>BDC Explorer</u>), ou o catálogo de imagens CBERS e Amazônia-1 mantido pela então Divisão de Geração de Imagens (<u>Catálogo DGI</u>).

O catálogo integrado foi construído em conformidade com a especificação padrão STAC, acrônimo do termo *SpatioTemporal Asset Catalog* em inglês. O STAC foi desenvolvido pela comunidade de observação da Terra por satélites, considerando as inovações tecnológicas no uso da web para oferecer dados e serviços, visando simplificar e flexibilizar o desenvolvimento de aplicações.

#### 1.1 STAC

A especificação STAC é uma linguagem comum para descrever informações geoespaciais, para que possam ser indexadas, descobertas e utilizadas com mais facilidade. Todos os detalhes da especificação podem ser vistos em <u>https://stacspec.org</u>. O foco do STAC é tornar o mais fácil possível para os provedores exporem seus dados aos usuários.

Os conceitos principais do STAC são descritos a seguir. Um **ativo** espaço-temporal (*spatiotemporal asset*) refere-se a qualquer arquivo que representa informações sobre a Terra relacionadas a uma determinada região geográfica e um determinado intervalo ou

2

instante de tempo (por exemplo uma imagem de satélite ou um dado de coleta em campo). Um **item** representa um conjunto atômico e inseparável de dados e metadados, com um ou mais itens. Uma **coleção** agrega os itens que são semanticamente similares. Um **catálogo** agrega coleções ou ainda outros catálogos. Os itens possuem ativos, ou seja, arquivos de dados que possuem uma identificação única na internet, através da da qual podem ser acessados. A Figura 1 ilustra o modelo de dados de um catálogo STAC.



Figura 1 – Representação de um catálogo STAC. Fonte: Brazil Data Cube (https://data.inpe.br/bdc/web/en/stac-spatiotemporal-asset-catalog-2/).

As próximas seções mostram roteiros para acesso ao catálogo integrado através de diferentes interfaces e aplicações.

#### 2 Roteiro para utilização do repositório de dados no servidor STAC do INPE

#### 2.1 Utilizando o STAC Browser

- 1. Acesse o link <u>https://data.inpe.br</u>
- 2. O STAC Browser é uma aplicação web que permite navegar nas coleções de imagens
- 3. Escolha a coleção de imagens de interesse. Por exemplo a CBERS-4A WPM fusionadas



- 4. Observe as informações sobre a coleção
- 5. Essa interface permite a filtragem por período, data inicial, data final, e por área de interesse (AOI)



6. filtrar por AOI marque a caixa *Filter by spatial extent*. Altere o zoom e posicione para fazer com que a AOI fique dentro do quadrado em destaque. Clique em *Submit*.

Description		Items
This collection contains 2 Fusion (PCA) method, with product is derived from th	I meter high-resolution, RGB products, generated using the Principal Components In values coded between 1 and 255, with 0 being reserved for "No Data". This is original CBERS-AA WPM Lavel-A Digital Number with 10 bit of quantization.	Arra Demos Next : Quelds Fitters
	Columnitor ( periff) ( Col) ( Columnitor ( Colo))	
License Temporal Extent	Creative Commons Attribution 4.0 International Statement Statement Attribution 4.0 International	Filters Temporal Extent
	action action and the statement of the one	8
		X8 times in Chordhated Universal Time (UTC).
•		Spatial Extent
-		Contract of shares errors
Provider		- Later & Constraints produces
> National Institu	te for Space Research (INPE) COLORD COLORD COLORD COLORD	
		Rems per page
Metadata		Default (12)
		Number of Semi-requested per page, mas 19993 and
General		
Data Version	1	Store Read
GSD	2m	

7. As imagens que atendem os critérios de busca ficam listadas, da mais recente até a mais antiga. Os resultados estão paginados pelo número de itens por página. Observe os botões de *Next* no final da lista.

		Expetial Extent	
Provider		T	
> National Institu	Ite for Space Research (INPE) (#20000) (#20000) (#20000)		
		Rems per page	
Metadata		Celault (CD)	-
General		Number of terms requested per page, mas, 10000 her	
Data Version	1	School Read	/
GSD	2 m		
Platform	(bers-ka		
instruments	wpm		
Constallation	chers-4a	1460	
Bdc			
Type	collection		
Public	2	C8ER54A_WPM_PCA_RG8321_2023	CBERS4A_WPM_PCA_RGB321_2023
Periodian		(m)	(ma)

- 8. Clique em uma imagem de interesse e observe os seus metadados. Cada imagem é um item da coleção. Cada item tem recursos associados (*assets*), os quais possuem uma URI ou seja, um identificador único de um arquivo na internet. Neste exemplo, os recursos dessa imagem, ou item, são:
  - a. Tci: *True Color Image*, uma composição RGB em um único arquivo em formato Cloud Optimized Geotiff (COG)

b. Thumbnail: uma amostra da imagem em formato PNG



 Clique no recurso tci. Observe os links para download ou cópia da URL da imagem (URL - Uniform *Resource Locator*, ou seja, o endereço único da imagem da internet). Recursos em formato Cloud Optimized Geotiff (COG) podem ser visualizados via o serviço Tilling Mapping Service (TMS).

		Collection		
		Consolution Cases-day Components Strayeout, so Metadata General Time of Data	WPM - M lection con fusion (PCA toto Au Lift)	Utispectral and Panchromatic Bands Fusioned tains 2 meter high-esolution, RGB products, generated using the Principal 0 method, with wakes coded between 1 and 256, with 0 being reserved fo - ISENSED - US 0000 AM UTC
Assets		Time of Data	ends	\$(2)2923, 3:00:00 AM UTC
∨ tci		Created Updated		6/22/2023, 12:15:50 AM UTC 6/22/2023, 12:15:50 AM UTC
Cloud-Optimized	Corp Life: Show on map	Electro-Opti Cloud Cover	cal	via
Metadata				
General				
Created	6/22/2023, 12:00:36 AM UTC			
Updated	6/22/0023, 12:15:50 AM UTC			
Bdc				
Size	6,106,201,237			
Chunk Size	X: 512 Y: 512			
March 1994	w 70.000			

## 2.2 Acessando o serviço STAC através do plugin STAC API Browser do QGis

- Faça a instalação do complemento (plug-in) STAC API Browser para o sistema de informações geográficas QGis. Este plugin permite navegar diretamente nas coleções de um servidor STAC diretamente na interface QGis
- 2. Inserir o endpoint do serviço de conexão STAC ao repositório do INPE:

https://data.inpe.br/bdc/stac/v1

	TAC API Breaser		
learch Results Settings		Edit Connection	
Connections	Connection Information		
NPE-data	Nome INPE-data		
New Edit Remove	URL https://data.inpedi/bdc/stac/v/	v	
Collections	Sem autenticação		
	Opções		
13 STMC collection(a)	API Capabilities		*
to a construction of the	Page size	10	e :
MAADDex. WWF - Level 4 - Doud Optimize CREME-4AVMM - Availagement of another CREME-4AVMM - Availagement and parches Dete Colle Landrai - (www 2 - LC) 16 Gray Data Cube Bentoni 2 AVM - Lavel-3 A - LC) 1 Landraid Clarket and 2 AVM - Lavel-3 A - LC) 1 Landraid Clarket and 2 AVM - Lavel-3 A - LC) 1 MCP 1301 - 4004 - Claud Clarket and available MCP 1301 - 4004 - Claud Clarket and available Statement - 1 - Lavel - 1 - Inter-Fersman M Biol St Satement - 1 - Lavel - 1 - Inter-Fersman M Biol St Satement - Months - I avail - P Fetch collectors File the ty date * 4 Extende (struct defined peter smaller)	Anaccel     Macrosoft     Macrosoft     All Conformation (seases     Conformation (s <sup>+</sup> ) <sup>+</sup> Conformation UR		
Norte -23,8	Get conformance classes		
Devie -#8,748050060		Test connection	
Suil -28.50 Calcular a partir de Laye			Cancel OK
Estensão da teia do may	d Deserthar na tela		
Advanced filter			
Conte driven queryables	Bort by	· Reverse order	

3. Expresse os critérios de busca, conforme mostrado na seção anterior.



4. Observe os resultados



5. Selecionar os assets para download ou para criar camadas diretamente no QGis

10			Assets			
Antipativ C T T T O Mappen Global Terrain Coentitieethap	Nern AMA/20NA_1_WFL202402H9_034_019 6 available asset(s) Nern					
portand II Res	BAND1	image(till, application-geotill; pro optimized	elle-cloud- 🔅 Select to add as a leyer	Select to download		
WPM	BAND2	image/UR: application-geotif; pre splinized	iffercloud- 🗇 Select to add as a layer	C Belect to download		
ArcGIS REST Servers CeoNode	BANDS	image/till; application-geotilt; pro optimized	effenctiond- C Select to add as a layer	Select to download		
Metadados Prévreuellasgile Gental	BAND4	imagentift; application-repetitit; pro optimized	file-cloud- 🔅 Select to add as a layer	Gelect to dowritaed		
Name OperdirectMas	CMASK	image/UR; application-gent/R; pro optimized	iffeechood	Select to download		
A . Y	thumbnail	imagelong	The second secon	Select to download		
V Victual Danda 3 Danda 4 Danda 1 Danda 1				-22 Add assets as team, <sup>1</sup> Downlaid for assets		
V V BANDS V V BAND2 V V BAND4 V Detailint V V OpenStreetMap		1 - The	G Select footprint (View assets)			
	- 10		Add the salected footprint(s) (1) Add a	il fuotprinta		
		A STATE OF THE STA	Clear	Contract. New		

6. Manipular as camadas conforme as funcionalidades tradicionais do QGis



#### 2.3 Acessando o catálogo via STAC API REST

A API STAC do INPE foi implementada como uma interface de serviço Web para consultar o grupo de coleções mantidas no repositório. O serviço implementa a API STAC versão v1.0.0. A versão exposta no Catálogo integrado ainda está sujeita a alterações à medida que a qualidade dos metadados continua a melhorar. No entanto, essa interface já suporta pesquisa básica pelas coleções e seus itens.

#### Endereço do serviço

https://data.inpe.br/bdc/stac/v1/

#### Busca em coleções STAC

Permite acessar as informações sobre as coleções disponíveis no catálogo Data INPE.

https://data.inpe.br/bdc/stac/v1/collections

- Coleções de dados de satélites brasileiros
  - o CBERS-4 MUX Reflectância de Superfície
  - Amazonia-1 WFI Reflectância de Superfície
  - o ...

Para acessar as informações sobre uma coleção específica (ex. 'CB4A-WPM-PCA-FUSED-

1')

https://data.inpe.br/bdc/stac/v1/collections/CB4A-WPM-PCA-FUSED-1

#### **Busca por itens STAC**

Por uma questão de performance é recomendado que se faça uma busca por itens de uma determinada coleção.

#### Busca por itens em uma dada coleção STAC

Para listar todos os itens de uma dada coleção

https://data.inpe.br/bdc/stac/v1/collections/CB4A-WPM-PCA-FUSED-1/items

Por default a interface mostra de 10 em 10 itens. E a resposta inclui um link para a próxima página.

Cada item tem seu identificador único. Para buscar um item específico:

https://data.inpe.br/bdc/stac/v1/collections/CB4A-WPM-PCA-FUSED-1/items/CBERS4A WPM PCA RGB321 20240513 199 129

#### Busca por itens com critérios de busca

Atualmente a API permite a busca por itens com os seguintes critérios:

 bbox: retorna todos os itens cuja geometria tem uma interseção uma área de interesse (AOI) definida por dois pares de coordenadas geográficas, canto inferior esquerdo e canto superior direito.

https://data.inpe.br/bdc/stac/v1/search?collections=CB4A-WPM-PCA-FUSED-1&bbox=-57,-33.0,-51.0,-30.0

• **datetime**: retorno todos os itens que foram criados dentro de um intervalo de tempo de interesse.

https://data.inpe.br/bdc/stac/v1/search?collections=CB4A-WPM-PCA-FUSED-1&datetime=2024-05-01/2024-05-15

os critérios de intervalo (datetime) podem ser escritos de diferentes formas:

- 2024-05-09: no dia 09 de maio de 2024
- 2024-01-01/2024-05-16: entre os dias 01 de janeiro a 16 maio de 2024
- /2024-01-01: de 01 de janeiro de 2024 em diante
- 2024-05-16/: até 16 de maio de 2024

#### Busca por itens com dois ou mais critérios de busca

https://data.inpe.br/bdc/stac/v1/search?collections=CB4A-WPM-PCA-FUSED-1&datetime=2024-05-01/2024-05-15&bbox=-57,-33.0,-51.0,-30.0

#### 2.4 Acessando o catálogo usando a linguagem Python

Para programar o acesso ao catálogo integrado e buscar seus dados utilizando a linguagem de programação Python é possível biblioteca pystac\_client conforme mostrado a seguir. Outros exemplos podem ser encontrados em <u>https://stacspec.org/en/tutorials/1-read-stac-python/.</u>

```
# importa a biblioteca pystac_client
from pystac_client import Client
# endereço do ponto de acesso ao serviço
datainpe = "https://data.inpe.br/bdc/stac/v1/"
# cria um objeto para o catálogo
catalogo = Client.open(datainpe)
# mostra algumas informações básicas do catálogo
print(f"ID: {catalogo.id}")
print(f"Titulo: {catalogo.title or 'N/A'}")
print(f"Descricao: {catalogo.description or 'N/A'}")
# lista as coleções disponíveis no catálogo
collections = list(catalogo.get_collections())
print(f"Esse catálogo possui {len(collections)} coleções")
print("IDs coleções:")
for collection in collections:
   print(f"- {collection.id}")
# consulta uma coleção em particular "CB4A-WPM-PCA-FUSED-1"
cbers4af = client.get_collection("CB4A-WPM-PCA-FUSED-1")
print(f"ID: {cbers4af.id}")
print(f"Título: {cbers4af.title or 'N/A'}")
print(f"Descrição: {cbers4af.description or 'N/A'}")
print(f"Palavras chave: {cbers4af.keywords or 'N/A'}")
# consulta uma coleção
consulta = catalogo.search(collections=['CB4A-WPM-PCA-FUSED-1'], max items=10)
print(f" A coleção CB4A-WPM-PCA-FUSED-1 possui {consulta.matched()} itens.")
# consulta com parâmetros
bbox = [-57, -33.0, -51.0, -30.0]
intervalo = "2024-05-01/2024-05-15"
search = catalogo.search(collections=["CB4A-WPM-PCA-FUSED-1"], bbox=bbox,
                         datetime=intervalo)
items = search.item collection()
len(items)
# apresenta os resultados
for i in items:
   print(f"ID: {i.id}")
   print(f"Data: {i.datetime or 'N/A'}")
   print(f"Geometria: {i.geometry or 'N/A'}")
# recupera os ativos da 1ª imagem
primeiro = items[0]
ativos = primeiro.assets
print(ativos.keys())
print(ativos["thumbnail"].href)
print(ativos["tci"].href)
```

#### 3 O serviço de visualização Tile Mapping Service

O serviço *Tile Mapping Service* (TMS) está disponível para todas as imagens do catálogo integrado que estão em formato Cloud Optimzed GeoTiff (COG). O ponto de entrada do serviço está localizado no endereço:

# https://data.inpe.br/bdc/tms/tiles/WebMercatorQuad/{z}/{x}/{y}?<parametro=valor> &<parametro=valor>

Esta versão do serviço suporta os seguintes parâmetros:

- url: identifica a URI do arquivo COG a ser visualizado
- color\_formula: permite a execução de operações simples, orientadas a cores, sobre a imagem sendo requisitada. Essas operações estão implementadas pela biblioteca rio-color. Para usar esse parâmetro define-se- uma string que indica a operação de ajuste a ser aplicada. As operações disponíveis são ajustes chamados em Gamma, Sigmoidal, Saturation ou combinações entre elas. A documentação completa dessas operações está disponível em: <a href="https://github.com/mapbox/rio-color#operations">https://github.com/mapbox/rio-color#operações</a> está disponível em: <a href="https://github.com/mapbox/rio-color#operations">https://github.com/mapbox/rio-color#operações</a> está disponível em: <a href="https://github.com/mapbox/rio-color#operations">https://github.com/mapbox/rio-color#operations</a>.

### 3.1 Acessando o serviço de visualização TMS no QGis

1. Crie uma conexão XYZ, indicando a conexão ao servidor TMS do INPE. Insira a URL da imagem escolhida no parâmetro "url" da conexão. Por exemplo:

https://data.inpe.br/bdc/tms/tiles/WebMercatorQuad/{z}/{x}/{y}?url= https://data.inpe.br/bdc/data/S2\_L2A/v001/21/K/WV/2023/7/S2B\_TCI\_20230730T135709 \_N0509\_R067\_T21KWV\_20230730T181459.tif 2. Crie a camada a partir da conexão XYZ



 Opcionalmente, insira uma fórmula para operação das cores, como mais um parâmetro da conexão XYZ. Exemplo:

# color\_formula=gamma rg 1.3





4. Zoom para uma área da imagem (aeroporto)



#### 3.2 Acessando o serviço de visualização TMS em aplicações web

A inserção do serviço de visualização em uma aplicação web, vai requisitar o uso de uma componente de apresentação de mapas geográficos na web. Existem diferentes componentes, em diferentes linguagens de programação.

O exemplo mostrado nessa seção utiliza a biblioteca <u>Leaflet</u>, que é uma biblioteca de código aberto, na linguagem JavaScript, usada para a produção de mapas interativos em aplicações geográficas na web. E a biblioteca <u>Folium</u> para a linguagem Python, que permite combinar as capacidades do Leaflet com outras bibliotecas escritas em linguagem Python. A seguir apresentamos um programa que ilustra a apresentação um conjunto de imagens como uma camada TMS, resultando no mapa mostrado na figura a seguir.



```
# Exemplo: mostrando duas imagens em um mapa
# importando as bibliotecas usadas
import folium
import rasterio
from pyproj import Transformer
from shapely import box, normalize, ops, MultiPoint,Point
from shapely import wkt
# ponto de entrada ao serviço TMS do INPE
inpe_tms_endpoint='https://data.inpe.br/bdc/tms/tiles/WebMercatorQuad/{z}/{x}/{y}?url='
# formula de relace opcional
color_enh='&color_formula=gamma rg 1.3'
# imagens a serem mostradas
src_imgs = ['https://data.inpe.br/bdc/data/CB4A-WPM-PCA-
FUSED/v001/207/151/2024/4/CBERS4A_WPM_PCA_RGB321_20240403_207_151.tif',
            'https://data.inpe.br/bdc/data/CB4A-WPM-PCA-
FUSED/v001/206/151/2024/1/CBERS4A_WPM_PCA_RGB321_20240106_206_151.tif']
# cria uma marca no centro de cada imagem
marks = []
for img in src_imgs:
    # calcula o bounding box da imagem em EPSG:4326
    with rasterio.open(img) as src:
        xmin, ymin, xmax, ymax = src.bounds
        transformer = Transformer.from_crs(src.crs, "EPSG:4326")
        center = transformer.transform(xmin+(xmax-xmin)/2.0, ymin+(ymax-ymin)/2.0)
        marks.append(Point(center))
# cria um mapa centralizado no centroid de todos as marcas
points = MultiPoint(marks)
center = points.centroid
map = folium.Map(location=[center.x, center.y], zoom_start=10)
i=0
for img in src_imgs:
     # cria um layer TMS para cada imagem
    folium.TileLayer(tiles=inpe_tms_endpoint+img, opacity=1,attr="INPE",).add_to(map)
    # cria uma marca com o identificador para cada imagem
    img_id = img[img.rfind('/')+1:]
    folium.Marker(location=[marks[i].x, marks[i].y], popup=img_id).add_to(map)
    i=i+1
# mostra o mapa
m
# ou salva para um arquivo HTML
m.save("index.html")
```