



MINISTÉRIO DA
CIÊNCIA, TECNOLOGIA
E INOVAÇÕES



sid.inpe.br/mtc-m21c/2020/12.14.10.23-TDI

CATALOGAÇÃO DE IMAGENS DE SATÉLITES DE SENSORIAMENTO REMOTO VOLTADA PARA ACESSO A CUBOS DE DADOS DE OBSERVAÇÃO DA TERRA

Matheus Cavassan Zaglia

Dissertação de Mestrado do Curso de Pós-Graduação em Computação Aplicada, orientada pelos Drs. Lúbia Vinhas e Gilberto Ribeiro de Queiroz, aprovada em 22 de dezembro de 2020.

URL do documento original:

<<http://urlib.net/8JMKD3MGP3W34R/43PLACE>>

INPE
São José dos Campos
2020

PUBLICADO POR:

Instituto Nacional de Pesquisas Espaciais - INPE

Gabinete do Diretor (GBDIR)

Serviço de Informação e Documentação (SESID)

CEP 12.227-010

São José dos Campos - SP - Brasil

Tel.:(012) 3208-6923/7348

E-mail: pubtc@inpe.br

CONSELHO DE EDITORAÇÃO E PRESERVAÇÃO DA PRODUÇÃO INTELLECTUAL DO INPE - CEPPII (PORTARIA Nº 176/2018/SEI-INPE):

Presidente:

Dra. Marley Cavalcante de Lima Moscati - Centro de Previsão de Tempo e Estudos Climáticos (CGCPT)

Membros:

Dra. Carina Barros Mello - Coordenação de Laboratórios Associados (COCTE)

Dr. Alisson Dal Lago - Coordenação-Geral de Ciências Espaciais e Atmosféricas (CGCEA)

Dr. Evandro Albiach Branco - Centro de Ciência do Sistema Terrestre (COCST)

Dr. Evandro Marconi Rocco - Coordenação-Geral de Engenharia e Tecnologia Espacial (CGETE)

Dr. Hermann Johann Heinrich Kux - Coordenação-Geral de Observação da Terra (CGOBT)

Dra. Ieda Del Arco Sanches - Conselho de Pós-Graduação - (CPG)

Silvia Castro Marcelino - Serviço de Informação e Documentação (SESID)

BIBLIOTECA DIGITAL:

Dr. Gerald Jean Francis Banon

Clayton Martins Pereira - Serviço de Informação e Documentação (SESID)

REVISÃO E NORMALIZAÇÃO DOCUMENTÁRIA:

Simone Angélica Del Ducca Barbedo - Serviço de Informação e Documentação (SESID)

André Luis Dias Fernandes - Serviço de Informação e Documentação (SESID)

EDITORAÇÃO ELETRÔNICA:

Ivone Martins - Serviço de Informação e Documentação (SESID)

Cauê Silva Fróes - Serviço de Informação e Documentação (SESID)



MINISTÉRIO DA
CIÊNCIA, TECNOLOGIA
E INOVAÇÕES



sid.inpe.br/mtc-m21c/2020/12.14.10.23-TDI

CATALOGAÇÃO DE IMAGENS DE SATÉLITES DE SENSORIAMENTO REMOTO VOLTADA PARA ACESSO A CUBOS DE DADOS DE OBSERVAÇÃO DA TERRA

Matheus Cavassan Zaglia

Dissertação de Mestrado do Curso de Pós-Graduação em Computação Aplicada, orientada pelos Drs. Lúbia Vinhas e Gilberto Ribeiro de Queiroz, aprovada em 22 de dezembro de 2020.

URL do documento original:

<<http://urlib.net/8JMKD3MGP3W34R/43PLACE>>

INPE
São José dos Campos
2020

Dados Internacionais de Catalogação na Publicação (CIP)

Zaglia, Matheus Cavassan.

Za13c Catalogação de imagens de satélites de sensoriamento remoto voltada para acesso a cubos de dados de observação da Terra / Matheus Cavassan Zaglia. – São José dos Campos : INPE, 2020.

xxiv + 59 p. ; (sid.inpe.br/mtc-m21c/2020/12.14.10.23-TDI)

Dissertação (Mestrado em Computação Aplicada) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2020.

Orientadores : Drs. Lúbia Vinhas e Gilberto Ribeiro de Queiroz.

1. Metadados. 2. Catálogos. 3. Cubo de dados. 4. Imagens de satélite. 5. STAC. I.Título.

CDU 004.651:528.8



Esta obra foi licenciada sob uma Licença [Creative Commons Atribuição-NãoComercial 3.0 Não Adaptada](https://creativecommons.org/licenses/by-nc/3.0/).

This work is licensed under a [Creative Commons Attribution-NonCommercial 3.0 Unported License](https://creativecommons.org/licenses/by-nc/3.0/).



INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

DEFESA FINAL DE DISSERTAÇÃO DE MATHEUS CAVASSAN ZAGLIA BANCA Nº 279/2020, REG 139904/2018

No dia 22 de dezembro de 2020, às 09h00min, por teleconferência, o(a) aluno(a) mencionado(a) acima defendeu seu trabalho final (apresentação oral seguida de arguição) perante uma Banca Examinadora, cujos membros estão listados abaixo. O(A) aluno(a) foi APROVADO(A) pela Banca Examinadora por unanimidade, em cumprimento ao requisito exigido para obtenção do Título de Mestre em Computação Aplicada. O trabalho precisa da incorporação das correções sugeridas pela Banca Examinadora e revisão final pelo(s) orientador(es).

Novo título: "Catalogação de imagens de satélites de sensoriamento remoto voltada para acesso a cubos de dados de observação da Terra"

Eu, Rafael Duarte Coelho dos Santos, Presidente da Banca Examinadora, assino esta ATA, em nome de todos os membros, com o consentimento dos mesmos.

Membros da Banca

Dr. Rafael Duarte Coelho dos Santos - Presidente - INPE
Dra. Lúbia Vinhas - Orientador - INPE
Dr. Gilberto Ribeiro de Queiroz - Orientador - INPE
Dr. Pedro Riberito de Andrade Neto - Membro Interno - INPE
Dr. Jugurta Lisboa Filho - Membro Externo - UFV



Documento assinado eletronicamente por **Rafael Duarte Coelho dos Santos, Tecnologista**, em 18/05/2021, às 08:48 (horário oficial de Brasília), com fundamento no art. 6º do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site <http://sei.mctic.gov.br/verifica.html>, informando o código verificador **7338110** e o código CRC **01BF3592**.

“Um verdadeiro mestre é um eterno aprendiz”.

MASTER YI
em “League of Legends”, 2013

*A meus pais **Gerson** e **Ivone**, e à minha esposa
Carolina*

AGRADECIMENTOS

Agradeço a meus pais Gerson dos Santos Zaglia e Ivone de Carvalho Cavassan Zaglia, pelo amor e carinho durante toda minha vida.

Agradeço a minha esposa Carolina Galvão dos Santos Zaglia, por estar ao meu lado em todos os momentos.

Agradeço aos meus amigos Alberto, Carlos, Felipe, Jorge, Jefferson, Humberto, Matheus, Nicholas, Rodrigo, Rafael, entre outros, pela amizade e momentos de descontração.

Agradeço a todos meus colegas de mestrado, pela colaboração durante esta etapa.

Agradeço ao Carlos Alberto, Raphael Willian e Rolf Simões pela contribuição em diversas etapas do desenvolvimento desta dissertação.

Agradeço a todos os membros do projeto *Brazil Data Cube*, pelo tempo e contribuição.

Agradeço aos meus orientadores Dra. Lúbia Vinhas e Dr. Gilberto Ribeiro de Queiroz, pelos ensinamentos e dedicação ao orientarem esta dissertação.

Agradeço a CAPES, Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, pelo apoio financeiro.

RESUMO

Atualmente existem centenas de satélites que estão operacionais em órbita da Terra. Dezenas deles carregam instrumentos imageadores multiespectrais, visando observar a cobertura da Terra em escala global. Além disso, suas imagens são distribuídas segundo uma política de acesso livre, sem custos ou restrições de uso. Essa era de abundância de dados trouxe desafios tecnológicos e metodológicos para explorar esse imenso volume de dados. A organização de imagens de satélite em forma de cubos de dados tem se mostrado fundamental para possibilitar o processamento e análise de grandes volumes de dados, assim como para a criação de novas aplicações na área de observação da Terra. Um dos componentes básicos para a construção dos cubos de dados a partir dos provedores originais das imagens, e para permitir o acesso aos dados dos cubos, é o serviço de catálogo de dados. Neste contexto, essa dissertação teve como objetivo investigar o uso da especificação *SpatioTemporal Asset Catalog* (STAC) para criação de um catálogo de imagens de observação da Terra que permita a busca, recuperação e acesso aos metadados de coleções de imagens e cubos de dados do projeto *Brazil Data Cube* (BDC). Além disso, foi avaliado o uso dessa especificação em ambientes de computação em nuvem e sua capacidade de fornecer uma interface máquina-máquina mais eficiente para utilização em geoportais, serviços web e aplicações em ambientes de computação interativa. Como resultado, foi proposto e implementado um modelo de dados e um serviço de catálogo para o projeto BDC. Essa proposta mostrou-se eficiente para atender ao volume e diversidade de dados do projeto, bem como para a construção de diferentes aplicações.

Palavras-chave: Metadados. Catálogos. Cubo de Dados. Imagens de satélite. *STAC*.

CATALOGING OF REMOTE SENSING SATELLITE IMAGES AIMED AT ACCESS TO EARTH OBSERVATION DATA CUBES

ABSTRACT

There are currently hundreds of satellites that are operational in orbit around the Earth. Dozens of them carry multispectral imaging instruments, aiming to observe the Earth's coverage globally. Their images are also distributed according to a free access policy, with no cost or usage restrictions. This era of data abundance brought technological and methodological challenges to explore this immense volume of data. The organization of satellite images in the form of data cubes is fundamental to enable the processing and analysis of large volumes of data and the creation of new applications in the area of Earth observation. One of the basic components for the construction of the data cubes from the original image providers and to allow access to the data of the cubes is the data catalog service. In this context, this dissertation aimed to investigate the use of the *SpatioTemporal Asset Catalog* (STAC) specification to create a catalog of Earth observation images that allows the search, retrieval, and access to the metadata of image collections and data cubes of the *Brazil Data Cube* (BDC) project. We evaluated the use of this specification in cloud computing environments and its ability to provide a more efficient machine-machine interface for use in geoportals, web services, and applications in interactive computing environments. As a result, a data model and catalog service for the BDC project were proposed and implemented. This proposal proved to be efficient to meet the volume and diversity of project data and the construction of different applications.

Keywords: Metadata. Catalogs. Data cube. Satellite imagery. *STAC*.

LISTA DE FIGURAS

	<u>Pág.</u>
2.1 Sistema de referência por órbita (o) e ponto(p), para identificação das cenas da câmera MUX do satélite CBERS 4.	4
2.2 Exemplos de imagens com diferentes resoluções espaciais.	5
2.3 Faixas espectrais dos sensores a bordo do satélite Landsat-8.	5
2.4 Fluxo de processamento de dados do satélite CBERS-4.	7
2.5 STAC Model.	10
2.6 STAC Browser em um STAC estático.	11
2.7 Diagrama de sequência de um serviço STAC dinâmico.	12
2.8 Uma coleção de imagens georeferenciadas, com data de observação, são combinadas em mosaicos espaciais, que por sua vez são alinhados temporalmente, formando o Cubo de Dados.	14
3.1 Diagrama entidade relacionamento do catálogo de metadados de imagens.	19
3.2 Modelo de dados das coleções de imagens e cubos de dados do catálogo.	21
3.3 Arquitetura do serviço <i>web</i> BDC-STAC.	22
3.4 Componentes de software do serviço STAC do <i>Brazil Data Cube</i>	34
4.1 Portal Web do <i>Brazil Data Cube</i>	37
4.2 Portal Web do <i>Brazil Data Cube</i>	38
4.3 Código R, usando o RSTAC rodando no ambiente RStudio.	40
4.4 Listando as coleções disponíveis na STAC API do BDC usando o <code>stac.py</code> no IPython/Jupyter.	42
4.5 Descrevendo o cubo de dados <code>CB4_64_16D_STK-1</code> usando o <code>stac.py</code> no IPython/Jupyter.	43
4.6 Listando as dez primeiras imagens disponíveis do cubo de dados <code>CB4_64_16D_STK-1</code> <code>stac.py</code> no IPython/Jupyter.	44
4.7 Descrevendo o item <code>CB4_64_16D_STK_v001_021027_2020-07-11_-2020-07-26</code> usando o <code>stac.py</code> no IPython/Jupyter.	45
4.8 Uso do BDC-STAC no WTSS.	46
4.9 Classificação de imagens do BDC usando o SITS.	47
4.10 Arquitetura BDC ³	48
4.11 Arquitetura BDC-STAC na AWS.	49
A.1 Modelo de dados das coleções de imagens e cubos de dados do Projeto BDC.	59

LISTA DE TABELAS

	<u>Pág.</u>
2.1 Extensões do STAC utilizadas neste trabalho.	13
3.1 Número de cenas, por ano, necessárias para gerar os cubos de dados sobre o Brasil.	17
3.2 Horizontal Spatial Dimension Object.	23
3.3 Temporal Dimension Object.	23
3.4 Extensão Eletro Optical.	25
3.5 Band Object.	25
3.6 Extensão Version.	27
3.7 Extensão Checksum.	28
3.8 Propriedades da extensão BDC.	29
3.9 Propriedades da extensão BDC.	30
3.10 Parâmetros de pesquisa STAC API.	31
3.11 Parâmetros de pesquisa operação getSearchStac.	33
3.12 Parâmetros de pesquisa operação postSearchStac.	33

LISTA DE LISTAGENS

	<u>Pág.</u>
3.1 STAC Collection com extensão datacube.	24
3.2 STAC Item com extensão Eletro-Optical.	26
3.3 STAC Collection com extensão Version.	27
3.4 STAC Item com extensão Checksum.	28
3.5 Retorno da operação getCollections.	30
3.6 Collection Object.	31
3.7 Retorno da operação getFeatures.	32
3.8 STAC Item Object.	32
4.1 Uso da biblioteca stac.py.	39

LISTA DE ABREVIATURAS E SIGLAS

API	– Application Programming Interface
ARD	– Analysis Read Data
AWFI	– Wide Field Imager
AWS	– Amazon Web Service
BDC	– Brazil Data Cube
CBERS	– China-Brazil Earth Resources Satellite
CEMG	– Comitê de Estruturação de Metadados Geoespaciais
CEOS	– Committee on Earth Observation
COG	– Cloud Optimized GeoTIFF
CONCAR	– Comissão Nacional de Cartografia
CSV	– Comma-Separated-Values
CSW	– Catalog Web Service
EODC	– Earth Observation Data Cubes
ER	– Entidade Relacionamento
FWHM	– Full Width at Half Maximum
HTTP	– HyperText Transfer Protocol
IDE	– Infraestruturas de Dados Espaciais
INPE	– Instituto Nacional de Pesquisas Espaciais
ISO	– International Organization for Standardization
JSON	– Javascript Object Notation
MD5	– Message-Digest algorithm 5
MGB	– Metadados Geoespaciais do Brasil
MGRS	– Military Grid Reference System
MSI	– MultiSpectral Instrument
MUX	– Câmera Multiespectral Regular
NDVI	– Normalized Difference Vegetation Index
OGC	– Open Geospatial Consortium
OLI	– Operational Terra Imager
RDS	– Relational Database Service
REST	– Representational State Transfer
SGBD	– Sistema Gerenciador de Bancos de Dados
SGBD-R	– Sistema Gerenciador de Bancos de Dados Relacional
SITS	– Satellite Image Time Series Analysis for Earth Observation Data Cubes
STAC	– Spatial Temporal Asset Catalog
TAR	– Tape Archive
TIFF	– Tagged Image File Format
TIRS	– Thermal Infrared Sensor
TXT	– Arquivo Texto
URL	– Uniform Resource Locator
VANT	– Veículos Aéreos Não Tripulados

- WSGI – Web Server Gateway Interface
- WTSS – Web Time Series Service
- XML – Extensible Markup Language
- ZIP – Formato de compactação de arquivo

SUMÁRIO

	<u>Pág.</u>
1 INTRODUÇÃO	1
1.1 Objetivos	2
2 FUNDAMENTAÇÃO TEÓRICA	3
2.1 Imagens de satélites	3
2.2 Metadados geoespaciais	4
2.3 Catálogos geoespaciais	6
2.4 Serviços de catálogos	8
2.5 <i>SpatioTemporal Asset Catalog</i> (STAC)	9
2.6 Cubos de dados de imagens	13
2.7 Computação em nuvem	15
3 ACESSO AOS DADOS DO <i>BRAZIL DATA CUBE</i>	17
3.1 Modelo de dados do catálogo de imagens do BDC	18
3.2 Serviço STAC	21
3.2.1 Extensões STAC	22
3.2.2 STAC API	30
3.3 Implementação	33
4 ESTUDOS DE USO DA INTERFACE STAC	37
4.1 Portal de acesso e visualização das imagens do projeto BDC	37
4.2 Clientes para linguagens de <i>scripting</i>	38
4.2.1 Cliente Python	39
4.2.2 Cliente R	40
4.2.3 IPython/Jupyter	41
4.3 <i>Web Time Series Service</i> (WTSS)	46
4.4 <i>Satellite Image Time Series Analysis for Earth Observation Data Cubes</i> (SITS)	47
4.5 <i>Brazil Data CubeCloud Cover Viewer - BDC³ Viewer</i>	48
4.6 BDC-STAC na nuvem	48
5 CONSIDERAÇÕES FINAIS	51

REFERÊNCIAS BIBLIOGRÁFICAS	53
ANEXO A - MODELO DE DADOS DO BRAZIL DATA CUBE . .	59

1 INTRODUÇÃO

As mudanças ambientais globais são um dos principais temas de interesse nos dias atuais e que tem mobilizado esforços da comunidade de pesquisadores em todo o mundo com o objetivo de gerar dados e informações que possam apoiar a tomada de decisões para mitigar os impactos dessas mudanças. Nesse contexto, o monitoramento da cobertura terrestre usando imagens de satélites é de grande relevância e importância para esses estudos. Hoje, esse monitoramento é cada vez mais preciso graças às centenas de sensores a bordo de satélites que são capazes de capturar dados com cobertura global, de forma consistente, e com resolução espacial e temporal cada vez mais finas (GUO et al., 2015).

O volume de imagens de sensoriamento remoto hoje disponível tem sido chamado de *Big Earth Data*, gerando demandas por inovações tecnológicas que permitam a organização e a análise sistemática desse material para os mais diversos fins. As imagens de sensoriamento remoto são utilizadas em diferentes tipos de aplicações tais como o monitoramento de desastres (REIS et al., 2011), o monitoramento de desmatamento (VALERIANO et al., 2004) e a classificação do uso e cobertura da terra (COSTA et al., 2015).

Os estudos relacionados ao monitoramento da dinâmica de uso e cobertura da Terra requerem a análise de centenas, até mesmo milhares de imagens, a depender da abrangência da área de estudo e do intervalo de tempo de monitoramento. Por isso, o interesse nos Cubos de Dados de Observação da Terra. Tradicionalmente, os provedores de dados de sensoriamento remoto disponibilizam o seu acervo aos usuários como cenas, ou imagens, individuais, cada uma obtida por uma passagem do satélite/sensor sobre uma área, em uma data e hora. Para se trabalhar com uma série de dados no tempo, em grandes extensões, o trabalho de acesso as imagens através desses catálogos torna-se ineficiente e até mesmo inviável.

Por isso, a comunidade científica tem voltado sua atenção para o conceito de Cubo de Dados de Observação da Terra (*Earth Observation Data Cubes - EODC*). Os cubos de dados de observação da Terra visam trazer uma nova forma de organização de dados que permita o acesso a todas as imagens disponíveis para uso, como uma estrutura multidimensional, espacialmente e temporalmente contínua. Esse conceito tem sido transformado em tecnologias capazes de permitir a extração de informações a partir desses dados. Em particular, estudos e aplicações que trabalham com séries temporais de imagens são muito beneficiados pelos cubos de dados. É importante notar que os cubos de dados são construídos a partir de imagens prontas para análise,

ou seja, que possuem um grau de processamento que requeira nenhum, ou o mínimo, trabalho de preparação por parte dos usuários finais.

Giuliani e colaboradores (GIULIANI et al., 2017) e Lewis e colaboradores (LEWIS et al., 2017) apresentam iniciativas relacionadas a geração de cubos de dados de abrangência nacional para a Suíça e para a Austrália, respectivamente. No Brasil, o Instituto Nacional de Pesquisas Espaciais (INPE), através do projeto *Brazil Data Cube* (BDC), tem por objetivo desenvolver cubos de dados de observação da Terra para todo o país (FERREIRA et al., 2020).

1.1 Objetivos

O objetivo geral desse trabalho é *estudar, adaptar e implementar um componente de busca e acesso a um repositório volumoso de cubos de dados e coleções de imagens, produtos derivados de imagens e outros dados geoespaciais.*

Já os objetivos específicos dessa dissertação são:

- a) Analisar uso da especificação *SpatioTemporal Asset Catalog* (STAC) como base do componente de busca e acesso;
- b) Avaliar o atendimento da especificação em relação as demandas do projeto BDC que servirá de contexto para o desenvolvimento da dissertação;
- c) Avaliar quais as extensões são necessárias para a catalogação dos metadados de todas as informações que compõe o repositório e se necessário propor novas extensões;
- d) Implementar um serviço STAC robusto pra atender aplicações baseadas em serviços web;
- e) Implementar um serviço STAC robusto pra atender aplicações baseadas em serviços web e clientes em linguagens de programação iterativa.

Essa dissertação está organizada da seguinte forma: o Capítulo 2 apresenta os principais conceitos necessários para a compreensão do trabalho e a literatura relacionada; o Capítulo 3 apresenta as demandas relacionadas ao projeto onde a dissertação foi desenvolvida e a solução proposta; o Capítulo 4 apresenta os casos de uso da componente desenvolvido e, finalmente o Capítulo 5 apresenta as conclusões e trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Nessa Seção são apresentados os conceitos e literatura relevantes para o desenvolvimento dessa dissertação.

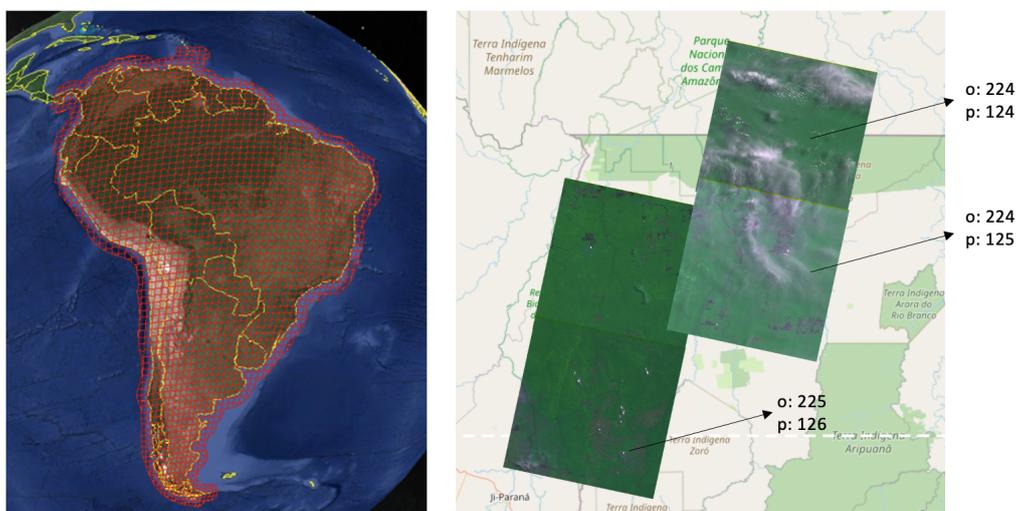
2.1 Imagens de satélites

Imagens de satélites são representações digitais de um momento da superfície terrestre. Essas imagens são adquiridas por sensores a bordo de satélites em órbita da Terra que medem a parte da energia que é refletida pelos diferentes alvos na superfície, quando atingidos por uma fonte de energia. Os sensores que medem a reflexão de energia natural do sol são chamados de sensores passivos; aqueles que fornecem sua própria fonte de energia são chamados de sensores ativos. Uma imagem digital é representada por uma matriz bidimensional de elementos, chamados *pixels*.

Após a recepção do sinal do satélite em uma antena, tem-se início o processo de geração, que consiste na transformação do sinal em imagens, organizadas espacialmente em cenas. Uma cena refere-se a uma divisão do espaço em uma partição pré determinada, comumente identificada por par *órbita* e *ponto*. A órbita refere-se ao caminho no sentido leste-oeste feito por um satélite conforme ele segue sua trajetória. O ponto refere-se a varredura lado a lado ao longo do caminho conforme a órbita progride, com o número do ponto se referindo à linha central latitudinal de cada cena. A Figura 2.1 mostra o sistema de referência por órbita e ponto para as cenas das imagens geradas pela Câmera Multiespectral (MUX) a bordo do satélite *China-Brasil Earth Resources Satellite* CBERS-4.

As imagens são caracterizadas por quatro *resoluções*, parâmetros que variam de acordo com o satélite e sensor: *espacial*, *espectral*, *radiométrica* e *temporal*. A resolução espacial se refere à área do solo capturada por um único pixel de imagem e influencia na precisão dos detalhes de uma imagem (Figura 2.2). A resolução espectral se refere ao número de canais espectrais (ou bandas) capturados por um sensor, juntamente a cada intervalo espectral associado a cada um deles (Figura 2.3). O tempo que um sensor leva para visitar uma localização geográfica específica é a resolução temporal. Finalmente, a sensibilidade de um sensor à energia eletromagnética de entrada (ou seja, as menores diferenças de intensidade que o sensor pode detectar) é dada pela resolução radiométrica, geralmente expressa em termos de profundidade de bits binária de para o valor de um pixel e se reflete também no número de níveis tonais nos quais os dados de uma determinada banda espectral são registrados por um sensor específico (KHORRAM et al., 2016).

Figura 2.1 - Sistema de referência por órbita (o) e ponto(p), para identificação das cenas da câmera MUX do satélite CBERS 4.



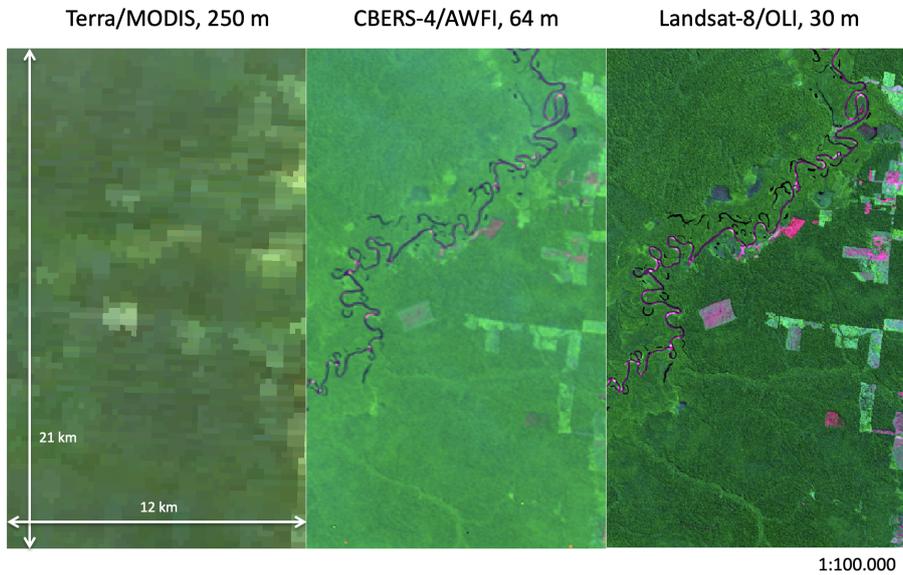
Fonte: Produção do Autor.

Assim, para gerenciar a distribuição de dados em arquivos, uma imagem se refere a uma cena obtida por um determinado sensor, a bordo de um determinado satélite, em uma passagem ou data de aquisição. Eventualmente um arquivo pode conter todas as bandas dessa passagem ou essas podem estar em arquivos separados, e com as compactações necessárias.

2.2 Metadados geospaciais

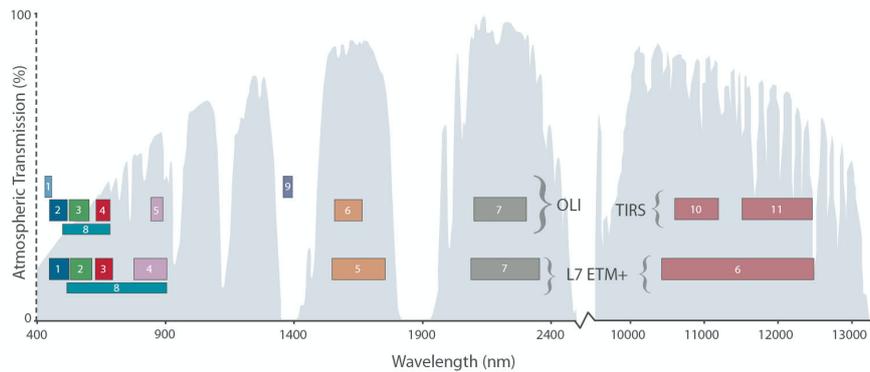
Metadados são informações sobre dados registradas para permitir que usuários descubram, acessem e avaliem a sua adequação antes de os adquirirem. No domínio geoespacial, os metadados devem caracterizar um objeto, dado, ou recurso com informações como a que se refere, qual seu propósito de criação, quando foi produzida, publicada ou atualizada, quem o criou ou desenvolveu, como foi produzida, qual sua confiabilidade, como obter acesso a ela, a que área se refere. Uma parte importante do amadurecimento da comunidade de usuários de dados geográficos, incluindo imagens de sensoriamento remoto, tem sido a capacidade de descobrir e compartilhar dados e serviços. Esse processo foi facilitado pela adoção de métodos padrão para documentar recursos geográficos através da criação de metadados especializados a esse domínio, e registrá-los de maneira organizada, similar a entradas no catálogo de fichas de uma biblioteca.

Figura 2.2 - Exemplos de imagens com diferentes resoluções espaciais.



Fonte: Produção do Autor.

Figura 2.3 - Faixas espectrais dos sensores a bordo do satélite Landsat-8.



Fonte: NATIONAL AERONAUTICS AND SPACE ADMINISTRATION - NASA (2020).

Desde meados da década de 1990, os formatos de registros de metadados evoluíram de padrões nacionais para internacionais. O padrão de metadados *ISO-19115* da *International Organization for Standardization* (ISO) define e padroniza um conjunto abrangente de elementos de metadados necessário para documentar amplamente dados geográficos. O padrão ISO-19115 é aplicável a conjuntos de dados, séries de conjuntos de dados, feições geográficas individuais e seus atributos. O padrão define um conjunto mínimo de elementos de metadados, que são necessários para servir a ampla gama de aplicações. Além desses, prevê elementos opcionais que possam complementar a descrição dos dados geográficos. Também fornece uma maneira padronizada para os usuários estenderem a documentação, com metadados específicos de um usuário ou comunidade, mas ainda assim considerando questões de interoperabilidade (AALDERS, 2005).

O padrão *ISO-19115 - Parte 2* estende os metadados definidos no ISO 19115 e identifica elementos adicionais necessários para descrever, em particular, imagens e dados geoespaciais com representação matricial, incluindo informações sobre as resoluções e outros parâmetros relacionados a representação e informações sobre o processo de aquisição e qualidade de dados. Além de um dado individualizado (uma imagem por exemplo), o padrão permite detalhar como um conjunto de dados foi adquirido, que tipo de instrumento foi usado para produzir os dados, que tipo de algoritmo foi usado para processar os dados e observações sobre a qualidade de um conjunto de dados. O padrão ISO 19130 suporta a anotação das informações de geolocalização e a ligação dessas com o processo de aquisição. Uma descrição mais completa desses padrões pode ser vista em (XIE; SHIBASAKI, 2007).

Já um perfil de metadados se refere a um conjunto básico e necessário de elementos que retrate as características dos recursos de uma determinada comunidade visando garantir sua identificação, avaliação e utilização consistente dentro dessa comunidade. Por exemplo, o *Perfil de Metadados Geoespaciais do Brasil* (Perfil MGB) foi elaborado pelo Comitê de Estruturação de Metadados Geoespaciais (CEMG), da Comissão Nacional de Cartografia (CONCAR) para facilitar e padronizar os metadados geoespaciais produzidos por órgãos e entidades públicos ou privadas no Brasil, estabelecendo um padrão nacional com base na norma ISO 19115:2003 (COMITÊ DE ESTRUTURAÇÃO DE METADADOS GEOESPACIAIS - CEMG, 2009).

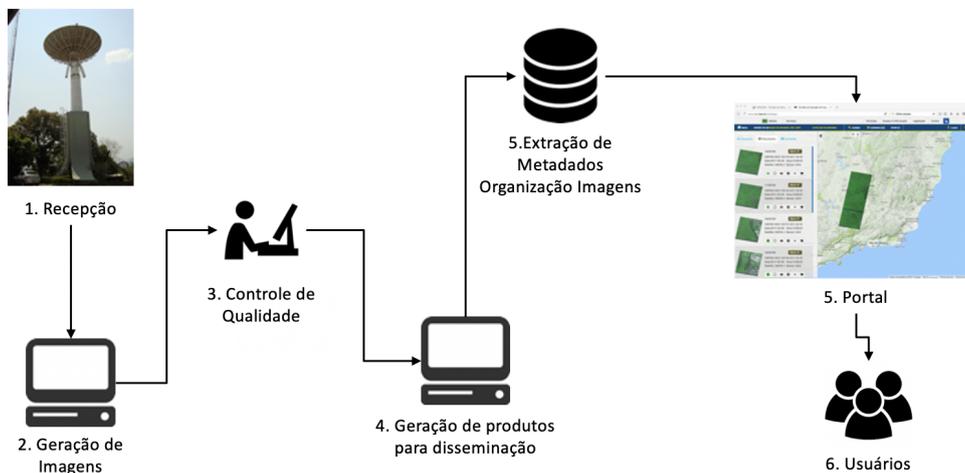
2.3 Catálogos geoespaciais

Um catálogo é formado por uma coleção de dados em conjunto com ferramentas de pesquisas e gerenciamento. Essas ferramentas de pesquisa utilizam os metadados

para buscar, listar e organizar os dados (GUPTILL, 1999) com o objetivo de conectar o produtor de dados com o usuário do dado. Essa definição, mais genérica, também pode ser aplicada a recursos digitais e em particular a coleções de imagens de sensoriamento remoto.

A partir dos anos 1980 as organizações provedoras de imagens de sensoriamento remoto passaram a recuperar seus dados em arquivos de mídias *offline* a fim de gerar todas as imagens coletadas por suas missões e distribuí-las para os usuários finais. Com a popularização da Internet, no final dos anos 1990, tornou-se possível oferecer o acesso ao acervo de modo *online* (SOUZA V. C. O.; CAMARA, 2008), e a atividade de geração das imagens e sua disponibilização para os usuários através da Internet foi completamente integrada a operação diária de recepção de dados de um órgão provedor. Assim, a forma mais comum de acessar e consultar catálogos de imagens passou a ser através de portais *web*. A Figura 2.4 ilustra o processamento da recepção e distribuição de imagens do satélite CBERS-4. Ao final desse processamento, as imagens do CBERS-4 podem ser descobertas e acessadas através do portal localizado em www.dgi.inpe.br/catalogo.

Figura 2.4 - Fluxo de processamento de dados do satélite CBERS-4.



Fonte: Produção do Autor.

2.4 Serviços de catálogos

Considerando o ambiente da internet e o crescente interesse da comunidade geoespacial de construir interoperabilidade entre serviços e aplicações, a padronização dos metadados define a terminologia necessária para descrever os dados, e é complementada com a definição de padrões para *serviços de catálogos*. Esses padrões definem como os dados podem ser consultados, requisitados e retornados através de uma Interface de Programação de Aplicações (do em inglês *Application Programming Interface (API)*). O padrão permite a implementação de aplicações orientadas a serviços, de maneira interoperável entre clientes e servidores. Ou seja, os aplicativos clientes não precisam ser desenvolvidos pela mesma organização, ou usando a mesma tecnologia, que implementou um servidor de catálogos geoespaciais. Os serviços de catálogos compõem uma parte essencial das chamadas *Infraestruturas de Dados Espaciais (IDE)*, que visam facilitar a descoberta, acesso, gerenciamento, distribuição, reutilização e preservação de recursos geoespaciais digitais. Esses recursos podem incluir mapas, imagens, serviços geoespaciais e outras ferramentas.

Na comunidade geoespacial, o *Open Geospatial Consortium (OGC)* (opengeospatial.org) é a principal organização que desenvolve padrões de interoperabilidade para dados e serviços geoespaciais. Dentre esses, está o *Catalog Web Service (CSW)* (OPEN GEOSPATIAL CONSORTIUM, 2016), um padrão OGC criado para suportar a publicação e a busca em coleções de metadados para dados e serviços geoespaciais. O padrão caracteriza elementos “consultáveis” ou seja, os elementos de metadados que podem ser solicitados por um serviço CSW. O padrão também define o conjunto de elementos obrigatórios que um serviço CSW deve tratar como consultável e “retornáveis” por um serviço CSW, e também o conjunto de elementos obrigatórios que um serviço CSW deve retornar para fornecer informações mínimas aos usuários. Uma revisão detalhada dos padrões de metadados e serviços de catálogo do OGC pode ser encontrada em (BRODEUR et al., 2019).

Existem muitos perfis de metadados e implementações dos padrões OGC, como pode ser verificado na página de produtos em conformidade com seu os padrões mantida pelo OGC em <https://www.ogc.org/compliance>, bem como exemplos de aplicações desenvolvidas com base nesses padrões (BAI; DI, 2011; SONG; DI, 2017). São usadas também como base de vários “geoportais” de internet. No entanto, existe ainda uma dificuldade grande na criação sistemática dos metadados e falta de integração com mecanismos tradicionais de busca, visto que os metadados que alimentam os

geoportais não são indexados nesses mecanismos tradicionais (BRODEUR et al., 2019).

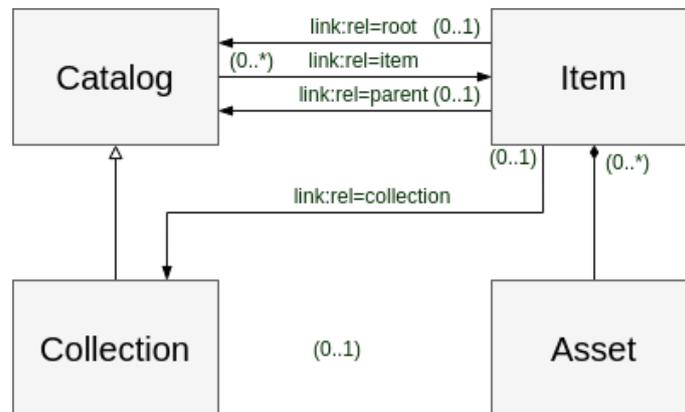
Fora das organizações OGC e ISO, um desenvolvimento interessante no tema de metadados e catálogos geoespaciais é o *Spatial Temporal Asset Catalog* (STAC). Trata-se de uma especificação aberta que foi proposta por um conjunto de 14 organizações diferentes. O objetivo dessa especificação é o de aumentar a interoperabilidade na construção de aplicações que dependem de busca por imagens de satélite no ambiente da internet. O STAC é baseado em tecnologias bem sedimentadas na construção de aplicativos internet, como o formato *Javascript Object Notation* (JSON) um formato leve para transferência e intercâmbio de dados; e também nos princípios de arquitetura *Representational State Transfer* (REST). Com isso, o STAC simplifica a implementação de catálogos estáticos, que são construídos apenas através da localização dos itens a serem disponibilizados. Essa é uma vantagem para o caso de repositórios grandes de imagens, por exemplo. O padrão CSW descreve um *serviço de consulta*, ou seja, para a implementação de um catálogo OGC CSW, é necessário a implementação de um servidor que fornece uma API para responder às consultas CSW.

Este trabalho adota o STAC como a principal ferramenta para descoberta e acesso aos repositórios de imagens, em um ambiente voltado para análise de dados sobre grandes volumes de dados, numa interface máquina-máquina em plataforma de alto desempenho. Por isso, a especificação STAC será descrita com mais detalhes ao longo dos próximos capítulos.

2.5 *SpatioTemporal Asset Catalog* (STAC)

Nos últimos anos, a indústria e as instituições ligadas à produção de dados de observação da Terra reconhecendo os problemas associados à catalogação e os padrões existentes, vêm desenvolvendo uma nova especificação chamada *SpatioTemporal Asset Catalog* (STAC) (RADIANT EARTH FOUNDATION, 2019). A especificação STAC visa padronizar como recursos geoespaciais são expostos para consulta e acesso, considerando as inovações tecnológicas utilizadas na web, compreendendo formatos de imagem estruturados para os ambientes de computação em nuvem como o (*Cloud Optimized GeoTIFF* (COG) e visando simplificar e facilitar a criação de aplicações. Conceitualmente, a especificação STAC é baseada nos conceitos do modelo de dados apresentado na Figura 2.5.

Figura 2.5 - STAC Model.



Fonte: Adaptado de RADIANT EARTH FOUNDATION (2019).

A descrição dos conceitos apresentados a seguir é adaptada da especificação original STAC:

- **Asset:** um *asset* é qualquer recurso geoespacial, como um arquivo de imagem ou arquivo vetorial, contendo informações sobre a Terra, em um determinado espaço e tempo.
- **Item:** corresponde à unidade atômica de metadados no STAC, fornecendo *links* para os *assets* associados. Um *Item* é descrito através da notação GeoJSON (BUTLER et al., 2016), como uma feição (*feature*) contendo atributos específicos como a coleção a que ele pertence, propriedades temporais, *links* para os *assets* e coleções ou catálogos associados.
- **Catalog:** fornece uma estrutura para vincular vários itens STAC juntos ou mesmo a outros catálogos ou coleções STAC.
- **Collection:** é uma especialização do catálogo que permite incluir informações adicionais sobre uma determinada coleção espaço-temporal.

A especificação conceitual do STAC permite dois tipos de implementações, uma denominada STAC estático, baseada em um conjunto de documentos JSON ligados, e outra denominada STAC dinâmico, baseada em uma API RESTful. Ambas permitem criar uma página web de itens compatíveis com o modelo de dados do STAC. A principal diferença entre um STAC estático e um STAC dinâmico é como

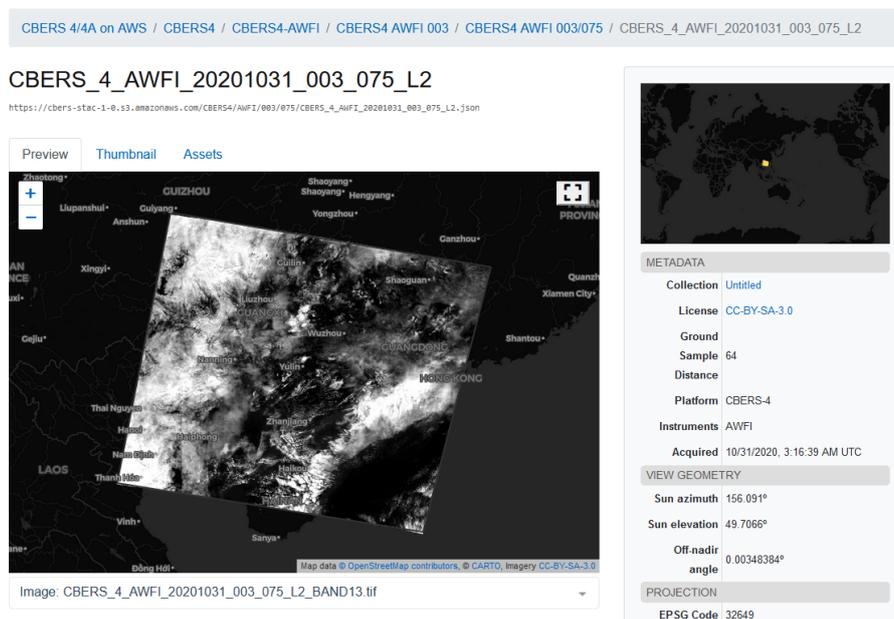
os metadados são acessados. No STAC estático, o acesso aos itens é feito através da navegação por links entre os documentos JSON. Já no STAC dinâmico, essa navegação é realizada através de uma API de serviço web que permite realizar consultas utilizando uma linguagem padrão para acessar subconjuntos do catálogo.

A Figura 2.6 apresenta a interface gráfica de uma aplicação para um navegador web que acessa um catálogo STAC estático para apresentar os metadados de um item de uma coleção. A rota para chegar a esse item é:

```
https://cbers-stac-1.0.s3.amazonaws.com/CBERS4/AWFI/003/075/CBERS_4_-  
AWFI_20201031_003_L2.json
```

onde, `https://cbers-stac-1.0.s3.amazonaws.com` é o endereço do servidor CBERS4, AWFI, 003 e 075 são as coleções, e `CBERS_4_AWFI_20201031_003_L2` um item da coleção. Através dele chega-se aos Asset.

Figura 2.6 - STAC Browser em um STAC estático.

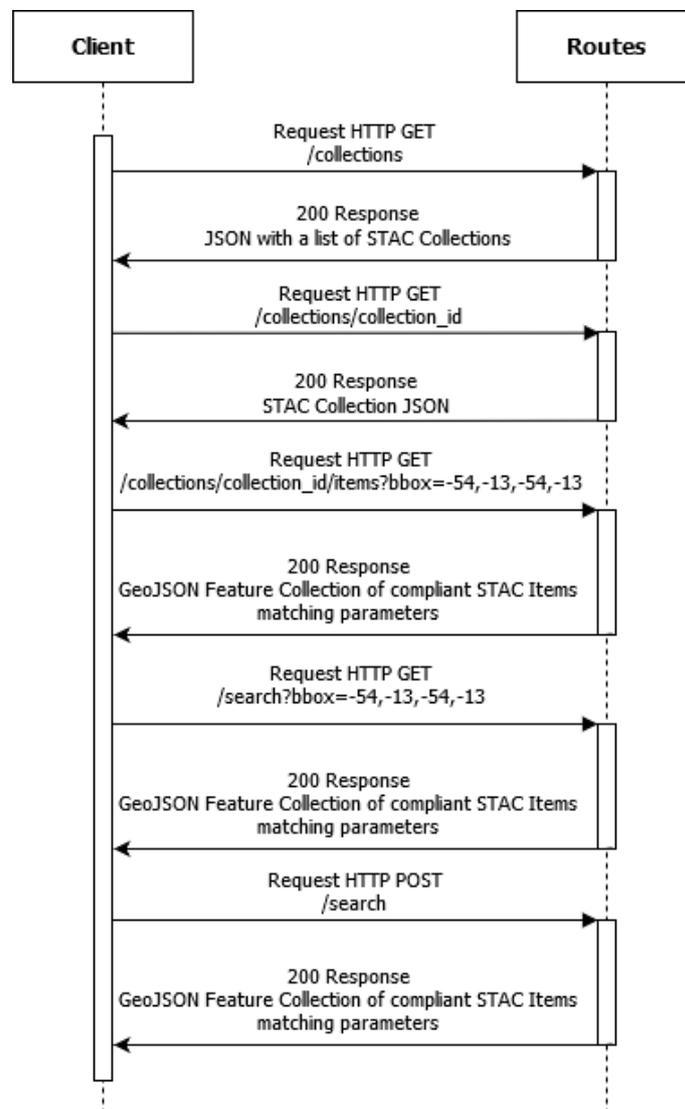


Fonte: [RADIANT EARTH FOUNDATION \(2020\)](#).

A Figura 2.7 apresenta um diagrama de sequência que fornece as rotas disponíveis num típico serviço STAC dinâmico. Pode-se observar na figura que o cliente pode consultar as coleções do catálogo através da rota `/collections`. Através do

identificador de uma coleção, é possível usar a rota `/collections/collection_id` para recuperar a entrada de uma coleção específica no catálogo. A rota `/collections/collection_id/items` permite recuperar os registros de metadados dos itens associados a uma dada coleção. Além dessas rotas, um serviço STAC dinâmico deve suportar a rota `/search`, que permite realizar busca de itens entre as diversas coleções do catálogo. Atualmente existem diversas instituições que disponibilizam seus dados através do STAC. No site *STAC Index*¹ é possível explorar uma lista de múltiplos catálogos que adotam a especificação STAC.

Figura 2.7 - Diagrama de sequência de um serviço STAC dinâmico.



Fonte: Produção do Autor.

¹<https://stacindex.org/catalogs>

Outro aspecto importante a ser considerado na especificação STAC é a possibilidade de criação de extensões. Como essa especificação define um conjunto minimalista de propriedades necessárias à maioria das aplicações, encontra-se em proposição diversas extensões para lidar com nichos mais específicos de aplicações e tipos de dados. A Tabela 2.1 apresenta uma lista das extensões utilizadas nesta dissertação de mestrado.

Tabela 2.1 - Extensões do STAC utilizadas neste trabalho.

Extensão	Descrição	Situação
Checksum	Código de verificação da integridade de um <i>asset</i> .	Proposta
Commons	Campos comuns aos itens de uma coleção que podem ser compartilhados como metadados da coleção.	Proposta
Data Cube	Metadados relacionados aos cubos de dados, como informações das dimensões do cubo.	Proposta
Electro-Optical	Metadados de imagens ópticas.	Piloto
Satellite	Metadados satélite.	Proposta
Versioning Indicators	Versionamento das coleções ou itens.	Proposta

Fonte: Adaptado de [RADIANT EARTH FOUNDATION \(2019\)](#).

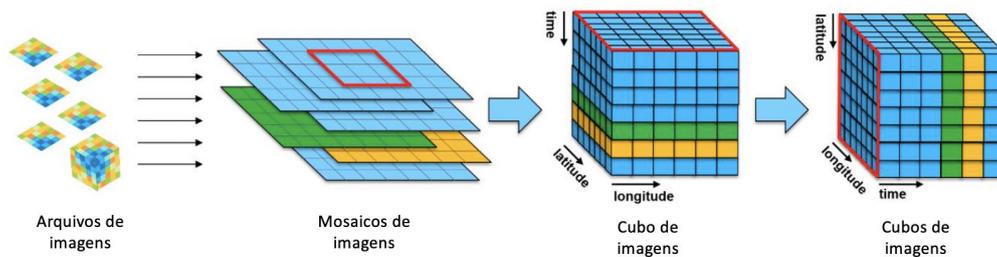
2.6 Cubos de dados de imagens

Os catálogos geoespaciais mostrados na Seção 2.3, permitem o acesso a repositórios de dados de imagens através da seleção e acesso a uma cena, ou conjunto de cenas, selecionados de acordo com algum critério de interesse. Como consequência, até algum tempo atrás, os projetos e trabalhos eram limitados pela disponibilidade e facilidade de aquisição de um conjunto de cenas. Atualmente, devido a grande disponibilidade de imagens de sensoriamento remoto, sem custos de licença, observa-se uma superação dessa limitação, com aplicações não mais definidas pela disponibilidade de cenas, pois é possível considerar a existência de observações (valores de pixel) em toda a área de interesse, sem lacunas e sem a percepção das bordas entre cenas. Essa percepção de disponibilidade “sem costuras” de dados é materializada no conceito de *Cubos de Dados de Imagens*, ou Cubos de Dados de Observação da Terra.

“Cubo de dados” é um termo genérico usado para descrever uma matriz de múltiplas dimensões, que ajudam a organizar os dados, simplificando o seu gerenciamento e potencialmente melhorando o desempenho de consultas e análises. No contexto geoespacial, as dimensões mínimas de um cubo de dados são as dimensões de referência espaciais (por exemplo, latitude e longitude) e uma terceira dimensão que se refere ao tempo, e pelo menos uma outra dimensão referente aos atributos dos dados que formam o cubo (por exemplo, bandas espectrais, índices criados a partir de bandas, ou valores de altimetria).

A Figura 2.8 ilustra o processo de construção de cubos de dados de imagens. Esse processo, que depende de decisões e processamentos intermediários, de fato rearranja o repositório de imagens convencional que disponibilizam cenas individuais. Na dimensão espacial não se tem mais o limite de aquisição por cena, e os pixels são alinhados em um sistema de referência potencialmente diferente daquele das cenas originais. Na dimensão temporal, em geral, toma-se uma granularidade regular de referência, por exemplo a cada 16 dias ou 1 mês, e para cada ponto no tempo, aplica-se uma estratégia de combinação de um ou mais observações encontradas nesse intervalo, para cada pixel.

Figura 2.8 - Uma coleção de imagens georeferenciadas, com data de observação, são combinadas em mosaicos espaciais, que por sua vez são alinhados temporalmente, formando o Cubo de Dados.



Fonte: Kopp et al. (2019).

Os cubos de dados são construídos usando imagens de satélite que foram processadas para atender um conjunto mínimo de requisitos de forma a permitir sua análise imediata, requerendo o mínimo de esforço adicional dos usuários. Além disso, devem permitir a análise integrada de observações ao longo do tempo e conjuntamente com outros dados. Esse conceito tem sido tratado na comunidade de especialistas pelo termo “*Analysis Read Data*” (ARD). O *Committee on Earth Observation* (CEOS) é a principal organização criando as definições e definindo as melhores práticas para obtenção de ARD na comunidade de sensoriamento remoto ([COMMITTEE ON EARTH OBSERVATION, 2020](#)).

O grau de pré-processamento necessário para se chegar aos ARDs é específico da aplicação, mas algumas etapas de pré-processamento são comuns e incluem a geolocalização e alinhamento espacial para permitir a comparação de observações do mesmo local na superfície da Terra ao longo do tempo; a calibração radiométrica para fornecer dados consistentes que refletem mudanças na superfície da Terra não devidas a variações do sensor; a correção atmosférica para reduzir a influência radiométrica variável da atmosfera; e a geração de marcadores de qualidade por pixel, para permitir que os usuários filtrem observações inadequadas (por exemplo, nuvens ou sombra de nuvens) ([Dwyer et al., 2018](#)).

Gomes e colaboradores ([Gomes et al., 2020](#)) apresentam uma visão geral de sete plataformas para o gerenciamento e análise de imagens de observação da Terra, comparando-as de acordo com critérios específicos de interesse da comunidade de sensoriamento remoto e geoinformática.

Por isso, no contexto do trabalho com Cubos de Dados de Imagens, novas definições de metadados sobre esse conjunto de dados torna-se necessária, bem como novos mecanismos de consulta e acesso que garantam consistência e agilidade no uso dessas informações. Nessa dissertação, a especificação STAC é utilizada para prover o acesso a coleções de imagens, ARDs e também a cubos de dados de imagens.

2.7 Computação em nuvem

Computação em Nuvem é uma tecnologia que vem crescendo rapidamente e se estabelecendo como a próxima geração do setor de Tecnologia de Informação e de negócios. Essa tecnologia é baseada em um modelo que assume acesso onipresente, conveniente e sob demanda a uma série de recursos de computação (por exemplo, servidores de armazenamento, aplicativos e serviços) que podem ser rapidamente provisionados e liberados com esforço mínimo de gerenciamento ou interação com o

provedor desses serviços. A computação em nuvem tornou-se especialmente importante em conjunção com o conceito de *Big Data*. Esse termo tem diferentes definições similares, mas se referem a dados com grande volume, variedade, complexidade, importância e que exigem novas formas de integração para revelar informações não triviais e em larga escala a partir desses dados. Os termos “*big data*” e “computação em nuvem” aparecem muitas vezes juntos, sugerindo que as plataformas de computação em nuvem são a principal alternativa para criar ambientes de processamento para *big data* (HASHEM et al., 2015).

No domínio geoespacial, os sistemas de observação da Terra a bordo de aviões, Veículos Aéreos Não Tripulados (VANT) e satélites, as simulações e dados de modelagem já estão gerando grandes volumes de dados geoespaciais díspares, dinâmicos e geograficamente distribuídos, com resoluções espaço-temporais cada vez mais refinadas. Acrescente-se a isso a coleta de dados geoespaciais através dispositivos inteligentes com sensores baseados em localização e as plataformas de mídia social que fornecem também informações geográficas sobre as atividades da vida diária. Assim, já se cunhou o termo “*Big Geospatial Data*” (LI et al., 2016) e é grande o interesse no desenvolvimento de infraestruturas, métodos e aplicações para tratar e extrair informações sobre esses dados (LI et al., 2020).

Em particular, os satélites de observação da Terra geram grandes volumes de imagens, e uma grande parte com uma política de distribuição sem custos e sem restrições de uso. Para 2019, a estimativa do volume de dados a ser produzida pelos satélites Landsat-7, Landsat-8, Terra, Aqua, Sentinel-1, Sentinel-2 e Sentinel-3 foi de 5 Petabytes (SOILLE et al., 2018). Provedores comerciais de computação em nuvem começaram a fornecer cópias desses acervos em suas plataformas de armazenamento e processamento, impulsionando o desenvolvimento de aplicações inovadoras. Grandes empresas de tecnologia da informação como Amazon, Google e Microsoft estão investindo no segmento de *big geospatial data* para oferecer plataformas para processar cubos de dados de observação da Terra integralmente, ou seja, em escala nacional e até global e com dados de décadas de observação.

3 ACESSO AOS DADOS DO *BRAZIL DATA CUBE*

O projeto *Brazil Data Cube* (BDC) tem dois objetivos principais (FERREIRA et al., 2020): (1) construir cubos multidimensionais de imagens de sensoriamento remoto, de média resolução, para todo o território brasileiro; (2) desenvolver técnicas e ferramentas computacionais para extração de informações dos cubos de dados, em particular, usando algoritmos de inteligência artificial.

Para atingir o primeiro objetivo este projeto prevê a utilização de imagens dos sensores MSI, a bordo dos satélites Sentinel-2A e 2B (EUROPEAN SPACE AGENCY, 2020), OLI e TIRS a bordo do satélite Landsat-8 (UNITED STATES GEOLOGICAL SURVEY, 2020) e também MUX e AWFII a bordo do satélite CBERS-4 (INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS, 2020b). A Tabela 3.1 mostra a quantidade de cenas a serem processadas por ano, para cada um desses instrumentos, para a construção dos cubos de dados que estão sendo gerados no projeto. Os cubos expandem-se por intervalos de tempo de 10 ou mais anos e espacialmente são organizados em subdivisões sobre grades pré-definidas, que juntas não apresentam divisões de bordas. Além dos cubos, o projeto mantém as imagens originais, as suas versões ARD e produtos derivados já processados como mosaicos, novos índices e mapas classificados. Nesse cenário, os itens a serem catalogados são diferentes daqueles presentes nos repositórios de metadados dos geoportais tradicionais, mencionados na Seção 2.3.

Tabela 3.1 - Número de cenas, por ano, necessárias para gerar os cubos de dados sobre o Brasil.

Sensor	Número de imagens	Volume de Dados
Sentinel-2	115.000	62 TB
Landsat-8 OLI	8.855	10 TB
CBERS-4 AWFII	3.600	7 TB

Quanto ao acesso, os geoportais tradicionais de distribuição de imagens são usados, principalmente, para o *download* de cenas em um trabalho manual. Já o acesso esperado aos cubos de dados é, principalmente, por APIs que são usadas para construção de aplicações principalmente no ambiente web. Um exemplo de exercício dessa possibilidade são os ambientes de computação iterativa, disponíveis para diferentes linguagens de programação, conhecidos como “*notebooks*”. Essas ferramentas têm revolucionado a criação, o registro e o compartilhamento de fluxos de processa-

mento, fornecendo uma melhor organização e visualização dos algoritmos codificados e dos dados que estão sendo processados por meio da plotagem de gráficos e mapas. Também tem se tornado uma importante ferramenta de análises geoespaciais, sendo usados para treinamento, prototipação e reprodutibilidade de experimentos.

Assim como em qualquer infraestrutura de dados, a componente de catálogo é central na arquitetura desse projeto. Por isso, considerando as características principais do projeto BDC: volume grande de dados, organização em cubos de ARD e demanda de acesso por API's, nessa dissertação essa componente foi desenvolvida visando prover um mecanismo flexível de catalogação e ao mesmo tempo eficiente para construção de diversos tipos de aplicações.

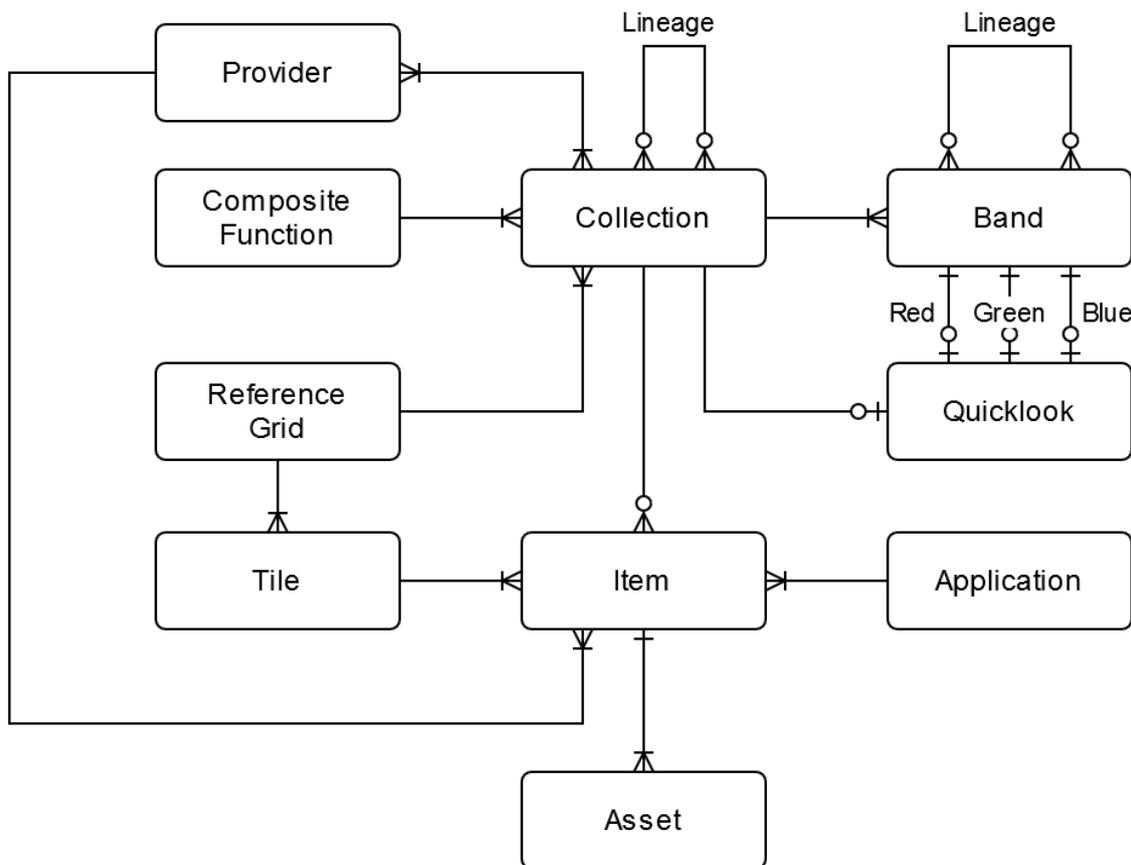
3.1 Modelo de dados do catálogo de imagens do BDC

Para armazenar e organizar os metadados do Catálogo de Imagens do BDC, foi proposto um modelo de dados capaz de representar: (a) Cubos de Dados de Imagens; (b) Coleções de Imagens utilizadas na produção dos cubos; e (c) Coleções de Imagens Classificadas referentes aos resultados dos processamentos de classificação automática realizadas sobre os cubos. O modelo proposto pode ser visto no diagrama da Figura 3.1.

Nesse modelo, a entidade **Collection** representa os diversos tipos de coleções. Uma coleção possui um identificador único, um nome, um título para apresentação pelas aplicações, uma breve descrição textual, a área geográfica de abrangência, período de imagens disponíveis, metadados que incluem: descrição completa da coleção, processamentos aplicados à coleção, plataforma de origem, créditos/autoria, licença de uso e distribuição, *links* para recursos associados (ex: artigos, portais, relatórios e documentação), entre outros. No caso de coleções que são cubos de dados, há ainda a informação de composição temporal, isto é, tanto a função de agregação dos dados quanto o intervalo de tempo dessa agregação. O modelo ainda prevê o versionamento das coleções e sua linhagem. Tanto as coleções predecessoras quanto sucessoras (versões) de uma dada coleção são tratadas nesse modelo. A questão da linhagem das coleções é tratada no auto-relacionamento denominado **Lineage**, que permite saber a origem de uma dada coleção ou sua derivação.

A entidade **Band** representa informações sobre as bandas (ou atributos) de uma dada coleção. Uma banda possui nome, resolução espacial, resolução espectral, faixa de valores possíveis, valor que indica ausência de dados (*missing value* ou *nodata*), escala de valores e metadados gerais. O auto-relacionamento **Lineage** permite man-

Figura 3.1 - Diagrama entidade relacionamento do catálogo de metadados de imagens.



Fonte: Produção do Autor.

ter a relação de proveniência das bandas. A informação sobre as bandas utilizadas na composição de uma pré-visualização da coleção (*quicklooks* ou *thumbnails*) são representadas na entidade **Quicklook**.

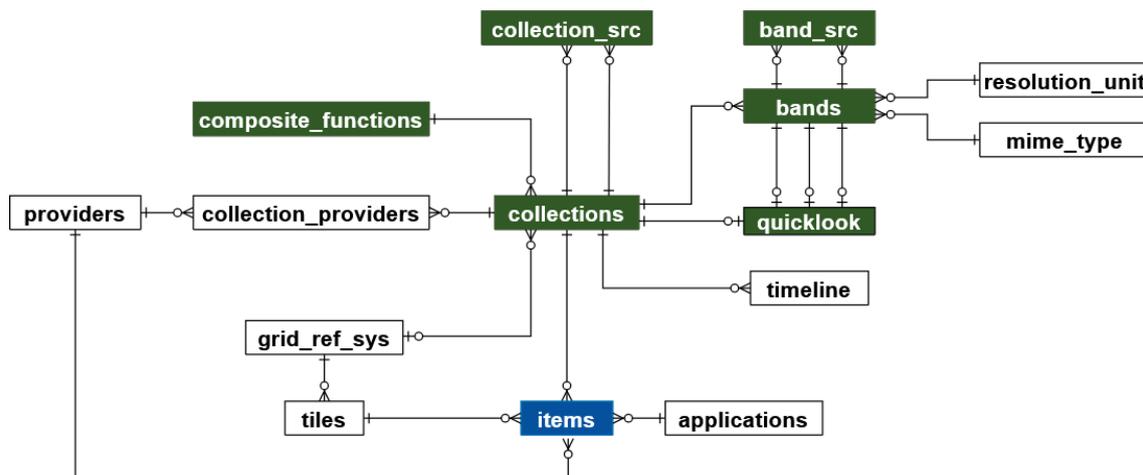
A entidade **Item** representa de fato os metadados sobre cada imagem de uma dada coleção. Um *item* pode ser composto por um ou mais *assets*. Um **Asset** representa os metadados dos arquivos, que por sua vez podem ser imagens no formato COG, arquivos binários no formato zip ou tar.gz, ou arquivos de metadados em formato texto (JSON, XML, CSV, TXT). As entidades **Item** e **Asset** também possuem atributos que permitem associar metadados mais gerais, além de informações de tamanho de arquivo e um código *hash* para garantir aos clientes do catálogo que os dados acessados estejam corretos (sem perda de bits durante a transmissão do dado).

As demais entidades do modelo possuem a seguinte interpretação:

- **Provider**: toda coleção é obtida a partir de um ou mais provedores de dados. Por exemplo, para a coleção *Sentinel-2 L-1C* tem-se como fontes de dados o repositório primário do Copernicus e a *Earth on AWS*.
- **Application**: os sistemas do Brazil Data Cube registram quais as aplicações são responsáveis pela aquisição ou geração de uma determinada imagem.
- **ReferenceGrid**: essa entidade representa as grades utilizadas como subdivisões para os cubos de dados e coleções.
- **Tile**: representa uma célula da grade, com um identificador (ex: órbita e ponto, *path* e *row*, *TileID*) e a geometria da célula.
- **CompositeFunction**: representa todas as funções de composição conhecidas pelos sistemas de geração de cubos de dados do projeto.

A Figura 3.2 mostra o mapeamento realizado do diagrama ER da Figura 3.1 para o modelo relacional. Algumas tabelas foram acrescentadas ao modelo para agilizar o processamento de alguns tipos de consultas ao catálogo de metadados, como é o caso da tabela **timeline**, que contém a lista de instantes de tempo em que a coleção possui imagens. O Anexo A contém os detalhes de todos os campos das tabelas apresentadas nesse modelo lógico.

Figura 3.2 - Modelo de dados das coleções de imagens e cubos de dados do catálogo.



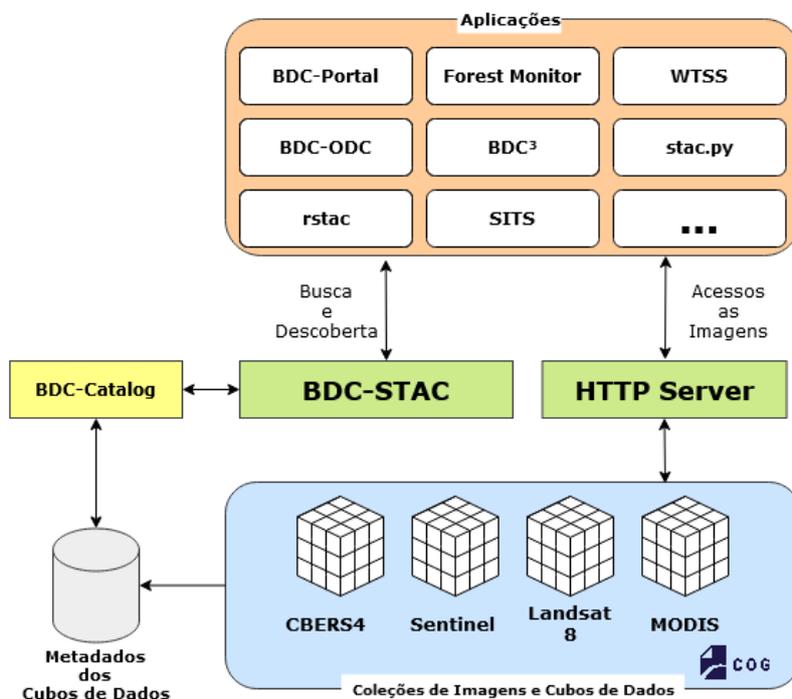
Fonte: Produção do Autor.

3.2 Serviço STAC

Com o objetivo de fornecer um serviço de catálogo capaz de atender as necessidades das aplicações do projeto BDC, que lidam com a busca e acesso às imagens, foi desenvolvido um serviço baseado no modelo de dados apresentado na Seção 3.1 e na especificação STAC (Seção 2.5). O diagrama da Figura 3.3 apresenta uma visão geral da arquitetura deste serviço e das aplicações apoiadas nele.

Como pode ser observado na Figura 3.3, o serviço denominado BDC-STAC fornece às aplicações um ponto de descoberta e acesso aos metadados dos produtos de dados do projeto BDC. Esses metadados encontram-se armazenados em tabelas, em um Sistema Gerenciador de Bancos de Dados Relacional (SGBD-R), em conformidade com o modelo apresentado na Seção 3.1 e aqui representado pelo componente BDC-Catalog. Os metadados disponíveis no repositório BDC-Catalog são transformados pelo serviço BDC-STAC em documentos JSON em conformidade com a especificação STAC na versão 0.9. A API STAC fornecida por este serviço permite que as aplicações recuperem os metadados de itens que contêm os *links* para os *assets* associados. A partir dos *links* para os *assets*, fornecidos nesse metadado, as aplicações podem realizar acesso às imagens. No projeto BDC, essas imagens são disponibilizadas no formato COG. O COG possibilita organizar internamente os dados da imagem em blocos e em pirâmides de múltiplas resoluções, de maneira que as aplicações possam utilizar o protocolo HTTP para recuperar os dados por partes

Figura 3.3 - Arquitetura do serviço *web* BDC-STAC.



Fonte: Produção do Autor.

ou mesmo navegar numa determinada resolução da pirâmide.

Ainda na Figura 3.3, pode-se verificar que existe um grande número de aplicações alvo e, conseqüentemente, uma grande variedade de requisitos em relação aos metadados necessários por cada aplicação. Para manter a simplicidade da especificação, e ao mesmo tempo ter a flexibilidade no atendimento das aplicações, usa-se o mecanismo de extensões a especificação básica. A seguir, são descritas as extensões utilizadas no serviço desenvolvido nessa dissertação.

3.2.1 Extensões STAC

Conforme mencionado na Seção 2.5, a especificação STAC define uma quantidade mínima de campos para as aplicações de uma maneira geral. No entanto, esta especificação possibilita que as implementações do STAC ampliem esses metadados básicos através de um mecanismo chamado de extensões. Nesse trabalho foi necessário aprimorar algumas extensões propostas na especificação original, visando permitir a documentação de outras características dos dados do repositório, conforme descrito a seguir.

Extensão Data Cube

A organização de dados na forma de cubos traz a necessidade de se anotar informações particulares, como as dimensões espacial e temporal dos cubos. A extensão *Data Cube*, acrescenta campos de metadados que possibilitam este tratamento. As Tabelas 3.2 e 3.3 mostram os campos incorporados nesta extensão para descrever, respectivamente, a dimensão espacial e temporal.

Tabela 3.2 - Horizontal Spatial Dimension Object.

Propriedade	Tipo	Descrição
type	string	Obrigatório. Tipo da dimensão, sempre espacial.
axis	string	Obrigatório. Eixo da dimensão (x, y).
description	string	Descrição detalhada para explicar a dimensão.
extent	[number]	Obrigatório. Extensão (limites inferiores e superiores) da dimensão como um vetor de duas dimensões.
values	[number]	Opcional, um conjunto com todos valores possíveis.

Fonte: Adaptado de [RADIANT EARTH FOUNDATION \(2019\)](#).

Tabela 3.3 - Temporal Dimension Object.

Propriedade	Tipo	Descrição
type	string	Obrigatório. Tipo da dimensão, sempre temporal.
description	string	Descrição detalhada para explicar a dimensão.
extent	[string null]	Obrigatório. Extensão (limites inferiores e superiores) da dimensão como um vetor de duas dimensões. As datas e horas devem ser compatíveis com a ISO 8601. <code>null</code> é permitido para datas em aberto.
values	[string]	Opcional, um conjunto com todos valores possíveis. As datas e horas devem ser compatíveis com a ISO 8601.

Fonte: Adaptado de [RADIANT EARTH FOUNDATION \(2019\)](#).

A Listagem 3.1 mostra o conjunto de metadados associados a um dos cubos de dados em produção no projeto. Trata-se de um cubo denominado *CB4_64_16D_STK_1*, criado a partir de imagens CBERS-4/AWFI com 64m de resolução espacial, com uma granularidade temporal de 16 dias, e método de composição temporal *stacking*. É possível observar pelos metadados que esse cubo tem imagens no período de 01/01/2016 a 31/07/2020 e que compreende uma região delimitada pelas coordenadas do retângulo [(-61,0002123 035,631935) (-26,947548 0,949458)], que basicamente compreende o território brasileiro.

Listagem 3.1 - STAC Collection com extensão datacube.

```

1  {
2    "id": "CB4_64_16D_STK-1",
3    "stac_version": "0.9.0",
4    "stac_extensions": ["datacube"],
5    "title": "CBERS-4 (AWFI) Cube Stack 16 days - v001",
6    "description": "This datacube contains the all available images from CBERS-4/AWFI
7    ↪ resampled to 64 meters of spatial resolution, reprojected and cropped to BDC_LG
8    ↪ grid, composed each 16 days using the best pixel (Stack) composite function.",
9    "license": "",
10   "extent": {...},
11   "cube:dimensions": {
12     "x": {
13       "type": "spatial",
14       "axis": "x",
15       "extent": [ -61.002123, -35.631935 ],
16       "reference_system": "+proj=aea +lat_0=-12 +lon_0=-54 +lat_1=-2 +lat_2=-22
17       ↪ +x_0=5000000 +y_0=10000000 +ellps=GRS80 +units=m +no_defs "
18     },
19     "y": {
20       "type": "spatial",
21       "axis": "y",
22       "extent": [ -26.947548, 0.949458 ],
23       "reference_system": "+proj=aea +lat_0=-12 +lon_0=-54 +lat_1=-2 +lat_2=-22
24       ↪ +x_0=5000000 +y_0=10000000 +ellps=GRS80 +units=m +no_defs "
25     },
26     "temporal": {
27       "type": "temporal",
28       "extent": [
29         "2016-01-01T00:00:00",
30         "2020-07-31T00:00:00"
31       ],
32       "values": ["2016-01-01", "2016-01-17", "2016-02-02", ...]
33     },
34     "bands": {
35       "type": "bands",
36       "values": [
37         "CLEAROB", "CMASK", "PROVENANCE", "TOTALOB",
38         "EVI", "NDVI", "BAND13", "BAND14", "BAND15",
39         "BAND16"
40       ]
41     }
42   }
43 }
44 "links": [...]
45 }

```

Fonte: Produção do Autor.

Extensão Eletro Optical

A extensão *Eletro Optical* permite descrever as informações espectrais das imagens óticas de sensoriamento remoto, e características de aquisição da imagem. Esta extensão aplica-se aos itens de uma coleção. No entanto, algumas propriedades, quando compartilhadas por todos os itens da coleção, podem ser apresentadas no nível dos metadados da coleção. As Tabelas 3.4 e 3.5 apresentam as propriedades incluídas por esta extensão. A Listagem 3.2 ilustra o uso dessas propriedades para os itens da coleção *CB4_64_16D_STK_1*.

Tabela 3.4 - Extensão Eletro Optical.

Propriedade	Tipo	Descrição
eo:gsd	number	<i>Ground Sample Distance</i> (GSD). Tamanho de cada pixel representado na imagem.
eo:bands	[Band Object]	Lista de bandas disponíveis no Item.
eo:cloud_cover	number	Porcentagem estimada da área coberta por nuvem em uma imagem.

Fonte: Adaptado de [RADIANT EARTH FOUNDATION \(2019\)](#).

Tabela 3.5 - Band Object.

Propriedade	Tipo	Descrição
name	string	Nome da banda
common_name	string	Nome comum utilizado na banda, facilitando a consulta entre instrumentos.
description	string	Descrição completa para explicar a banda.
centerwave	number	O centro do comprimento de onda em micrometros (μm)
full_width_half_max	number	<i>Full width at half maximum</i> (FWHM). Largura da banda, medida na metade do máximo da transmissão em micrômetros (μm).

Fonte: Adaptado de [RADIANT EARTH FOUNDATION \(2019\)](#).

Listagem 3.2 - STAC Item com extensão Eletro-Optical.

```
1 {
2   "type": "Feature",
3   "id": "CB4_64_16D_STK_v001_021022_2020-07-27_2020-07-31",
4   "collection": "CB4_64_16D_STK-1",
5   "stac_version": "0.9.0",
6   "stac_extensions": ["eo"],
7   "geometry": {...},
8   "bbox": [...],
9   "properties": {
10    "eo:gsd": 64,
11    "eo:bands": [{
12      "name": "BAND13",
13      "common_name": "blue",
14      "description": null,
15      "min": 0,
16      "max": 10000,
17      "nodata": -9999,
18      "scale": 0.0001,
19      "center_wavelength": 0.485,
20      "full_width_half_max": 0.035,
21      "data_type": "int16"}],
22    ...
23  },
24  "assets": {
25    "BAND13": {
26      "eo:bands": [6],
27      ...
28    }
29  },
30  "links": [...]
31 }
```

Fonte: Produção do Autor.

Extensão Commons

A extensão *Commons* não adiciona nenhuma chave nova à especificação. Sua proposta é que os metadados comuns entre os itens de uma coleção sejam agrupados e descritos no nível da coleção. Caso isto aconteça pode-se remover os metadados dos itens e mantê-los somente na coleção.

Como no projeto BDC todas as imagens de uma mesma coleção compartilham diversos metadados, o uso desta extensão facilita na composição das respostas no serviço web.

Extensão Version

A extensão *Version* permite o versionamento das coleções e itens. Também é possível depreciar versões antigas. A Tabela 3.6 apresenta as duas propriedades adicionadas por esta extensão. A Listagem 3.3 ilustra o uso dessa extensão para o registro de metadado da coleção *CB4_64_16D_STK_1*.

Tabela 3.6 - Extensão Version.

Propriedade	Tipo	Descrição
version	string	Obrigatório. Versão da coleção ou item.
deprecated	boolean	Especifica se a coleção ou item é depreciado e pode ser removido no futuro.

Fonte: Adaptado de [RADIANT EARTH FOUNDATION \(2019\)](#).

Listagem 3.3 - STAC Collection com extensão Version.

```
1 {
2   "id": "CB4_64_16D_STK-1",
3   "stac_version": "0.9.0",
4   "stac_extensions": ["version"],
5   "title": "CBERS-4 (AWFI) Cube Stack 16 days - v001",
6   "version": 1,
7   "deprecated": false,
8   "description": "...",
9   "license": "",
10  "extent": {...},
11  "links": [...]
12 }
```

Fonte: Produção do Autor.

Extensão Checksum

Esta extensão permite a utilização de *hash SHA-256* para garantir a integridade dos dados disponibilizados. A Tabela 3.7 apresenta o único campo incluído nessa especificação. A Listagem 3.4 ilustra o uso dessa extensão, onde os *assets* associados aos itens da coleção *CB4_64_16D_STK_1* incluem uma chave denominada `checksum:multihash`. As aplicações clientes podem utilizar o valor associado a esta chave na verificação de integridade dos dados recebidos do servidor.

Tabela 3.7 - Extensão Checksum.

Propriedade	Tipo	Descrição
checksum:multihash	string	<i>Multihash</i> para o arquivo correspondente, codificado em texto hexadecimal (base 16) com letras minúsculas.

Fonte: Adaptado de [RADIANT EARTH FOUNDATION \(2019\)](#).

Listagem 3.4 - STAC Item com extensão Checksum.

```

1  {
2  "type": "Feature",
3  "id": "CB4_64_16D_STK_v001_021022_2020-07-27_2020-07-31",
4  "collection": "CB4_64_16D_STK-1",
5  "stac_version": "0.9.0",
6  "stac_extensions": ["checksum"],
7  "geometry": {...},
8  "bbox": [...],
9  "properties": {...},
10 "assets": {
11   "EVI": {
12    "href": "http://brazildatacube.dpi.inpe.br/Repository/Mosaic/CB4_64_16D_STK/v001/0
        ↪ 21022/2020-07-27_2020-07-31/CB4_64_16D_STK_v001_021022_2020-07-27_2020-07-31_E
        ↪ VI.tif",
13    "type": "image/tiff; application=geotiff; profile=cloud-optimized",
14    "roles": ["data"],
15    "created": "2020-11-04T04:53:21",
16    "updated": "2020-11-04T04:53:21",
17    "checksum:multihash":
        ↪ "12202334b8a30af174901a5e78ff3bd84a474abf98570f10b0577cc855966dff9cd7",
18   },
19   ...
20  },
21  "links": [...]
22 }

```

Fonte: Produção do Autor.

Extensão BDC

Para complementar o suporte necessário às aplicações do BDC, foi desenvolvida uma extensão de propósito específico denominada *bdc* que se aplica tanto aos registros das coleções quanto dos itens. Esta extensão inclui informações da função de agregação temporal utilizada na construção dos cubos de dados (*composite_function*), informações sobre a grade usada para particionamento das imagens (*grs*), lista de *tiles* associados a uma coleção (*tiles*), informações sobre o *tile* de referência para um item da coleção (*tiles*) e a forma da composição temporal (*temporal_composition*). As Tabelas 3.8 e 3.9 apresentam as propriedades incluídas nesta extensão.

Tabela 3.8 - Propriedades da extensão BDC.

Propriedade	Objeto STAC	Valor	Descrição
composite_function	Collection	string	Tipos de processamento para escolha do pixel.
bands_quicklook	Collection	[string]	Bandas que compõe o quicklook do item.
tiles	Collection, Item	[string]	Quais <i>tiles</i> possuem imagens no cubo ou a qual <i>tile</i> a imagem pertence.
grs	Collection	string	Grade de referenciamento utilizado no cubo.
temporal_composition	Item	Temporal Composition Object	Tipo de processamento para escolha do pixel da imagem
metadata	Collection, Item	Object	Metadados adicionais não representados pelo STAC, tais como autoria e processadores usados na geração dos dados.
type	Collection	string	Tipo da coleção: datacube, collection, mosaic, classification.
crs	Collection	string	Sistema de referência de coordenadas.

Tabela 3.9 - Propriedades da extensão BDC.

Propriedade	Valor	Descrição
step	number	Valor numérico de período.
unit	string	Unidade de tempo utilizada pelo período. Pode ser <i>day</i> , <i>week</i> , <i>month</i> ou <i>year</i>
schema	cyclic continuous	Informa se o período é contínuo no tempo ou se ele possui ciclos.
cycle	step:number, unit:string	Define o ciclo do período caso o <i>schema</i> seja <i>cyclic</i>

3.2.2 STAC API

A STAC API define um conjunto de operações para a consulta e acesso dos metadados disponibilizados no catálogo. Estas operações são realizadas através de requisições HTTP utilizando o padrão RESTful. A seguir são descritos os caminhos, as respostas e os parâmetros que podem ser utilizados em cada operação.

getCollections: Recupera uma lista com as coleções disponíveis no catálogo. O acesso é feito através do caminho `/collections`.

O retorno desta operação é um arquivo no formato JSON (Listagem 3.5) com uma lista composta por `Collection Object` (Listagem 3.6).

Listagem 3.5 - Retorno da operação getCollections.

```

1 {
2   "collections": [Collection Object]
3 }
```

Fonte: Adaptado de [RADIANT EARTH FOUNDATION \(2019\)](#).

describeCollection: Recupera uma única coleção disponível no catálogo. O identificador único de uma coleção (`collection_id`) deve ser informado no caminho da URL. O retorno desta operação é um `Collection Object` (Listagem 3.6) representado por um arquivo no formato JSON. O acesso é feito através do caminho `/collections/collection_id`.

Listagem 3.6 - Collection Object.

```

1  {
2
3    "stac_version": "string"
4    "id": "string",
5    "description": "string",
6    "license": "string",
7    "extent": {
8      "spatial":{
9        "bbox": [[number]]
10     },
11     "temporal":{
12       "interval": [["string"|null]]
13     }
14   },
15   "links": [{
16     "href": "string",
17     "rel": "string"
18   }]
19 }

```

Fonte: Adaptado de [RADIANT EARTH FOUNDATION \(2019\)](#).

getFeatures: Recupera os itens de uma coleção, que deve ser informada através do seu identificador único (`collection_id`) na URL. Nesta operação é suportado o uso de filtros através da utilização de parâmetros na URL. A Tabela 3.10 apresenta os parâmetros que podem ser utilizados como filtro. O acesso desta operação é feito através do caminho `/collections/collection_id/items`.

Tabela 3.10 - Parâmetros de pesquisa STAC API.

Parâmetro	Tipo	Descrição
<code>bbox</code>	<code>[string]</code>	Retângulo envolvente [oeste, sul, leste, norte].
<code>datetime</code>	<code>string</code>	Data, data+tempo, ou intervalo ('/' separador), formatada de acordo com RFC 3339, seção 5.6.
<code>page</code>	<code>number</code>	Número de páginas de resultados. Padrão é 1.
<code>limit</code>	<code>number</code>	Número máximo de resultados por página. Padrão é 10.

Fonte: Adaptado de [RADIANT EARTH FOUNDATION \(2019\)](#).

O retorno é um arquivo no formato GeoJSON (Listagem 3.7). A lista de **Features** é composta por objetos STAC Item Object (Listagem 3.8).

Listagem 3.7 - Retorno da operação getFeatures.

```
1 {
2   "stac_version": "string",
3   "type": "FeatureCollection",
4   "features": [STAC Item Object],
5
6 }
```

Fonte: Adaptado de [RADIANT EARTH FOUNDATION \(2019\)](#).

Listagem 3.8 - STAC Item Object.

```
1 {
2   "stac_version": "string",
3   "id": "string",
4   "type": "Feature",
5   "geometry": GeoJSON Geometry Object|null,
6   "bbox": [number],
7   "properties": {
8     "datetime": "string"
9   },
10  "assets": {
11    "thumbnail":{
12      "href": "string"
13    }
14  },
15 }
```

Fonte: Adaptado de [RADIANT EARTH FOUNDATION \(2019\)](#).

getFeature: Recupera um item de uma coleção, que deve ser informada através do seu identificador único (`collection_id`) na URL. O identificador único (`item_id`) do item também deve ser informado na URL. O acesso desta operação é feito através do caminho `/collections/collection_id/items/item_id`.

O retorno desta operação é um STAC Item Object (Listagem 3.8).

getSearchSTAC: Recupera os itens de todo o catálogo. Nesta operação é suportado o uso de filtros através da utilização dos parâmetros utilizados na URL. Os parâmetros apresentados nas Tabelas 3.10 e 3.11 podem ser utilizados como filtro. O retorno é o mesmo da operação `getFeatures`, a diferença é que mais de uma coleção pode ser consultada em uma só requisição. O acesso desta operação é feito através do caminho `/search`.

Tabela 3.11 - Parâmetros de pesquisa operação `getSearchStac`.

Parâmetro	Tipo	Descrição
<code>collections</code>	[string]	Lista de strings separadas por vírgula contendo o identificador único de cada coleção a ser consultada
<code>ids</code>	[string]	Lista de strings separadas por vírgula contendo o identificador único de cada item. Ignora os outros filtros exceto <code>page</code> e <code>limit</code> .

Fonte: Adaptado de [RADIANT EARTH FOUNDATION \(2019\)](#).

postSearchSTAC: Recupera os itens de todo o catálogo. Esta operação pode receber filtros especificados através de um documento JSON enviado na requisição. Os parâmetros que podem ser utilizados são apresentados nas Tabelas 3.10 e 3.11. A principal diferença desta operação para a `getSearchStac` é a possibilidade de passar uma geometria GeoJSON no parâmetro `intersects` apresentado na Tabela 3.12. O acesso desta operação é feito através do caminho `/search`.

Tabela 3.12 - Parâmetros de pesquisa operação `postSearchStac`.

Parâmetro	Tipo	Descrição
<code>intersects</code>	GeoJSON Feature	Busca itens através da intersecção através de sua geometria e o GeoJSON informado

Fonte: Adaptado de [RADIANT EARTH FOUNDATION \(2019\)](#).

3.3 Implementação

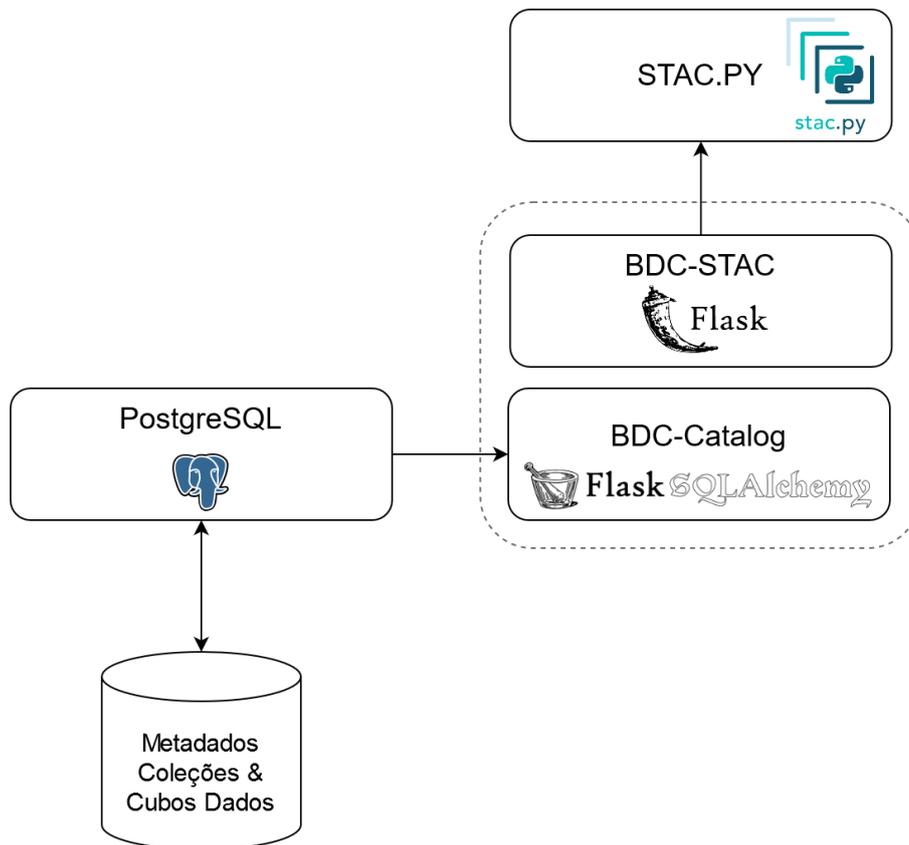
A Figura 3.4 apresenta os componentes de software desenvolvidos nesta dissertação e o cenário tecnológico envolvido em sua implementação. No nível mais baixo do diagrama, tem-se o repositório do catálogo de metadados das coleções de imagens e cubos de dados. Este repositório é mantido em um SGBD-R, o PostgreSQL¹. A implementação do modelo de dados do catálogo nesse SGBD utilizou colunas do tipo `JSONB`. Este tipo permite armazenar documentos texto no formato JSON e

¹<https://www.postgresql.org/>

manipulá-los em linguagem SQL de maneira simples e eficiente. A entidade **Asset** (diagrama da Figura 3.1), por exemplo, foi transformada em uma coluna da tabela **items** (diagrama da Figura 3.2) usando o tipo **JSONB**. Assim, as informações sobre cada *item* do STAC podem ser recuperadas de maneira fácil e já numa estrutura semelhante à do documento JSON que deve ser entregue pelo serviço STAC.

O componente **BDC-Catalog** é um pacote da linguagem Python², que utiliza o **Flask-SQLAlchemy**³ para implementar o modelo de dados do catálogo. Esse pacote fornece a camada de abstração de dados para implementação do serviço, mostrado como o componente **BDC-STAC** na Figura 3.4. O serviço **BDC-STAC** foi implementado com o uso do *framework* **Flask**, e fornece todas as rotas apresentadas na Seção 3.2.2.

Figura 3.4 - Componentes de software do serviço STAC do *Brazil Data Cube*.



Fonte: Produção do Autor.

²<https://www.python.org/>

³<https://flask-sqlalchemy.palletsprojects.com>

O componente `stac.py` é uma biblioteca Python que permite a construção de aplicações clientes do serviço STAC nessa linguagem.

Todo o código fonte dos produtos de software produzidos nesta dissertação encontram-se disponíveis na forma de código livre, hospedados em repositórios Git sob a organização *Brazil Data Cube* no GitHub⁴:

- `stac.py`: <https://github.com/brazil-data-cube/stac.py>.
- BDC-STAC: <https://github.com/brazil-data-cube/bdc-stac>.
- BDC-Catalog: <https://github.com/brazil-data-cube/bdc-catalog>.

Os pacotes utilizam o Travis CI⁵ para dar suporte à integração contínua. No caso do pacote `stac.py`, assim que uma nova *tag* é criada no repositório, versionando o software, o Travis CI realiza a construção do pacote e sua disponibilização no serviço de registro de pacotes Python denominado PYPPI⁶. Assim, automaticamente novas versões desse pacote ficam disponíveis para os usuários da linguagem Python, que podem instalar o pacote através do comando `pip`⁷. Além de realizar testes, essa integração do Travis CI permite a utilização do COVERALLS⁸ para apresentação de métricas sobre a cobertura dos testes dos pacotes.

Para a geração de documentação foi utilizada a biblioteca Sphinx⁹. Assim, os documentos do pacote podem ser disponibilizados através de serviços como o *Read the Docs*. Por exemplo, a documentação do pacote `stac.py` encontra-se disponível em: <https://stacpy.readthedocs.io/en/latest/>.

No próximo capítulo são apresentados os detalhes de uso do serviço STAC, desenvolvido nesta dissertação, para suporte às diversas aplicações que utilizam dados do projeto *Brazil Data Cube*. Também serão apresentados estudos de caso que utilizam a biblioteca `stac.py` para construção de aplicações.

⁴<https://github.com/brazil-data-cube>

⁵<https://travis-ci.com/>

⁶<https://pypi.org/project/stac.py/>

⁷<https://pip.pypa.io/>

⁸<https://coveralls.io/>

⁹<https://www.sphinx-doc.org/>

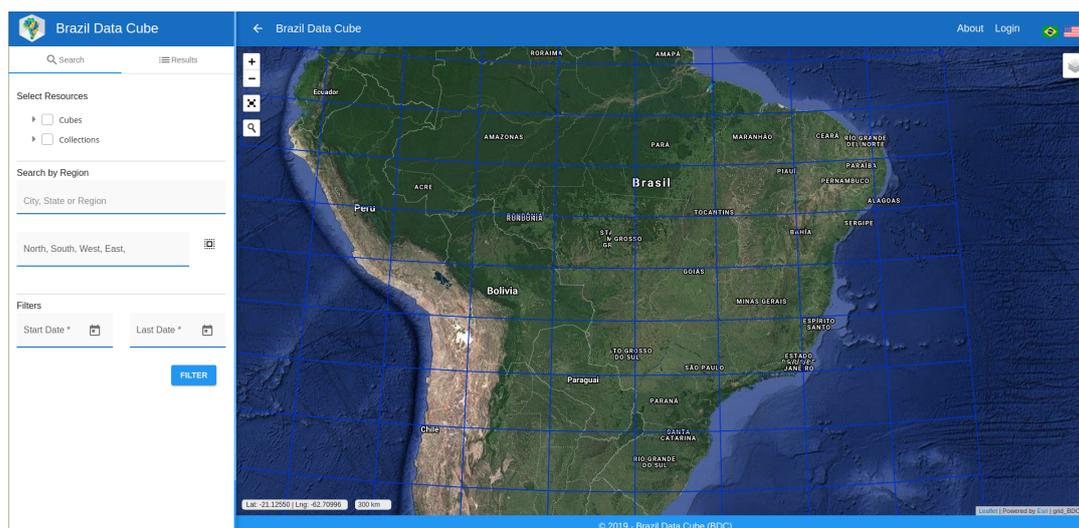
4 ESTUDOS DE USO DA INTERFACE STAC

Neste capítulo são apresentados estudos de caso onde o serviço STAC, descrito no Capítulo 3, foi empregado para acesso aos dados do projeto *Brazil Data Cube*(BDC).

4.1 Portal de acesso e visualização das imagens do projeto BDC

O Portal do *Brazil Data Cube*, mostrado na Figura 4.1, é a interface gráfica web, de acesso público, para que os usuários explorem e visualizem as coleções e os cubos de dados do projeto. O portal usa exaustivamente a API de acesso ao serviço STAC, desenvolvido nessa dissertação, para acessar todo o conjunto de dados disponível no projeto, a começar pela lista de coleções, e cubos de dados, que estão disponíveis para consulta e visualização.

Figura 4.1 - Portal Web do *Brazil Data Cube*.

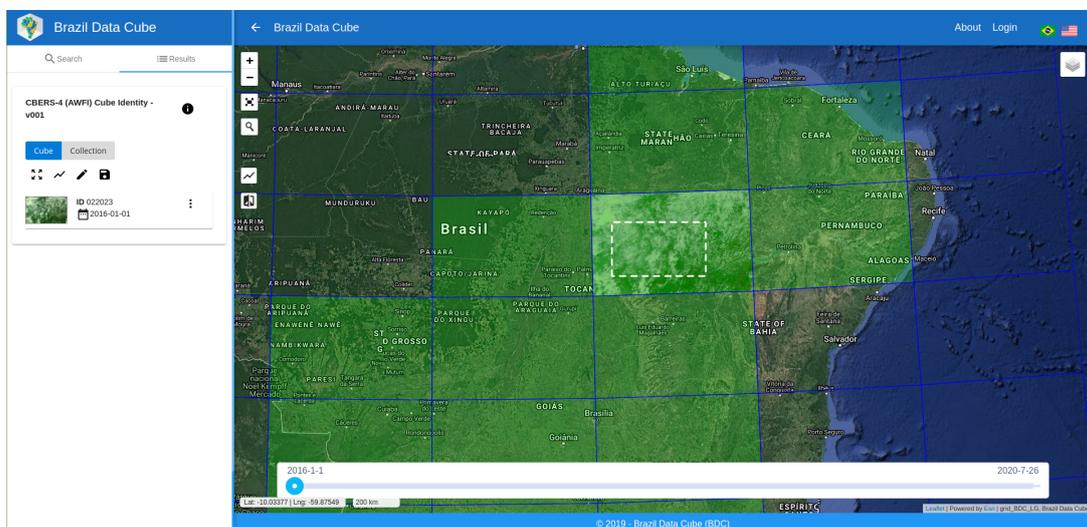


Fonte: INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS (2020a)

O componente de filtro (no painel à esquerda do mapa na Figura 4.1) permite definir critérios espaciais e temporais para busca de itens. Este componente utiliza a API do STAC para recuperar os itens da coleção que satisfaçam os critérios informados pelo usuário. O serviço retorna os metadados e a aplicação do portal apresenta a miniatura do item e seus metadados, como mostrado na parte esquerda da Figura 4.2. Como os itens das coleções possuem informações temporais, é possível no portal a utilização de um componente para exibição das imagens ao longo de uma linha do tempo. O componente na parte inferior do mapa permite ao usuário con-

trolar a data de visualização das imagens. A extensão BDC do STAC, que foi criada nessa dissertação, é utilizada para fornecer os metadados específicos do projeto, que detalham créditos autorais dos produtos de dados, citação, licença, sistemas utilizados no processamento das imagens, ligações com outros tipos de recursos, como artigos científicos e vídeos sobre os produtos de dados. A opção de download de dados, oferecida pelo portal, é facilmente implementada pois a especificação STAC prevê o fornecimento das URLs dos *assets* dos itens descobertos e acessados via a especificação STAC.

Figura 4.2 - Portal Web do *Brazil Data Cube*.



Fonte: INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS (2020a)

4.2 Clientes para linguagens de *scripting*

Atualmente linguagens interpretadas têm se tornado muito populares devido a diversos fatores, como por exemplo o fato de possuírem uma sintaxe mais simples e clara, serem acessíveis e oferecerem documentação e suporte de comunidades de usuários. Para aplicações científicas e que necessitam de exaustivas análises de dados, essas linguagens são particularmente atrativas pela integração com ambientes de simulação e visualização de dados. Além disso, já é possível encontrar uma abundância de bibliotecas que fornecem funções, estruturas de dados e implementações de operações matemáticas flexíveis e eficientes. Entre as linguagens de *scripting* mais populares para aplicações científicas e de ciência de dados estão as linguagens Python (NAGPAL; GABRANI, 2019) e R (R CORE TEAM, 2020). Visando responder

a essa tendência e aumentar as possibilidades de acesso aos dados do *Brazil Data Cube*, foram desenvolvidas interfaces cliente para o serviço STAC a serem usadas em aplicações escritas nessas linguagens.

4.2.1 Cliente Python

A biblioteca `stac.py` permite criar aplicações clientes do serviço STAC. A Listagem 4.1 exemplifica o seu uso para consultar o catálogo de dados, recuperar coleções e navegar entre seus itens, e consumir os *assets* desses itens. Nesse pequeno, mas representativo, exemplo, pode-se notar uma interface de alto nível de abstração para a descoberta e acesso aos dados.

Listagem 4.1 - Uso da biblioteca `stac.py`.

```
1 import stac
2
3 service = stac.STAC("http://brazildatacube.dpi.inpe.br/stac/")
4
5 print(service.catalog)
6 # OUTPUT:
7 # ['LC8_30-1', 'L5SR-1', 'S2T0A-1', 'HLS.S30-1', 'MYD13Q1-1', 'HLS.L30-1', 'LC8DN-1',
8 ↪ 'LC8SR-1', 'MOD13Q1-1', 'S2_MSI_L2_SR_LASRC-1', 'S2_10-1', 'CB4_64-1',
9 ↪ 'S2_10_16D_STK-1', 'LC8_30_16D_STK-1', 'CB4_64_16D_STK-1']
10
11 collection = service.collection("CB4_64_16D_STK-1")
12
13 items = collection.get_items(filter={"bbox": "-54.40429,-16.17247,-47.32910,-9.75237",
14                                     "datetime": "2017-01-01/2017-12-31",
15                                     "limit": 2})
16
17 for item in items:
18     arr = item.read("NDVI")
19     print(f"ID: {item.id}\nDatetime: {item.properties['datetime']}\nShape: {arr.shape}")
20 # OUTPUT:
21 # ID: CB4_64_16D_STK_v001_022023_2017-12-19_2017-12-31
22 # Datetime: 2017-12-19T00:00:00
23 # Shape: (6865, 10504)
24 # ID: CB4_64_16D_STK_v001_022024_2017-12-19_2017-12-31
25 # Datetime: 2017-12-19T00:00:00
26 # Shape: (6865, 10504)
```

Fonte: Produção do Autor.

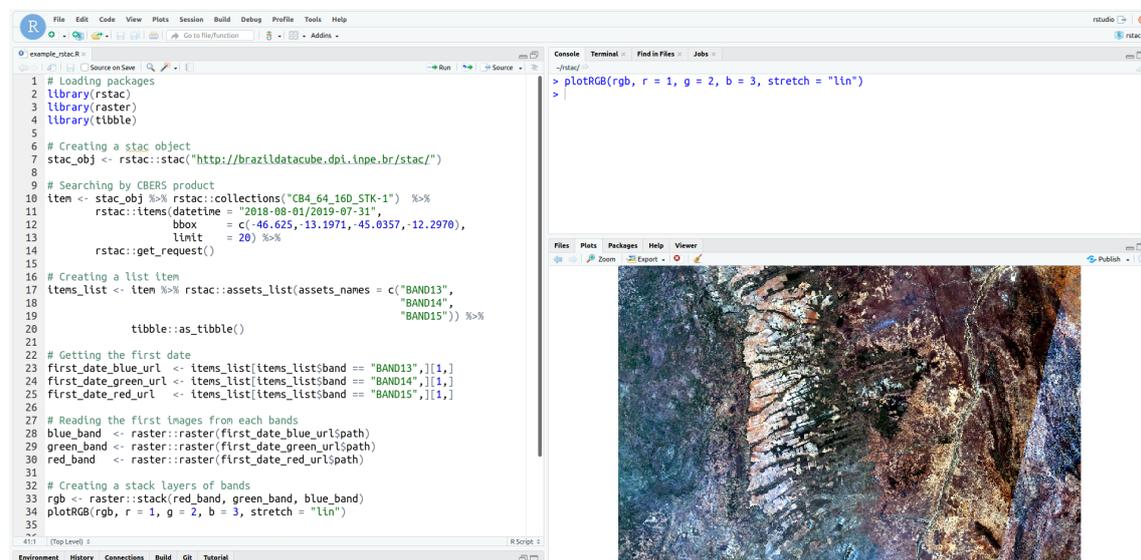
Na linha 01, as funcionalidades da biblioteca `stac.py` são importadas para o espaço de nomes denominado `stac`. Na linha 03, é criado um objeto capaz de realizar a ligação com o serviço STAC, no endereço fornecido (`http://brazildatacube.dpi.inpe.br/stac/`). Na linha 05 é mostrado uma lista com as coleções disponíveis no serviço STAC. Na linha 09 é feita a chamada ao método `collection`,

que como parâmetro recebe o nome da coleção CB4_64_16D_STK-1, onde um objeto `Collection` será retornado. Nesse objeto retornado, é possível acessar todos os metadados relacionados à coleção informada. A linha 11 mostra o uso do método `get_items` disponível em um objeto `Collection`. Esse método permite que o cliente faça uma consulta com filtros e recupere um subconjunto de itens da coleção. As linhas 15, 16 e 17 mostram como realizar a travessia na lista de itens e a leitura do *asset* NDVI.

4.2.2 Cliente R

A linguagem R é provavelmente tão usada para aplicações científicas, em ciência de dados e, principalmente, estatísticas, quanto Python. Para demonstrar a flexibilidade do componente de catálogo criado nessa dissertação, foi desenvolvido também um cliente R para o serviço STAC, chamado "rstac". O cliente permite a consulta ao catálogo e a recuperação das coleções, a navegação sobre seus itens até chegar aos *assets*. Esses são lidos e manipulados em estruturas e funções da linguagem R. Um exemplo de uso desse cliente é mostrado na Figura 4.3, que mostra a visualização de uma imagem do catálogo.

Figura 4.3 - Código R, usando o RSTAC rodando no ambiente RStudio.



Fonte: Produção do Autor.

4.2.3 IPython/Jupyter

O IPython é um ambiente projetado para Computação Interativa originalmente criado para a linguagem Python (PÉREZ; GRANGER, 2007). Atualmente, este ambiente suporta diversas outras linguagens de programação, como R e Julia. Basicamente, esse ambiente fornece um terminal interativo com capacidade de destacar símbolos da linguagem Python (ou de outras linguagens), possui recursos avançados de auto-completar que inclui introspecção do objeto em questão para determinar atributos e métodos, manutenção do histórico de comandos, além de forte suporte para visualização interativa. A biblioteca `stac.py`, apresentada na Seção 3.3, pode ser usada nesse ambiente.

No entanto, diversos usuários dos dados do Projeto *Brazil Data Cube* têm utilizado um ambiente mais amigável, conhecido como Jupyter (KLUYVER et al., 2016), derivado da tecnologia IPython. Esse ambiente, conhecido popularmente por *Jupyter Notebooks*, oferece uma forma poderosa de construir documentos envolvendo texto livre, textos formatados em *Markdown*, com uso de HTML e expressões matemáticas em \LaTeX , elementos gráficos resultantes de plotagem de figuras dentro da própria página, além de diversos outros tipos de mídias. A biblioteca `stac.py` implementa diversas facilidades para integração com esse ambiente, principalmente em relação à apresentação dos objetos dessa biblioteca nas células dos *notebooks*.

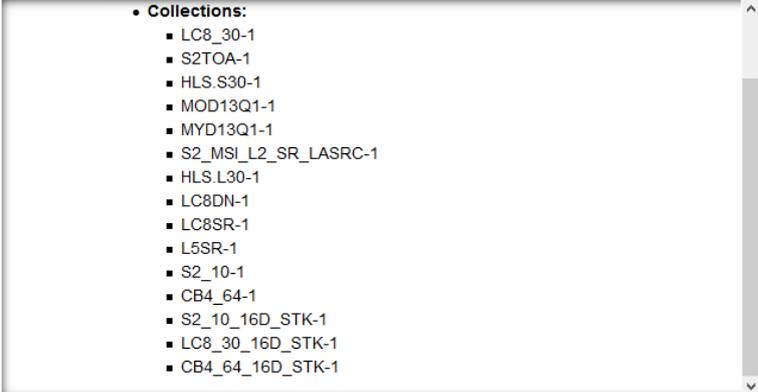
As Figuras 4.4, 4.5, 4.6 e 4.7 ilustram partes de um *notebook* que faz uso do cliente Python no ambiente Jupyter. A Figura 4.4 apresenta a importação da biblioteca cliente, a associação ao endereço do serviço STAC, desenvolvido nessa dissertação, e a apresentação das coleções contidas no catálogo desse serviço. Repare que a última célula faz uma apresentação das coleções na forma de uma lista HTML, ao invés de simplesmente apresentar o documento JSON com essas informações.

Figura 4.4 - Listando as coleções disponíveis na STAC API do BDC usando o stac.py no IPython/Jupyter.

```
In [1]: import stac

In [2]: service = stac.STAC('http://brazildatacube.dpi.inpe.br/stac/')

In [3]: service
```



- **Collections:**
 - LC8_30-1
 - S2TOA-1
 - HLS.S30-1
 - MOD13Q1-1
 - MYD13Q1-1
 - S2_MSI_L2_SR_LASRC-1
 - HLS.L30-1
 - LC8DN-1
 - LC8SR-1
 - L5SR-1
 - S2_10-1
 - CB4_64-1
 - S2_10_16D_STK-1
 - LC8_30_16D_STK-1
 - CB4_64_16D_STK-1

Fonte: Produção do Autor.

A Figura 4.5 mostra a continuação do *notebook*, com o acesso aos metadados da coleção *CB4_64_16D_STK-1*. O objeto *collection* contém os metadados retornados pelo serviço STAC. Com a integração da biblioteca cliente ao ambiente Jupyter, esse objeto é apresentado de maneira gráfica como um documento HTML com textos em negrito, tabelas e listas.

Figura 4.5 - Descrevendo o cubo de dados CB4_64_16D_STK-1 usando o stac.py no IPython/Jupyter.

```
In [4]: collection = service.collection('CB4_64_16D_STK-1')
In [5]: collection
```

Out[5]: **Collection:** CB4_64_16D_STK-1
Description: CBERS 4 version 1
Start date: 2016-01-01T00:00:00 **End date:** 2020-07-26T00:00:00
STAC Version: 0.9.0

Bands

name	common_name	min	max	nodata	scale	data_type
BAND13	blue	0.0	10000.0	-9999.0	0.0001	int16
BAND14	green	0.0	10000.0	-9999.0	0.0001	int16
BAND15	red	0.0	10000.0	-9999.0	0.0001	int16
BAND16	nir	0.0	10000.0	-9999.0	0.0001	int16
CLEAROB	ClearOb	0.0	255.0	0.0	0.0001	uint8
CMASK	quality	0.0	4.0	255.0	0.0001	uint8
EVI	evi	0.0	10000.0	-9999.0	0.0001	int16
NDVI	ndvi	0.0	10000.0	-9999.0	0.0001	int16
PROVENANCE	Provenance	1.0	366.0	-1.0	0.0001	int16
TOTALOB	TotalOb	0.0	255.0	0.0	0.0001	uint8

Extent

xmin	ymin	xmax	ymax
-61.002123	-26.947548	-35.631935	0.949458

Timeline

- 2016-01-01 ^
- 2016-01-17
- 2016-02-02
- 2016-02-18
- 2016-03-05
- 2016-03-21
- 2016-04-06
- 2016-04-22
- 2016-05-08

Fonte: Produção do Autor.

A Figura 4.6 mostra o cliente recuperando os itens de uma coleção. Novamente, a integração da biblioteca com ambiente Jupyter proporciona a exibição dos itens na forma gráfica. Ao expandir um dos itens, uma apresentação semelhante à da Figura 4.7 é exibida na célula.

Figura 4.6 - Listando as dez primeiras imagens disponíveis do cubo de dados CB4_64_16D_STK-1 stac.py no IPython/Jupyter.

```
In [6]: items = collection.get_items()

In [7]: items

Out[7]: Matched: 1785 Returned: 10 Limit: 10
```

CB4_64_16D_STK_v001_021027_2020-07-11_2020-07-26	+
CB4_64_16D_STK_v001_022021_2020-07-11_2020-07-26	+
CB4_64_16D_STK_v001_022022_2020-07-11_2020-07-26	+
CB4_64_16D_STK_v001_022023_2020-07-11_2020-07-26	+
CB4_64_16D_STK_v001_022024_2020-07-11_2020-07-26	+

Fonte: Produção do Autor.

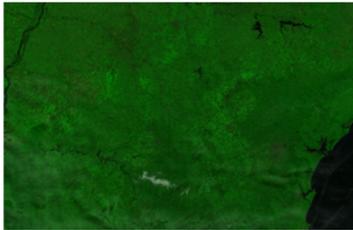
A Figura 4.7 apresenta como os usuários conseguem visualizar os objetos representando metadados de um item de maneira gráfica, incluindo a miniatura associada ao item.

Figura 4.7 - Descrevendo o item CB4_64_16D_STK_v001_021027_2020-07-11_-2020-07-26 usando o stac.py no IPython/Jupyter.

```
In [8]: items.features[0]
```

```
Out[8]: CB4_64_16D_STK_v001_021027_2020-07-11_2020-07-26
Collection: CB4_64_16D_STK-1
Datetime: 2020-07-11T00:00:00
STAC Version: 0.9.0
```

Thumbnail:



Bands

name	common_name	min	max	nodata	scale	data_type
BAND13	blue	0.0	10000.0	-9999.0	0.0001	int16
BAND14	green	0.0	10000.0	-9999.0	0.0001	int16
BAND15	red	0.0	10000.0	-9999.0	0.0001	int16
BAND16	nir	0.0	10000.0	-9999.0	0.0001	int16
CLEAROB	ClearOb	0.0	255.0	0.0	0.0001	uint8
CMASK	quality	0.0	4.0	255.0	0.0001	uint8
EVI	evi	0.0	10000.0	-9999.0	0.0001	int16
NDVI	ndvi	0.0	10000.0	-9999.0	0.0001	int16
PROVENANCE	Provenance	1.0	366.0	-1.0	0.0001	int16
TOTALOB	TotalOb	0.0	255.0	0.0	0.0001	uint8

BBox

xmin	ymin	xmax	ymax
-54.368967	-26.947548	-47.735397	-22.873503

Fonte: Produção do Autor.

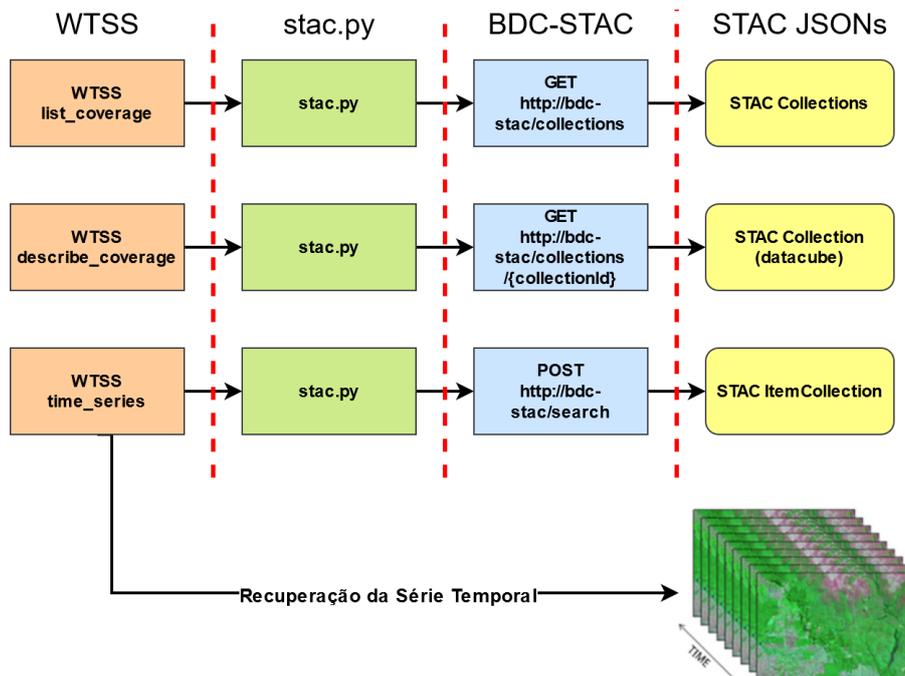
Este estudo de caso reforça a flexibilidade do serviço STAC e da sua facilidade de uso através de clientes em linguagens de *scripting*. Além disso, sua integração em ambientes de computação interativa, como o Jupyter, proporciona aos usuários do catálogo do projeto *Brazil Data Cube* novas formas de acesso. Também é importante ressaltar que esse mesmo exemplo poderia ser realizado utilizando-se do cliente R.¹

¹No GitHub do projeto BDC, existe uma galeria de notebooks criada para demonstrar esse tipo de uso dos clientes STAC em ambiente de computação interativa: <https://github.com/brazil-data-cube/jupyter-gallery>.

4.3 Web Time Series Service (WTSS)

O Web Time Series Service, ou WTSS, é um serviço voltado para recuperação de séries temporais a partir de imagens de sensoriamento remoto (VINHAS et al., 2017). A primeira versão deste serviço, descrita em Queiroz et al. (2015), utilizava uma arquitetura baseada em um SGBD Matricial. No projeto BDC, com a mudança da forma de organização dos dados matriciais e utilização do formato COG, foi necessário desenvolver uma nova versão desse serviço. Essa nova versão, mostrada no diagrama da Figura 4.8, utiliza o serviço STAC, desenvolvido nesta dissertação, para acesso aos metadados dos cubos de dados de onde o serviço realiza a extração das séries temporais.

Figura 4.8 - Uso do BDC-STAC no WTSS.



Fonte: Produção do Autor.

A operação `list_coverages` do WTSS oferece aos seus clientes a lista de produtos de dados, denominados de *coverage*, para consulta às séries temporais. No diagrama da Figura 4.8, pode-se observar que essa operação do WTSS utiliza o cliente Python `stac.py` para obter essas informações a partir do serviço BDC-STAC.

De maneira similar, a operação `describe_coverage` do WTSS, que fornece os me-

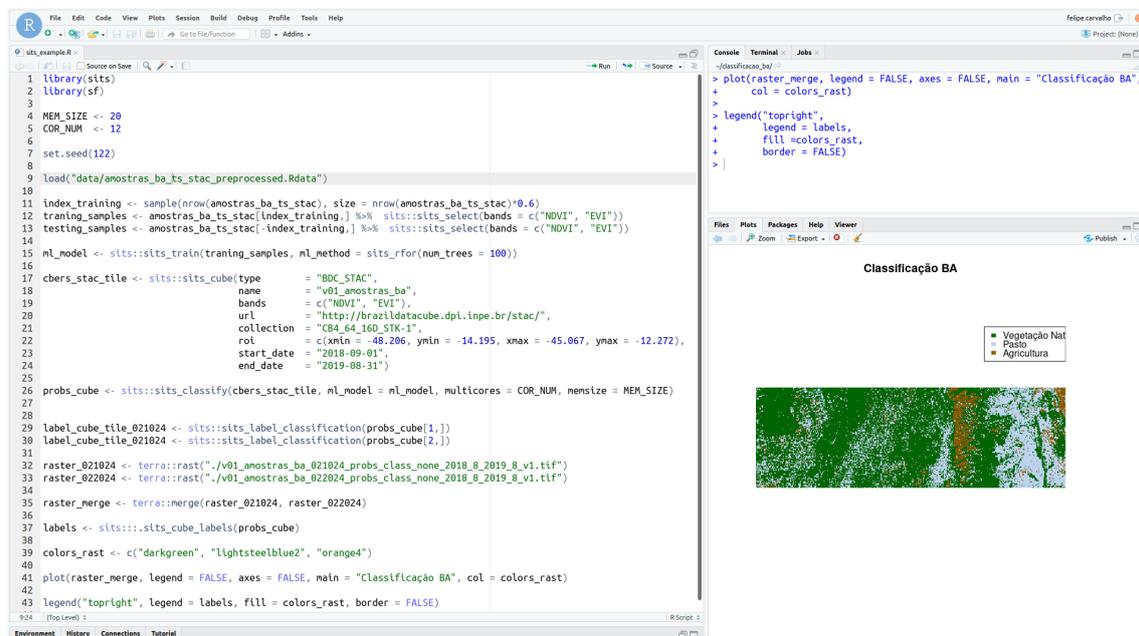
tadados de uma determinada *coverage*, utiliza o serviço BDC-STAC para entregar esses metadados.

Por último, a operação `time_series` do WTSS, que dada uma localização e um período de tempo recupera a série temporal para esta localização, utiliza o BDC-STAC para determinar todos os *assets* necessários à operação e, através do protocolo HTTP obtém os dados de fato a partir dos arquivos no formato COG.

4.4 *Satellite Image Time Series Analysis for Earth Observation Data Cubes* (SITS)

O *Satellite Image Time Series Analysis for Earth Observation Data Cubes* (SITS)² é um pacote em R que permite a análise, visualização e classificação de séries temporais de imagens de satélite. Através do cliente R denominado `rstac` (mostrado na Seção 4.2.2), o SITS é capaz de acessar o serviço BDC-STAC e descobrir os cubos de dados disponíveis no projeto BDC para análise. A Figura 4.9 mostra um *script* que realiza a consulta a um cubo de dados disponível através do BDC-STAC, e a classificação de desse cubo de dados usando métodos do SITS.

Figura 4.9 - Classificação de imagens do BDC usando o SITS.



Fonte: Produção do Autor.

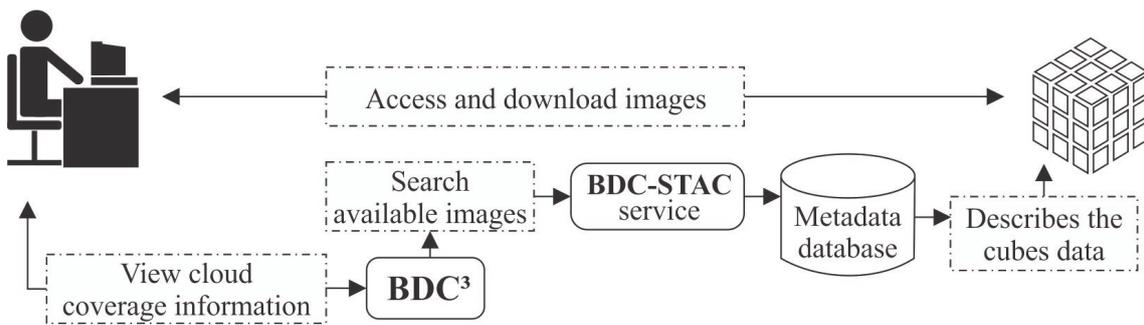
²<https://github.com/e-sensing/sits>

4.5 Brazil Data CubeCloud Cover Viewer - BDC³ Viewer

O visualizador BDC³ (LUCENA et al., 2020) é uma aplicação que propõe uma extensão das possibilidades de consulta ao BDC-STAC para aquisição e visualização de informações específicas sobre a cobertura de nuvens, no tempo, em uma determinada região. Esse trabalho, que ainda não está completo, será capaz de responder a perguntas como “qual ano teve a maior cobertura de nuvens em dezembro, no estado do Acre?”.

Para isso, o visualizador consome as informações sobre a cobertura de nuvens, disponível no BDC-STAC e customiza uma visualização específica. O diagrama da Figura 4.10 mostra como esta aplicação realiza sua interface com o BDC-STAC.

Figura 4.10 - Arquitetura BDC³.



Fonte: Lucena et al. (2020).

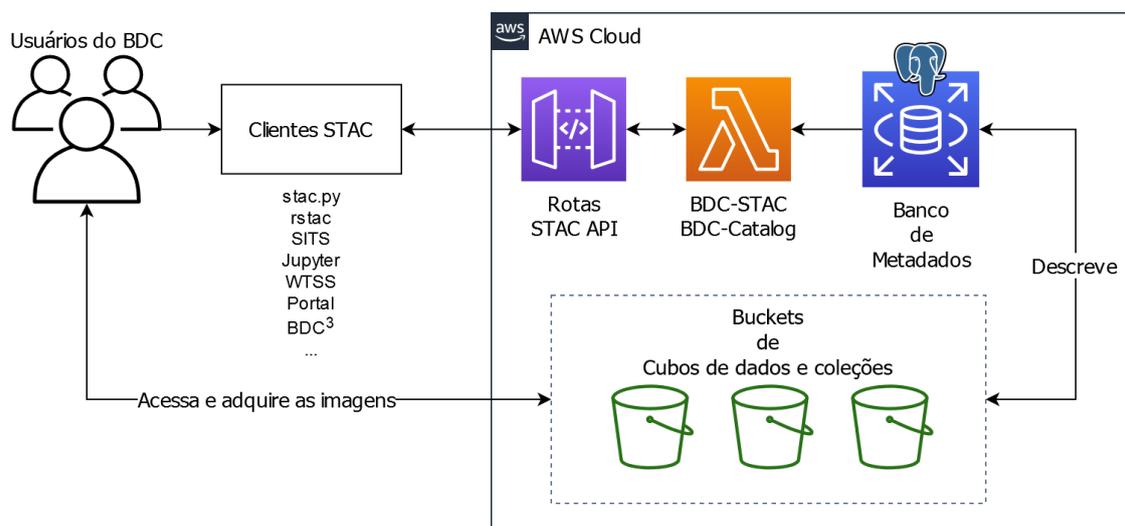
4.6 BDC-STAC na nuvem

A *Amazon Web Services* (AWS) é um produto da Amazon que oferece diversos tipos de soluções de computação na nuvem, entre eles, a *AWS Lambda*, *AWS API Gateway* e a *AWS RDS* (*Relational Database Service*). A *AWS Lambda* é uma solução *serverless* que permite a execução de código sem o provisionamento de um servidor. Já a *AWS API Gateway* é uma solução que cria caminhos em um endereço web para direcionar requisições às funções criadas na *AWS Lambda*. O *RDS* é uma solução na nuvem para sistemas gerenciadores de bancos de dados, como o PostgreSQL.

Neste trabalho foi realizada a integração do serviço BDC-STAC no ambiente da

AWS. Para isso, foi empregada a biblioteca *Zappa*³, capaz de realizar a implantação na AWS de aplicações baseadas na interface *Web Server Gateway Interface* (WSGI) através do serviço *AWS Lambda*. A biblioteca *Flask* usada na implementação do serviço BDC-STAC (diagrama da Figura 3.4), dessa dissertação, implementa essa interface WSGI e, logo, permite o uso do *Zappa*. Com isso, a implementação do BDC-STAC na nuvem da AWS, segue a arquitetura apresentada na Figura 4.11.

Figura 4.11 - Arquitetura BDC-STAC na AWS.



Fonte: Produção do Autor.

A biblioteca *Zappa* cria as rotas na *AWS API Gateway* de acordo com as rotas equivalentes declaradas na aplicação *Flask* (BDC-STAC). Os pacotes Python que encapsulam o BDC-STAC e o BDC-Catalog são utilizados na função *lambda* criadas pelo *Zappa*. Esta função é que irá realizar a computação necessária para atender as requisições dos clientes. O serviço *RDS* utiliza uma instância do servidor PostgreSQL com o modelo de metadados definido na Seção 3.1. No entanto, na nuvem da AWS, os *assets* são associados a recursos armazenados em *buckets* do serviço *S3* ao invés de arquivos armazenados em servidores de dados convencionais.

³<https://github.com/Miserlou/Zappa>

5 CONSIDERAÇÕES FINAIS

O contexto atual de alta disponibilidade de imagens de sensoriamento remoto tem impulsionado a criação de cubos de dados de imagens, necessários para a efetiva extração de informações considerando séries temporais de imagens. Para a implementação dessas infraestruturas diversos componentes têm sido propostos a partir de paradigmas já existentes, como os serviços web e as especificações de serviços e metadados geográficos. Um importante componente dessa infraestrutura é o que fornece as funcionalidades de catalogação de dados e metadados visando permitir a descoberta e acesso aos diferentes tipos de dados da infraestrutura de cubo de dados.

Essa dissertação estudou e propôs o padrão SpatioTemporal Asset Catalog (STAC) para catalogação dos metadados das imagens dos cubos de dados do projeto *Brazil Data Cube* (BDC). Como resultado desse trabalho, concluiu-se que o STAC se mostrou efetivo para implementar a componente de catalogação e acesso aos dados do projeto, especialmente pela facilidade de uso em um ambiente de serviços.

O STAC é uma especificação da comunidade e que está em amadurecimento. Ela define uma quantidade mínima de campos de metadados necessários para todas as aplicações. No entanto, essas podem não ser suficientes, por isso, a especificação STAC prevê a possibilidade de que outros metadados sejam incluídos, através de um mecanismo chamado de extensões. Nessa dissertação foi criada uma extensão específica para acomodar os metadados da arquitetura do projeto BDC, chamada BDC-STAC. Dessa forma, essa dissertação contribui para a comunidade de software livre ao implementar, testar em um ambiente real de uso, e propor melhorias a uma especificação de grande interesse para a comunidade de sensoriamento remoto e geoinformática.

O serviço *BDC-STAC*, ou seja, o serviço STAC para o BDC foi implementado e avaliado, e está suportando a arquitetura de serviços do projeto. Nessa dissertação foram apresentadas sete casos de uso em aplicações que requerem acesso ao catálogo de dados, e que são representativas da flexibilidade do serviço proposto e desenvolvido.

Todos os códigos fontes do serviço desenvolvido, bem como dos clientes, encontram-se disponíveis como código fonte aberto e livre de licença no repositório de código GitHub disponível em <https://github.com/brazil-data-cube/bdc-stac> e <https://github.com/brazil-data-cube/stac.py>.

Como trabalhos futuros, serão investigadas novas formas de explorar o serviço STAC

usando tecnologias de computação em nuvem, bem como compreender os modelos de custo desse tipo de infraestrutura, de maneira a oferecer soluções com maior escalabilidade com bom compromisso de custo.

REFERÊNCIAS BIBLIOGRÁFICAS

- AALDERS, H. J. An introduction to metadata for geographic information. In: MOELLERING, H.; AALDERS, H.; CRANE, A. (Ed.). **World spatial metadata standards**. [S.l.]: Elsevier, 2005. p. 3–27. 6
- BAI, Y.; DI, L. Providing access to satellite imagery through OGC catalog service interfaces in support of the Global Earth Observation System of Systems. **Computers & Geosciences**, v. 37, n. 4, p. 435–443, abr. 2011. ISSN 00983004. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S0098300410003195>>. 8
- BRODEUR, J.; COETZEE, S.; DANKO, D.; GARCIA, S.; HJELMAGER, J. Geographic information metadata—an outlook from the international standardization perspective. **ISPRS International Journal of Geo-Information**, v. 8, n. 6, p. 280, 2019. 8, 9
- BUTLER, H.; DALY, M.; DOYLE, A.; GILLIES, S.; SCHAUB, T.; SCHAUB, T. **The GeoJSON Format**. RFC Editor, ago. 2016. RFC 7946. (Request for Comments, 7946). Disponível em: <<https://rfc-editor.org/rfc/rfc7946.txt>>. 10
- COMITÊ DE ESTRUTURAÇÃO DE METADADOS GEOESPACIAIS - CEMG. **Perfil de metadados geoespaciais do Brasil: perfil MGB**. [S.l.]: COMCAR, 2009. 6
- COMMITTEE ON EARTH OBSERVATION. **CEOS analysis ready data**. nov. 2020. Disponível em: <<http://ceos.org/ard/>>. Acesso em: 16 nov. 2020. 15
- COSTA, W.; FONSECA, L.; KORTING, T. Classifying grasslands and cultivated pastures in the brazilian cerrado using support vector machines, multilayer perceptrons and autoencoders. In: INTERNATIONAL CONFERENCE IN MACHINE LEARNING, 11., 2015, Hamburg. **Proceedings...** Hamburg, 2015. 1
- DWYER, J. L.; ROY, D. P.; SAUER, B.; JENKERSON, C. B.; ZHANG, H. K.; LYMBURNER, L. Analysis ready data: enabling analysis of the landsat archive. **Remote Sensing**, v. 10, n. 9, p. 1363, 2018. 15
- EUROPEAN SPACE AGENCY. **Sentinel 2 mission**. nov. 2020. Disponível em: <<https://sentinel.esa.int/web/sentinel/missions/sentinel-2>>. Acesso em: 19 nov. 2020. 17

FERREIRA, K. R.; QUEIROZ, G. R.; VINHAS, L.; MARUJO, R. F. B.; SIMOES, R. E. O.; PICOLI, M. C. A.; CAMARA, G.; CARTAXO, R.; GOMES, V. C. F.; SANTOS, L. A.; SANCHEZ, A. H.; ARCANJO, J. S.; FRONZA, J. G.; NORONHA, C. A.; COSTA, R. W.; ZAGLIA, M. C.; ZIOTI, F.; KORTING, T. S.; SOARES, A. R.; CHAVES, M. E. D.; FONSECA, L. M. G. Earth observation data cubes for Brazil: requirements, methodology and products. **Remote Sensing**, v. 12, n. 24, 2020. 2, 17

GIULIANI, G.; CHATENOUX, B.; BONO, A. D.; RODILA, D.; RICHARD, J.-P.; ALLENBACH, K.; DAO, H.; PEDUZZI, P. Building an Earth Observations Data Cube: lessons learned from the Swiss Data Cube (SDC) on generating Analysis Ready Data (ARD). **Big Earth Data**, v. 1, n. 1-2, p. 100–117, dez. 2017. ISSN 2096-4471, 2574-5417. Disponível em: <<https://www.tandfonline.com/doi/full/10.1080/20964471.2017.1398903>>. 2

GOMES, V. C. F. et al. An overview of platforms for big earth observation data management and analysis. **Remote Sensing**, v. 12, n. 8, e1253, 2020. 15

GUO, H.-D.; ZHANG, L.; ZHU, L.-W. Earth observation big data for climate change research. **Advances in Climate Change Research**, v. 6, n. 2, p. 108–117, 2015. 1

GUPTILL, S. C. Metadata and data catalogues. **Geographical Information Systems**, v. 2, p. 677–692, 1999. 7

HASHEM, I. A. T.; YAQOOB, I.; ANUAR, N. B.; MOKHTAR, S.; GANI, A.; KHAN, S. U. The rise of “big data” on cloud computing: review and open research issues. **Information Systems**, v. 47, p. 98–115, 2015. 16

INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS. **Brazil Data Cube web portal**. 2020. Disponível em: <<https://brazildatacube.dpi.inpe.br/portal/>>. Acesso em: 26 jul. 2020. 37, 38

_____. **CBERS**. nov. 2020. Disponível em: <<http://www.cbbers.inpe.br/>>. Acesso em: 19 nov. 2020. 17

KHORRAM, S.; WIELE, C. F. van der; KOCH, F. H.; NELSON, S. A. C.; POTTS, M. D. **Principles of applied remote sensing**. New York: Springer International Publishing, 2016. 21–67 p. 3

KLUYVER, T.; RAGAN-KELLEY, B.; PÉREZ, F.; GRANGER, B.;
BUSSONNIER, M.; FREDERIC, J.; KELLEY, K.; HAMRICK, J.; GROUT, J.;
CORLAY, S.; IVANOV, P.; AVILA, D.; ABDALLA, S.; WILLING, C.; TEAM, J.
development. Jupyter notebooks - a publishing format for reproducible
computational workflows. In: LOIZIDES F.; SCMIDT, B. E. (Ed.). **Positioning
and power in academic publishing: players, agents and agendas**. [S.l.]:
IOS Press, 2016. p. 87–90. 41

KOPP, S.; BECKER, P.; DOSHI, A.; WRIGHT, D. J.; ZHANG, K.; XU, H.
Achieving the full vision of earth observation data cubes. **Data**, v. 4, n. 3, p. 94,
2019. 14

LEWIS, A.; OLIVER, S.; LYMBURNER, L.; EVANS, B.; WYBORN, L.;
MUELLER, N.; RAEVKSI, G.; HOOKE, J.; WOODCOCK, R.; SIXSMITH, J.;
WU, W.; TAN, P.; LI, F.; KILLOUGH, B.; MINCHIN, S.; ROBERTS, D.;
AYERS, D.; BALA, B.; DWYER, J.; DEKKER, A.; DHU, T.; HICKS, A.; IP, A.;
PURSS, M.; RICHARDS, C.; SAGAR, S.; TRENHAM, C.; WANG, P.; WANG,
L.-W. The Australian Geoscience Data Cube — Foundations and lessons learned.
Remote Sensing of Environment, v. 202, p. 276–292, dez. 2017. ISSN
00344257. Disponível em:
<<https://linkinghub.elsevier.com/retrieve/pii/S0034425717301086>>. 2

LI, S. et al. Geospatial big data handling theory and methods: a review and
research challenges. **ISPRS journal of Photogrammetry and Remote
Sensing**, v. 115, p. 119–133, 2016. 16

LI, Z.; TANG, W.; HUANG, Q.; SHOOK, E.; GUAN, Q. Introduction to big data
domputing for geospatial applications. **ISPRS International Journal of
Geo-Information**, v. 9, n. 8, p. 487, ago. 2020. ISSN 2220-9964. Disponível em:
<<https://www.mdpi.com/2220-9964/9/8/487>>. 16

LUCENA, F. R. d. S. M.; ESCOBAR-SILVA, E. V.; MARUJO, R. d. F. B.;
ZAGLIA, M. C.; VINHAS, L.; FERREIRA, K. R.; QUEIROZ, G. R. d. Brazil
data cube cloud coverage (bdc³) viewer. In: BRAZILIAN SYMPOSIUM ON
GEOINFORMATICS - GEOINFO, 21., 2020. **Proceedings...** São José dos
Campos, 2020. p. 234–239. 48

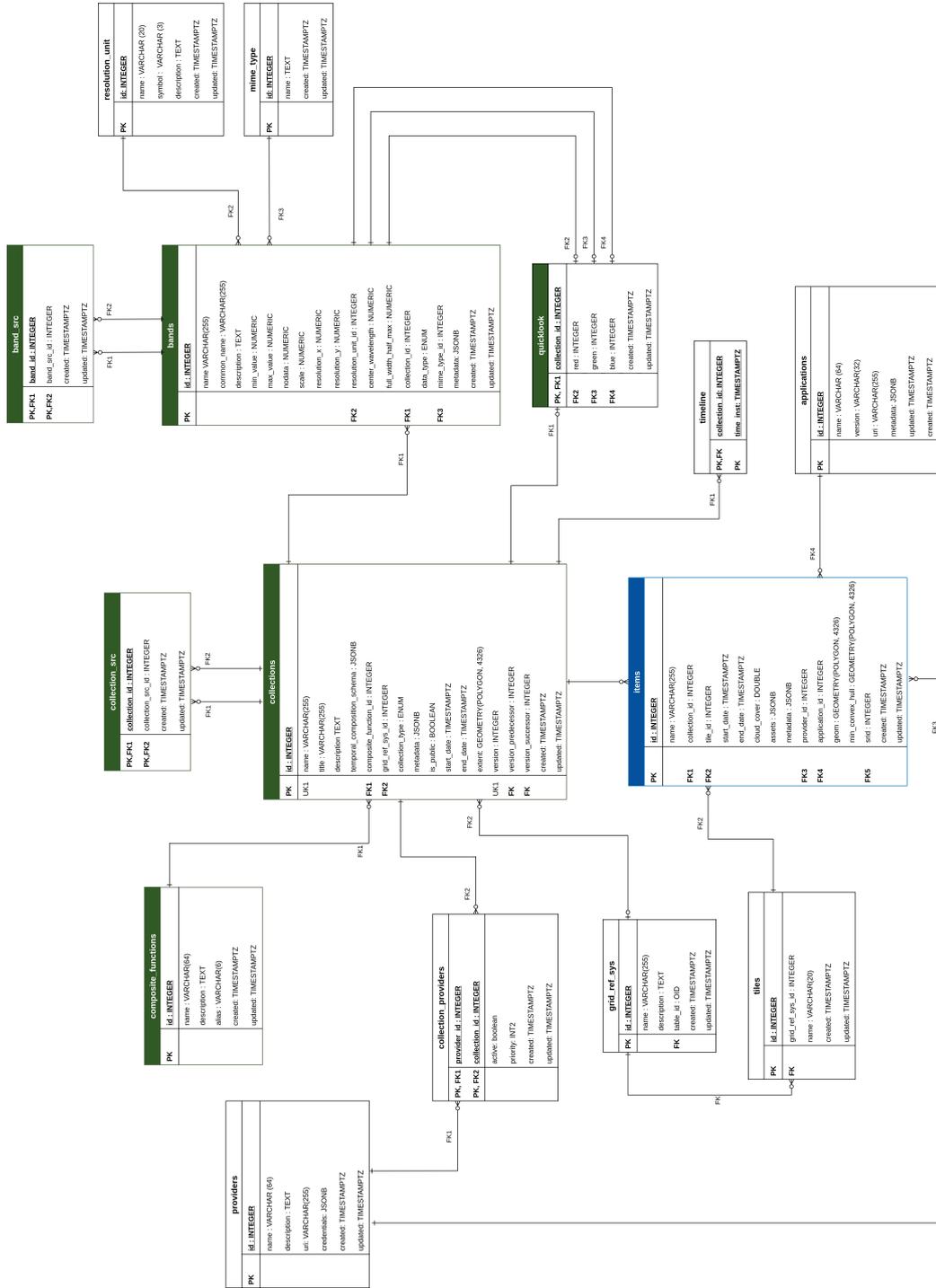
NAGPAL, A.; GABRANI, G. Python for data analytics, scientific and technical
applications. In: AMITY INTERNATIONAL CONFERENCE ON ARTIFICIAL
INTELLIGENCE, 2020. **Proceedings...** Dubai: IEEE, 2019. p. 140–145. 38

- NATIONAL AERONAUTICS AND SPACE ADMINISTRATION - NASA. **Landsat 8 overview**. 2020. Disponível em: <<https://landsat.gsfc.nasa.gov/landsat-data-continuity-mission>>. Acesso em: 14 jun. 2020. 5
- OPEN GEOSPATIAL CONSORTIUM. **OGC catalogue services 3.0 - general model**. 2016. Disponível em: <<http://docs.opengeospatial.org/is/12-176r7/12-176r7.html>>. Acesso em: 20 set. 2019. 8
- PÉREZ, F.; GRANGER, B. E. IPython: a system for interactive scientific computing. **Computing in Science and Engineering**, v. 9, n. 3, p. 21–29, maio 2007. ISSN 1521-9615. Disponível em: <<https://ipython.org/>>. 41
- QUEIROZ, G. R.; FERREIRA, K. R.; VINHAS, L.; CÂMARA, G.; COSTA, R. W.; SOUZA, R. C. M.; MAUS, V. W.; SANCHEZ, A. Wtss: um serviço web para extração de séries temporais de imagens de sensoriamento remoto. In: SIMPÓSIO BRASILEIRO DE SENSORIAMENTO REMOTO, 17., 2015. **Anais...** São José dos Campos: Instituto Nacional de Pesquisas Espaciais (INPE), 2015. p. 7553–7560. 46
- R CORE TEAM. **R: a language and environment for statistical computing**. Vienna, Austria, 2020. Disponível em: <<https://www.R-project.org/>>. 38
- RADIANT EARTH FOUNDATION. **SpatioTemporal asset catalog**. set. 2019. Disponível em: <<https://stacspec.org/>>. Acesso em: 19 jul. 2019. 9, 10, 13, 23, 25, 27, 28, 30, 31, 32, 33
- _____. **CBERS static STAC catalog served by stac-browser**. 2020. Disponível em: <<https://cbers.stac.cloud/?t=catalogs>>. Acesso em: 21 jul. 2020. 11
- REIS, J. B. C.; SANTOS, T. B.; LOPES, E. S. S. Monitoramento em tempo real de eventos extremos na região metropolitana de São Paulo – uma aplicação com o SISMADEN. In: SIMPÓSIO BRASILEIRO DE GEOGRAFIA FÍSICA APLICADA, 14., 2011, Dourados, MS. **Anais...** Dourados, MS, 2011. 1
- SOILLE, P.; BURGER, A.; MARCHI, D. D.; KEMPENEERS, P.; RODRIGUEZ, D.; SYRRIS, V.; VASILEV, V. A versatile data-intensive computing platform for information retrieval from big geospatial data. **Future Generation Computer Systems**, v. 81, p. 30–40, 2018. 16

- SONG, J.; DI, L. Near-real-time ogc catalogue service for geoscience big data. **ISPRS International Journal of Geo-Information**, v. 6, n. 11, p. 337, 2017. 8
- SOUZA V. C. O.; CAMARA, G. Global geoportal for remote sensing image centers. In: BRAZILIAN SYMPOSIUM ON GEOINFORMATICS - GEOINFO, 10., 2008, Rio de Janeiro. **Proceedings...** Rio de Janeiro: INPE, 2008. p. 63–74. 7
- UNITED STATES GEOLOGICAL SURVEY. **Landsat missions - Landsat 8**. nov. 2020. Disponível em:
<<https://www.usgs.gov/core-science-systems/nli/landsat/landsat-8>>. Acesso em: 19 nov. 2020. 17
- VALERIANO, D. M.; MELLO, E. M.; MOREIRA, J. C.; SHIMABUKURO, Y. E.; DUARTE, V.; SOUZA, I.; SANTOS, J.; BARBOSA, C. C.; SOUZA, R. Monitoring tropical forest from space: the prodes digital project. **International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences**, v. 35, p. 272–274, 2004. 1
- VINHAS, L.; QUEIROZ, G. R. de; FERREIRA, K. R.; CAMARA, G. Web services for big earth observation data. **Revista Brasileira de Cartografia**, v. 69, n. 5, 2017. 46
- XIE, R.; SHIBASAKI, R. Imagery metadata development based on iso/tc 211 standards. **Data Science Journal**, v. 6, p. 28–45, 2007. 6

ANEXO A - MODELO DE DADOS DO BRAZIL DATA CUBE

Figura A.1 - Modelo de dados das coleções de imagens e cubos do Projeto BDC.



Fonte: Produção do Autor.

PUBLICAÇÕES TÉCNICO-CIENTÍFICAS EDITADAS PELO INPE

Teses e Dissertações (TDI)

Teses e Dissertações apresentadas nos Cursos de Pós-Graduação do INPE.

Manuais Técnicos (MAN)

São publicações de caráter técnico que incluem normas, procedimentos, instruções e orientações.

Notas Técnico-Científicas (NTC)

Incluem resultados preliminares de pesquisa, descrição de equipamentos, descrição e ou documentação de programas de computador, descrição de sistemas e experimentos, apresentação de testes, dados, atlas, e documentação de projetos de engenharia.

Relatórios de Pesquisa (RPQ)

Reportam resultados ou progressos de pesquisas tanto de natureza técnica quanto científica, cujo nível seja compatível com o de uma publicação em periódico nacional ou internacional.

Propostas e Relatórios de Projetos (PRP)

São propostas de projetos técnico-científicos e relatórios de acompanhamento de projetos, atividades e convênios.

Publicações Didáticas (PUD)

Incluem apostilas, notas de aula e manuais didáticos.

Publicações Seriadas

São os seriados técnico-científicos: boletins, periódicos, anuários e anais de eventos (simpósios e congressos). Contam destas publicações o Internacional Standard Serial Number (ISSN), que é um código único e definitivo para identificação de títulos de seriados.

Programas de Computador (PDC)

São a seqüência de instruções ou códigos, expressos em uma linguagem de programação compilada ou interpretada, a ser executada por um computador para alcançar um determinado objetivo. Aceitam-se tanto programas fonte quanto os executáveis.

Pré-publicações (PRE)

Todos os artigos publicados em periódicos, anais e como capítulos de livros.