



MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÕES  
**INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS**

sid.inpe.br/mtc-m21c/2020/10.09.14.57-TDI

## PLATAFORMA PARA ACESSO E ANÁLISE DE TRAJETÓRIAS DE USO E COBERTURA DA TERRA

Fabiana Zioti

Dissertação de Mestrado do Curso de Pós-Graduação em Computação Aplicada, orientada pelos Drs. Gilberto Ribeiro de Queiroz e Karine Reis Ferreira Gomes, aprovada em 23 de outubro de 2020.

URL do documento original:

<<http://urlib.net/8JMKD3MGP3W34R/43CU7MB>>

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 INTELECTUAL 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  
**INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS**

sid.inpe.br/mtc-m21c/2020/10.09.14.57-TDI

## PLATAFORMA PARA ACESSO E ANÁLISE DE TRAJETÓRIAS DE USO E COBERTURA DA TERRA

Fabiana Zioti

Dissertação de Mestrado do Curso de Pós-Graduação em Computação Aplicada, orientada pelos Drs. Gilberto Ribeiro de Queiroz e Karine Reis Ferreira Gomes, aprovada em 23 de outubro de 2020.

URL do documento original:

<<http://urlib.net/8JMKD3MGP3W34R/43CU7MB>>

INPE  
São José dos Campos  
2020

Dados Internacionais de Catalogação na Publicação (CIP)

---

Zioli, Fabiana.

Z67p      Plataforma para Acesso e Análise de Trajetórias de Uso e Cobertura da Terra / Fabiana Zioli. – São José dos Campos : INPE, 2020.

xxii + 97 p. ; (sid.inpe.br/mtc-m21c/2020/10.09.14.57-TDI)

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

Orientadores : Drs. Gilberto Ribeiro de Queiroz e Karine Reis Ferreira Gomes.

1. Análise de trajetórias. 2. Integração de dados. 3. Uso e cobertura da terra. 4. Séries temporais. I.Título.

CDU 502.5:004.62

---



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/).



MINISTÉRIO DA  
CIÊNCIA, TECNOLOGIA  
E INOVAÇÕES



## INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

### DEFESA FINAL DE DISSERTAÇÃO DE FABIANA ZIOTI

No dia 23 de outubro de 2020, às 09h, por videoconferência, 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).

Título: "Plataforma para Acesso e Análise de Trajetórias de Uso e Cobertura da Terra"

Eu, Pedro Ribeiro de Andrade Neto, como Presidente da Banca Examinadora, assino esta ATA em nome de todos os membros.

#### Membros da Banca

Dr. Pedro Ribeiro de Andrade Neto, Presidente INPE.  
Dr. Gilberto Ribeiro de Queiroz, Orientador(a) INPE  
Dra. Karine Reis Ferreira Gomes, Orientador(a) INPE  
Dr. Thales Sehn Korting, Membro da Banca, INPE  
Dr. Júlio César Dalla Mora Esquerdo, Convidado(a) Embrapa



Documento assinado eletronicamente por **Pedro Ribeiro de Andrade Neto, Tecnologista**, em 04/11/2020, às 10:47 (horário oficial de Brasília), com fundamento no art. 6º, § 1º, 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 **6041238** e o código CRC **BFA38459**.



*“Science Is Wise, Follow Its Advice”.*

RICK SANCHEZ  
em “Rick and Morty”.



## AGRADECIMENTOS

Agradeço a meus pais Márcio da Conceição Zioti e Regina Celi Baroni Zioti, pelo enorme amor, carinho, dedicação e paciência durante todos esses anos e por tudo o que tenho na minha vida.

Agradeço a minhas queridas e amadas avós, Margarida de Souza Baroni e Sebastiana de Jesus Silva. Duas grandes guerreiras lindas que sempre vão estar no meu coração.

Ao meu amado irmão Éverton José Zioti pela amizade, paciência e apoio.

Agradeço especialmente ao meu namorado Leonardo de Souza Vieira por estar sempre ao meu lado durante esta jornada.

Agradeço a todos os meus familiares e amigos, em especial às minhas tias Maria Zioti, Lourdes de Carvalho e minha prima Mariana Carvalho.

Agradeço às minhas tias de coração Maria José Pinto e Cesa Pinto pelo apoio, conselhos e pela amizade.

Agradeço aos meus colegas de mestrado pelo apoio ao longo desta jornada.

Agradeço aos pesquisadores e desenvolvedores do projeto *Brazil Data Cube* pelas sugestões e experiências compartilhadas ao longo deste trabalho.

Agradeço aos meus orientadores Dr. Gilberto R. Queiroz e Dra. Karine Reis Ferreira, pelos ensinamentos e atenção com que me orientaram neste trabalho.

Agradeço ao CNPq, Conselho Nacional de Desenvolvimento Científico e Tecnológico - Brasil pelo apoio financeiro.

Agradeço a todos que contribuíram de alguma forma para desenvolvimento deste trabalho.



## RESUMO

As informações sobre o uso e cobertura da terra são essenciais para apoiar os governos na tomada de decisões sobre o impacto das atividades humanas no meio ambiente, para o planejamento do uso dos recursos naturais, a conservação da biodiversidade e o monitoramento das mudanças climáticas. Atualmente, diversos projetos fornecem sistematicamente informações sobre a dinâmica de uso e cobertura da terra. Projetos bem conhecidos incluem PRODES, TerraClass e MapBiomias. Embora esses projetos sigam políticas de dados abertos e forneçam uma rica coleção de dados, ainda há uma lacuna em ferramentas que facilitem o uso integrado dessas coleções. A coleta, organização e integração desses dados requer um esforço considerável por parte de pesquisadores, estudantes e o público em geral, consumindo uma boa parte do tempo em diversos projetos e atividades. Cada projeto adota seu próprio sistema de classificação de uso e cobertura da terra, fornecendo nomes de classes e significados diferentes para os elementos dessas coleções. As formas de distribuição dos dados dos projetos podem ser realizadas de diferentes maneiras, através de arquivos ou *web services*. Além disso, os dados possuem diferentes resoluções espaciais e temporais e sistemas de armazenamento (matricial ou vetorial). Neste contexto, este trabalho propõe o desenvolvimento de uma plataforma para acessar e recuperar trajetórias de uso e cobertura da terra provenientes de diferentes tipos e fontes de dados. Essa plataforma é composta por dois serviços: *Web Land Trajectory Service* (WLTS) e *Web Land Classification Systems Service* (WLCSS). A abordagem do WLTS consiste em usar um modelo de dados que define um conjunto mínimo de informações temporais e espaciais para representar diferentes fontes e tipos de dados, mas com foco em uso e cobertura da terra. O WLCSS fornece uma interface simplificada para representação, acesso e manipulação de sistemas de classificação, complementando o serviço WLTS. A plataforma proposta pode ser usada em uma variedade de aplicações, como na validação de conjuntos de dados de cobertura da terra, na seleção de amostras de treinamento para suportar algoritmos de Aprendizagem de Máquina usados na geração de novos mapas de classificação. Como estudo de caso, foi exercitado o uso da plataforma com os dados dos projetos TerraClass, PRODES, DETER, MapBiomias e *Brazil Data Cube*. Os resultados mostram a viabilidade e vantagens dessa integração de dados, mantendo uma interface amigável para acesso e análise de trajetórias de uso e cobertura da terra.

Palavras-chave: Análise de Trajetórias. Integração de Dados. Uso e Cobertura da Terra. Séries Temporais.



# PLATFORM FOR ACCESS AND ANALYSIS OF LAND USE AND COVER TRAJECTORY

## ABSTRACT

The information of on land use and land cover is essential to support governments in making decisions about the impact of human activities on the environment, planning the use of natural resources, conserving biodiversity and monitoring climate change. Currently, several projects systematically provide information on the dynamics of land use and coverage. Well-known projects include PRODES, TerraClass and MapBiomias. Although these projects follow open data policies and provide a rich collection of data, there is still a gap in tools that facilitate the integrated use of these collections. The collection, organization and integration of this data requires considerable effort on the part of researchers, students and the general public, consuming a good deal of time in various projects and activities. Each project adopts its own land use and land cover classification system, providing different class names and meanings for the elements of these collections. The distribution of project data can be done in different ways, through files or web services. In addition, the data has different spatial and temporal resolutions and storage systems (matrix or vector). In this context, we propose the development of a platform to access and recover trajectories from different types and data sources. This platform consists of two services, Web Land Trajectory Service (WLTS) and Web Land Classification System Service (WLCSS). The WLTS approach consists of using a data model that defines a minimum set of temporal and spatial information to represent different sources and types of data, but with a focus on land use and coverage. WLCSS provides a simplified interface for the representation, access and manipulation of classification systems, complementing the WLTS service. The proposed platform can be used in a variety of applications, such as validating land cover data sets, selecting training samples to support Machine Learning algorithms used in the generation of new classification maps. As a case study, the use of the platform was exercised with data from the TerraClass, PRODES, DETER, MapBiomias and Brazil Data Cube projects. The results show the feasibility and advantages of this data integration, maintaining a friendly interface for access and analysis of land use and coverage trajectories.

Keywords: Trajectory Analysis. Data Integration. Land Use and Coverage. Time Series.



## LISTA DE FIGURAS

	<u>Pág.</u>
2.1 Exemplo de diferença entre uso e cobertura da terra. . . . .	7
2.2 Estimativa do volume de dados livres por equipamento. . . . .	9
2.3 Séries Temporais extraídas de imagens de sensoriamento remoto. . . . .	10
2.4 Exemplo de representação de Trajetórias de UCT extraídas de imagens classificadas. . . . .	13
2.5 Programas de Monitoramento Ambiental do INPE. . . . .	18
2.6 Projeto MapBiomias coleção 3 da série anual de mapas de cobertura e uso da terra do Brasil do bioma Amazônia. . . . .	19
3.1 Visão geral da plataforma para acesso e análise de trajetórias de uso e cobertura da terra. . . . .	24
3.2 Modelo de dados do LCS-DB. . . . .	26
3.3 Representação da definição de <i>Trajectory</i> no contexto do WLTS. . . . .	33
3.4 Arquitetura do WLTS. . . . .	34
3.5 Exemplos de DataSource preenchidos no DIM. . . . .	36
4.1 Exemplo do modelo proposto para integração de dados: DIM. . . . .	44
4.2 Exemplo de como importar a biblioteca cliente <code>wlts.py</code> , através do Jupyter Notebook. . . . .	47
4.3 Trajetória de UCT usando a biblioteca cliente <code>wlts.py</code> . . . . .	49
4.4 DataFrame com Trajetórias de UCT usando a biblioteca cliente <code>wlts.py</code> , através do <i>Jupyter Notebook</i> . . . . .	50
4.5 Exemplos de Trajetórias para a região do Cerrado. . . . .	51
4.6 Trajetória Harmonizada. . . . .	53
4.7 Área de estudo com as amostras utilizadas no estudo de caso 2. . . . .	54
4.8 Classificação <i>Random forest</i> do C4_64_16D_MED. . . . .	55
4.9 Brazil Data Cube CBERS Versus MapBiomias. . . . .	57
4.10 Trajetória Brazil Data Cube CBERS Versus MapBiomias. . . . .	58
A.1 Diagrama de Classe simplificado da camada de CDS. . . . .	71
A.2 Diagrama do modelo de dados proposto para o WLTS . . . . .	72
C.1 Página do repositório do LCS-DB no Github. . . . .	93
C.2 Página do repositório do WLCSS no Github. . . . .	94
C.3 Página do repositório do Cliente em Python para WLCSS no Github. . . . .	95
C.4 Página do repositório do WLTS no Github. . . . .	96

C.5	Página do repositório do Cliente em Python para WLTS no Github. . . .	97
-----	---	----

## LISTA DE TABELAS

	<u>Pág.</u>
2.1 Tipos de Dados dos Padrões Serviços Web da OGC. . . . .	20
3.1 Dados inseridos na tabela <b>lccs.classification_systems</b> proveniente do LCS-DB. . . . .	27
3.2 Dados inseridos na tabela <b>lccs.classes</b> proveniente do LCS-DB. . . . .	27
3.3 Dados inseridos para o mapeamento na tabela <b>lccs.classes</b> provenientes do modelo LCS-DB. . . . .	28
3.4 Campos presentes nos <i>links</i> do WLCSS. . . . .	29
3.5 Parâmetros da operação que retorna informações sobre uma única classe. . . . .	30
3.6 Parâmetros da operação com os mapeamentos de sistemas de classificação. . . . .	31
3.7 Parâmetros da operação <code>describe_collections</code> no WLTS. . . . .	39
3.8 Parâmetros da operação <code>trajectory</code> no WLTS. . . . .	40
4.1 Informações dos <i>datasets</i> utilizados no estudo de caso. . . . .	43
4.2 Metadado das Collections no WLTS. . . . .	45



## LISTA DE ABREVIATURAS E SIGLAS

API	–	<i>Application Programming Interface</i>
AVHRR	–	<i>Advanced Very High Resolution Radiometer</i>
BDC	–	<i>Brazil Data Cube</i>
CBERS	–	<i>China-Brazil Earth Resources Satellite</i>
CDOBT	–	<i>Cubo de Dados de Observação da Terra</i>
CORINE	–	<i>Coordination of Information on the Environment</i>
DL	–	<i>Deep Learning</i>
EMBRAPA	–	<i>Empresa Brasileira de Pesquisa Agropecuária</i>
ESA	–	<i>European Space Agency</i>
EVI	–	<i>Enhanced Vegetation Index</i>
FAO	–	<i>Food and Agriculture Organization</i>
GLCC	–	<i>Global Land Cover Characterization</i>
IBGE	–	<i>Instituto Brasileiro de Geografia e Estatística</i>
IGBP	–	<i>International Geosphere-Biosphere Program</i>
INPE	–	<i>Instituto Nacional de Pesquisas Espaciais</i>
JSON	–	<i>JavaScript Object Notation</i>
LCS	–	<i>Land Classification System</i>
LCS-DB	–	<i>Land Classification System Database Model</i>
LCML	–	<i>Land Cover Meta Language</i>
MODIS	–	<i>Moderate Resolution Imaging Spectroradiometer</i>
ML	–	<i>Machine Learning</i>
NASA	–	<i>National Aeronautics and Space Administration</i>
NDVI	–	<i>Normalized Difference Vegetation Index</i>
NOAA	–	<i>National Oceanic and Atmospheric Administration</i>
OT	–	<i>Observação da Terra</i>
OGC	–	<i>Open Geospatial Consortium</i>
SR	–	<i>Sensoriamento Remoto</i>
UCT	–	<i>Uso e Cobertura da Terra</i>
USGS	–	<i>United States Geological Survey</i>
WCS	–	<i>Web Coverage Service</i>
WCPS	–	<i>Web Coverage Processing Service</i>
WFS	–	<i>Web Feature Service</i>
WLCSS	–	<i>Web Land Classification System Service</i>
WLTS	–	<i>Web Land Trajectory Service</i>
WMS	–	<i>Web Map Service</i>
WPS	–	<i>Web Processing Service</i>
WTSS	–	<i>Web Time Series Service</i>



## LISTA DE CÓDIGOS

1	Definição de retorno da operação <code>classification_systems</code> . . . . .	29
2	Definição de retorno da operação que descreve um sistema de classificação. . . . .	30
3	Definição de retorno da operação que busca uma única classe. . . . .	31
4	Definição de retorno da operação <code>list_collections</code> . . . . .	38
5	Definição de retorno da operação <code>describe_collection</code> . . . . .	39
6	Definição de retorno da operação <code>trajectory</code> . . . . .	40
7	Linha de comando do <code>wlts-cli</code> do <code>wlts.py</code> . . . . .	46
8	Listar as <code>collections</code> usando <code>wlts.py</code> . . . . .	47
9	Retorno de metadados de uma coleção usando <code>wlts.py</code> . . . . .	48
10	Acessando os metadados e classes do sistema PRODES . . . . .	52
11	Especificação Open API WLTS. . . . .	73
12	Especificação Open API WLTS Continuação 1. . . . .	74
13	Especificação Open API WLTS Continuação 2. . . . .	75
14	Especificação Open API WLTS Continuação 3. . . . .	76
15	Especificação Open API WLTS Continuação 4. . . . .	77
16	Especificação Open API WLTS Continuação 5. . . . .	78
17	Especificação Open API WLTS Continuação 6. . . . .	79
18	Especificação Open API WLCSS . . . . .	81
19	Especificação Open API WLCSS Continuação 1. . . . .	82
20	Especificação Open API WLCSS Continuação 2. . . . .	83
21	Especificação Open API WLCSS Continuação 3. . . . .	84
22	Especificação Open API WLCSS Continuação 4. . . . .	85
23	Especificação Open API WLCSS Continuação 5. . . . .	86
24	Especificação Open API WLCSS Continuação 6. . . . .	87
25	Especificação Open API WLCSS Continuação 7. . . . .	88
26	Especificação Open API WLCSS Continuação 8. . . . .	89
27	Especificação Open API WLCSS Continuação 9. . . . .	90
28	Especificação Open API WLCSS Continuação 10. . . . .	91
29	Especificação Open API WLCSS Continuação 11. . . . .	92



## SUMÁRIO

	<u>Pág.</u>
<b>1 INTRODUÇÃO</b> . . . . .	<b>1</b>
1.1 Motivação . . . . .	2
1.2 Objetivos . . . . .	5
1.3 Organização . . . . .	5
<b>2 REFERENCIAL TEÓRICO</b> . . . . .	<b>7</b>
2.1 Uso e cobertura da terra . . . . .	7
2.2 Geração de dados de uso e cobertura da terra . . . . .	10
2.3 Trajetórias de uso e cobertura da terra . . . . .	11
2.4 Sistemas de classificação . . . . .	14
2.5 Programas de monitoramento ambiental nacional . . . . .	16
2.5.1 PRODES . . . . .	16
2.5.2 DETER . . . . .	17
2.5.3 TerraClass . . . . .	17
2.5.4 MapBiomias . . . . .	18
2.6 Serviços Web Geoespaciais . . . . .	19
<b>3 PLATAFORMA PARA ACESSO E ANÁLISE DE TRAJETÓRIAS DE USO E COBERTURA DA TERRA</b> . . . . .	<b>23</b>
3.1 Visão Geral da Plataforma . . . . .	23
3.2 <i>Land Classification System Database Model</i> LCS-DB . . . . .	25
3.3 <i>Web Land Classification System Service</i> (WLCSS) . . . . .	28
3.4 <i>Web Land Trajectory Service</i> (WLTS) . . . . .	32
3.4.1 Definições . . . . .	32
3.4.2 Arquitetura . . . . .	34
3.4.3 Modelo de Integração de Dados . . . . .	35
3.4.4 <i>Collections e Data Source</i> . . . . .	37
3.4.5 <i>Operations</i> . . . . .	38
3.5 Implementação . . . . .	40
<b>4 ESTUDO DE CASO E RESULTADOS</b> . . . . .	<b>43</b>
4.1 Estudo de caso 1 . . . . .	43
4.2 Estudo de caso 2 . . . . .	53

4.2.1	Área de Estudo . . . . .	53
4.2.2	Dados . . . . .	54
4.2.3	Resultados . . . . .	56
<b>5</b>	<b>CONCLUSÕES . . . . .</b>	<b>59</b>
	<b>REFERÊNCIAS BIBLIOGRÁFICAS . . . . .</b>	<b>61</b>
	<b>ANEXO A - DIAGRAMAS . . . . .</b>	<b>71</b>
A.1	Diagrama camada CDS do WLTS . . . . .	71
A.2	Diagrama do modelo de dados proposto para o WLTS . . . . .	72
	<b>ANEXO B - ESPECIFICAÇÃO <i>OPEN API</i> . . . . .</b>	<b>73</b>
B.1	WLTS-SPEC . . . . .	73
B.2	WLCS-SPEC . . . . .	81
	<b>ANEXO C - REPOSITÓRIOS DE DADOS . . . . .</b>	<b>93</b>
C.1	LCS-DB . . . . .	93
C.2	WLCSS . . . . .	94
C.3	WLTS . . . . .	96

# 1 INTRODUÇÃO

As atividades humanas alteram a superfície da Terra, influenciando o clima, na demanda por água, na produção de alimentos e energia. O desmatamento e a queima de florestas podem perturbar o equilíbrio da atmosfera terrestre alterando os níveis de gases de efeito estufa no planeta. O aumento destes gases pode alterar o ciclo natural do carbono, o que contribui diretamente com alterações climáticas e o aquecimento global (FOLEY et al., 2005). Diante dessas mudanças observadas na superfície do planeta é essencial fomentar estudos socioambientais, desenvolver novas tecnologias e modelos socioeconômicos auto-sustentáveis e definir políticas e práticas públicas que diminuam o impacto das atividades humanas sobre o meio ambiente. Neste contexto, as informações sobre o Uso e Cobertura da Terra (UCT) desempenham um papel importante em estudos de mudanças globais. Segundo Lu et al. (2004) um dos objetivos de obter dados sobre o UCT é buscar entender as relações entre os seres humanos e o meio ambiente com o intuito de gerenciar os recursos naturais para um desenvolvimento sustentável.

O Sensoriamento Remoto (SR) aplicado à Observação da Terra (OT) é uma ferramenta para obtenção de informações da superfície terrestre. As imagens provenientes dos sensores a bordo dos satélites são uma importante fonte de dados empregadas no monitoramento da Terra em escala regional e global (ARVOR et al., 2011b; AGUIAR et al., 2010; GÓMEZ et al., 2016). Os satélites de OT conseguem, através dos sensores a bordo, monitorar lugares inacessíveis pelo homem, realizando varreduras de uma vasta área em um curto período de tempo.

Os avanços nas tecnologias de SR têm possibilitado a aquisição de dados com resoluções espaciais e temporais cada vez mais finas. Com isso, existe uma grande quantidade e diversidade de fontes de produtos derivados de sensoriamento remoto disponíveis para utilização. Para aproveitar os dados de OT disponíveis, diversas iniciativas e estudos visam o desenvolvimento de novas metodologias para geração de mapas de uso e cobertura da Terra (CÂMARA et al., 2016). Muitos projetos investem em técnicas de *Big Data* e métodos de classificação automática com o emprego de algoritmos de *Machine Learning* (ML) e *Deep Learning* (DL) Supervisionados (MAPBIOMAS, ; CÂMARA et al., 2018; PICOLI et al., 2020).

Devido à disponibilidade de séries históricas de dados e novas metodologias de acesso aos dados, o uso de séries temporais de imagens de SR tem sido amplamente disseminado nas últimas décadas (GÓMEZ et al., 2016; SPERA et al., 2014). A análise e classificação de séries temporais são temas recorrentes em pesquisas em SR, permi-

tindo a análise de padrões espaço-temporais de áreas ambientais e do impacto das atividades humanas. Diferentemente de abordagens clássicas que focam prioritariamente na relação espacial dos dados, a análise e classificação de séries temporais também utiliza relação temporal dos dados na geração de mapas de uso e cobertura.

Se por um lado essa disponibilidade crescente de dados permite o desenvolvimento de diferentes metodologias de monitoramento, por outro, a variabilidade e volume de dados exige o desenvolvimento de tecnologias cada vez mais complexas que permitam o armazenamento, gerenciamento, manipulação, processamento e análise destes dados (CÂMARA et al., 2018; VINHAS et al., 2017).

A análise temporal de mapas de uso e cobertura da terra fornece um conjunto de informações históricas que podem auxiliar no desenvolvimento de diferentes aplicações. A disponibilização de forma sistemática de informações da dinâmica do uso e cobertura da terra permite substituir metodologias que adotam prioritariamente mudanças entre duas datas de mapas. Um alvo (localização) mapeado ao longo do tempo, permite analisar as mudanças temporais de UCT ao longo do tempo e não apenas mudanças espaciais na cobertura da terra.

## 1.1 Motivação

Existem diversas iniciativas que geram mapas de UCT em escalas regionais, nacionais e globais, visando atender a um grande número de aplicações. Esses produtos de dados são gerados utilizando diferentes abordagens e metodologias para apresentar informações sobre diferentes classes da superfície da terra, como florestas, áreas agrícolas, entre outras. Cada iniciativa desenvolve sua metodologia baseada no objetivo do projeto, isto resulta em um sistema de classificação próprio com diferentes nomenclaturas e significados das classes.

No âmbito nacional, iniciativas como o PRODES<sup>1</sup>, o DETER<sup>2</sup> e o *TerraClass*<sup>3</sup> produzem informações sobre o UCT usadas pelo governo brasileiro para tomada de decisão de políticas públicas. Esses projetos são desenvolvidos pelo INPE (Instituto Nacional de Pesquisas Espaciais) e instituições parceiras, como a Embrapa (Empresa Brasileira de Pesquisa Agropecuária). No caso destes projetos, grande parte da informação produzida é obtida através da interpretação visual de imagens por especialistas. O INPE também desenvolve dois projetos, o *e-Sensing*<sup>4</sup> e o *Brazil*

---

<sup>1</sup><http://www.obt.inpe.br/OBT/assuntos/programas/amazonia/prodes>

<sup>2</sup><http://www.obt.inpe.br/OBT/assuntos/programas/amazonia/deter>

<sup>3</sup><https://www.terraclass.gov.br/>

<sup>4</sup><http://www.esensing.org>

*Data Cube (BDC)*<sup>5</sup>, que utilizam tecnologias de *Big Data* e métodos de ML para gerar mapas classificados de UCT. Outro projeto que faz o uso de métodos de ML para processar os dados espaciais no cenário nacional é o *MapBiomass*<sup>6</sup> (Projeto de Mapeamento Anual da Cobertura e Uso do Solo no Brasil). Em uma escala global, iniciativas como *Global Land Cover Characterization*<sup>7</sup> e *Copernicus Global Land Service*<sup>8</sup> também produzem mapas com informações sobre UCT.

Embora esses projetos disponibilizem de maneira aberta uma rica coleção de dados, o uso integrado dessas coleções requer um grande esforço para coleta, organização e integração. Como [Chen et al. \(2017\)](#) destaca, muitos portais de dados espaciais fornecem catálogos de informações e APIs (*Application Programming Interfaces*) que permitem o acesso aos dados. Entretanto, nenhuma dessas alternativas fornece uma solução abrangente para facilitar a integração e harmonização das informações de UCT com diferentes componentes semânticos, espaciais e temporais.

[Herold et al. \(2008\)](#) discutem a dificuldade em avaliar, selecionar e até mesmo reutilizar os dados com temáticas diferentes em outras aplicações. Uma possível solução para este problema é a construção de um conjunto de dados organizado e harmonizado para auxiliar na validação de novos mapas e na seleção de amostras para o treinamento.

Muitos projetos fazem o uso de técnicas de ML para gerar informações sobre o UCT. Porém, a geração de informação através da interpretação visual de especialistas ainda é a fonte mais confiável na produção desse tipo de dado ([INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS - INPE, 2008](#)). Diversos fatores influenciam diretamente na precisão dos algoritmos de ML, sendo a criação de uma base de dados adequada e abrangente para serem utilizados no treinamento um grande desafio no desenvolvimento, principalmente para classificações em grande escala ([LEINENKUGEL et al., 2019](#)). Muitas pesquisas constroem o conjunto de amostras rotuladas através de observações em campo e/ou análise visual de imagens por especialistas. Esse tipo de abordagem demanda um tempo considerável, o que pode limitar o número de amostras coletadas para as pesquisas.

Uma forma de adquirir um conjunto de treinamento suficientemente grande é a utilização de um conjunto de informações consolidadas ([ABDI, 2020](#)) no processo

---

<sup>5</sup><http://brazildatacube.org>

<sup>6</sup><https://mapbiomas.org>

<sup>7</sup>[https://www.usgs.gov/centers/eros/science/usgs-eros-archive-land-cover-products-global-land-cover-characterization-glcc?qt-science\\_center\\_objects=0#qt-science\\_center\\_objects](https://www.usgs.gov/centers/eros/science/usgs-eros-archive-land-cover-products-global-land-cover-characterization-glcc?qt-science_center_objects=0#qt-science_center_objects)

<sup>8</sup><https://land.copernicus.eu/global/products/lc>

de amostragem. Desta forma, é possível gerar um grande número de amostras a partir de conjuntos de dados já classificados e gerar novos mapas de UCT sem a necessidade de um processo manual de coleta.

Diversos trabalhos têm como o foco a geração de mapas de UCT a partir de séries temporais. Uma série temporal pode ser entendida como um conjunto de observações (por exemplo, níveis de reflectância) de uma mesma localidade ordenadas temporalmente (VINHAS et al., 2017). Ao se tratar de dados rotulados (classes de UTC), é usual utilizar o termo trajetória para definir um conjunto de observações temporalmente ordenadas. Neste sentido, as trajetórias definem as transição de rótulos em função do tempo para uma dada localidade (ZHOU et al., 2008). Mena (2008) apresenta em seu trabalho, como o uso de trajetórias pode ser empregado na avaliação e análises ligadas às consequências do uso da terra. A partir destes dois conceitos pode-se desenvolver um conjunto de análises temporais sobre dados de SR.

A análise de trajetórias de UCT pode ser utilizada para diferentes propósitos, seja na fase de pré-processamento ou pós-processamento dos métodos de ML. Na fase de pré-processamento a análise de trajetórias de UCT pode ser auxiliar na validação de amostras de UCT, ou seja, para verificar a precisão de dados específicos e assim gerar dados de treinamentos mais coerentes para minimizar as classificações incorretas. Na etapa de pós-processamento as trajetórias de UCT podem ser empregadas como uma metodologia auxiliar para validar novos mapas de UCT.

Muitas pesquisas fazem o uso de trajetórias de UCT, através da coleta e organização manual dos dados. Esta abordagem traz diferentes dificuldades inerentes a reusabilidade dos dados. As trajetórias provenientes de diferentes fontes possuem variações de semânticas de classe, diferentes resoluções espaciais e temporais, além de possuírem estruturas de armazenamento (raster ou vetorial). Essas variações agregadas com a falta de metadados que descrevam os conjuntos de dados impõem obstáculos significantes no uso conjunto de dados UCT (CHEN et al., 2017).

Neste contexto, este trabalho propõe o desenvolvimento de uma abordagem automatizada que permite o acesso e recuperação de trajetórias de UCT provenientes de dados de diferentes fontes. Com isto, espera-se que a abordagem proposta possa ser usada em diversas aplicações: na validação de conjuntos de dados de cobertura do solo, no auxílio na seleção de amostras de treinamento e no suporte ao desenvolvimento de mapas de classificação através de algoritmos de ML.

## 1.2 Objetivos

Considerando a demanda da comunidade científica, este trabalho tem o objetivo de desenvolver um modelo de integração de dados multi-fontes junto ao desenvolvimento de um serviço de acesso e recuperação de trajetórias de uso e cobertura da terra. Além disto, podem ser destacados os seguintes objetivos específicos:

- Propor e desenvolver um modelo de dados e um serviço web que facilite a identificação e acesso e mapeamento de diversos sistemas de classificação.
- Desenvolver o modelo de integração de dados a partir de um conjunto mínimo de informações provenientes das fontes distintas de dados espaço-temporais, permitindo a integração de diferentes tecnologias de armazenamento e acesso: SGBD, WFS, WCS.
- Propor e desenvolver um serviço web especializado na recuperação e integração de trajetórias de uso e cobertura da terra de diferentes fontes de dados.
- Possibilitar o uso dos serviços em ambientes de computação interativa (RStudio, Jupyter).

## 1.3 Organização

No Capítulo 2 são apresentados os conceitos e definições utilizados como base para o desenvolvimento do trabalho. No Capítulo 3, é apresentada a metodologia de desenvolvimento do trabalho e da plataforma propostos. Nesse capítulo são abordadas as tecnologias utilizadas para o desenvolvimento da plataforma proposta e como cada serviço (componente) da plataforma foi construído. No Capítulo 4 são evidenciados os resultados obtidos por meio de estudos de casos. Por fim, no Capítulo 5 são apresentadas as conclusões do trabalho.

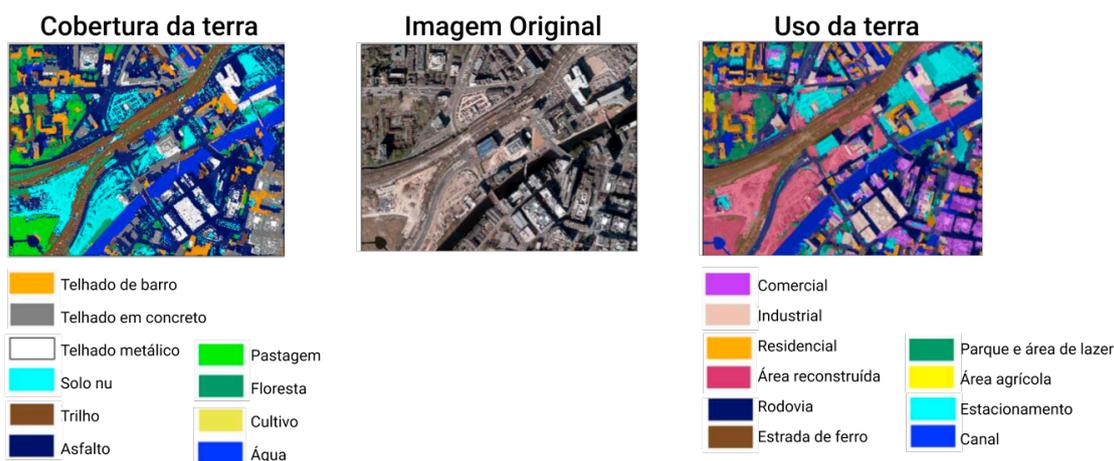


## 2 REFERENCIAL TEÓRICO

### 2.1 Uso e cobertura da terra

Cobertura da terra refere-se à biofísica observada da superfície da terra (JANSEN; GREGORIO, 2000; MEYER; TURNER, 1992). Alguns exemplos são: florestas, áreas urbanas, corpos d'água, entre outros (GREGORIO, 2016). O uso da terra é definido a partir de interpretações socioeconômicas de atividades que ocorrem na superfície terrestre (MATTILA et al., 2011). A conexão entre uso e cobertura da terra (UCT) é complexa, pois tipos diferentes de uso podem estar relacionados a vários tipos de cobertura. A Figura 2.1 ilustra exemplos de imagens classificadas, destacando a diferença entre as classes de uso e cobertura da terra.

Figura 2.1 - Exemplo de diferença entre uso e cobertura da terra.



Fonte: Adaptado de Zhang et al. (2019).

Os dados de UCT possuem aplicações em diversas áreas, podendo ser empregados, por exemplo, no gerenciamento de terras públicas, no manejo de recursos energéticos, no monitoramento da biodiversidade, na expansão urbana, entre outros (ANDERSON et al., 1976; SONG et al., 2016; WANG et al., 2018). As raízes históricas do mapeamento da cobertura da terra residem no início da história da fotografia aérea. Mas com a evolução das plataformas satelitais, as imagens de satélite tornaram-se a principal fonte de dados para a geração os dados de informações sobre a superfície da Terra. E isso impulsionou o monitoramento da Terra em escala regional e global, permitindo

detectar as mudanças de UCT com maior frequência temporal.

Considerando o monitoramento do UCT em escalas regionais e locais, os dados da missão *Landsat* são utilizados para realizar mapeamentos desde 1972 em iniciativas de diversos países (WULDER et al., 2016; WULDER et al., 2008; GIRI, 2016). Na década de 1980, foram aprimorados diversos aspectos do mapeamento do uso e cobertura da terra. O sensor TM (*Thematic Mapper*) trouxe melhores capacidades espaciais e multi-espectrais com as missões do *Landsat* 4 e 5. De maneira similar a missão SPOT (*French Satellite Pour l'Observation de la Terre*) também ajudou nas melhorias de mapeamento com sensores fornecendo dados de alta resolução espacial (GIRI, 2016).

Além das melhorias no mapeamento do uso e cobertura da terra desenvolvida a partir de sensores de resoluções mais finas na década de 1980, a evolução no desenvolvimento de satélites geoestacionários possibilitou, em função das características dos sensores de órbita polar (período, ângulo de varredura, etc.), diferentes aplicações em escala global. Tucker et al. (1985) utilizaram imagens de resolução grossa com o uso de AVHRR (*Advanced Very High Resolution Radiometer*) a bordo dos satélites meteorológicos NOAA (*National Oceanic and Atmospheric Administration*), para mapear a cobertura da terra para o continente africano. Diversos estudos utilizaram os dados AVHRR para mapear a cobertura da terra (LIU et al., 2003), evidenciando sua aplicação em escala global. O IGBP (*International Geosphere-Biosphere Program*) publicou o primeiro mapa global com 1.1 km de resolução espacial, denominado de *Discover*, proveniente de uma única fonte de dados: o *AVHRR Local Area Coverage* (MAYAUX et al., 2006).

Com a evolução nas tecnologias para aquisição de dados, foram desenvolvidos satélites com resoluções espaciais cada vez mais finas, o que tem gerado outra perspectiva na produção de mapas globais de UCT (FRIEDL et al., 2002; ARINO et al., 2007). Como exemplos de produtos de resolução mais finas pode-se citar: *MODIS Land Cover*<sup>1</sup>, *Global Land Cover 2000*<sup>2</sup> e *Globcover 2006*<sup>3</sup>.

Por muitas décadas a utilização de imagens de satélites era restrita, com limitações de disponibilidade, acessibilidade provenientes principalmente do custo financeiro para realizar a aquisição destes dados. O Brasil é pioneiro na adoção de políticas de dados livres e abertos. Desde 2004 o INPE aplica essas políticas para o programa CBERS (*China-Brazil Earth Resources Satellite*) (CEOS, 2018). Em 2008, seguindo

---

<sup>1</sup><https://modis.gsfc.nasa.gov/data/dataproduct/mod12.php>

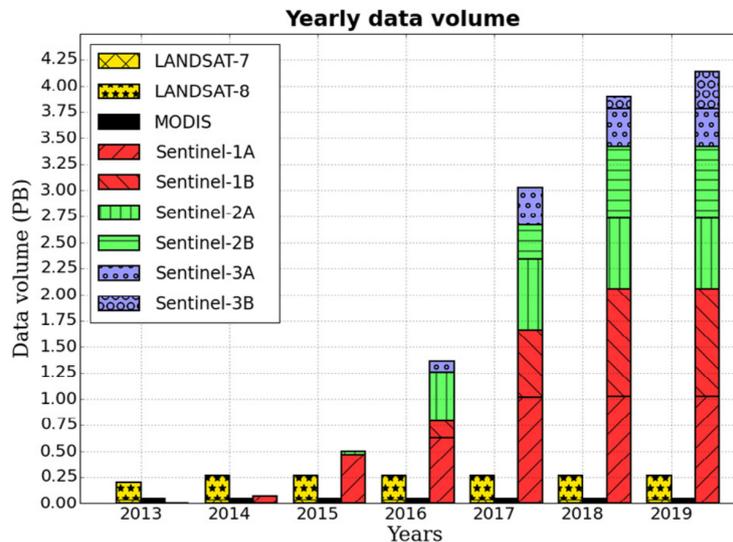
<sup>2</sup><https://ec.europa.eu/jrc/en/scientific-tool/global-land-cover>

<sup>3</sup>[http://due.esrin.esa.int/page\\_globcover.php](http://due.esrin.esa.int/page_globcover.php)

as tendências de dados abertos iniciada pelo programa CBERS, a USGS (*United States Geological Survey*) iniciou o processo de democratização do acervo de imagens da série Landsat (WULDER et al., 2012).

A adoção de políticas de dados abertos pelas agências espaciais e os avanços tecnológicos nos instrumentos de coleta de dados, refletiram no aumento de dados espaço-temporais disponíveis para utilização no monitoramento da Terra. A Figura 2.2 apresenta uma estimativa do volume de dados gerados pelas três primeiras missões Sentinel, as missões Landsat 7 e 8 e MODIS (*Moderate Resolution Imaging Spectrodiameter*) dos satélites TERRA e AQUA da NASA (*National Aeronautics and Space Administration*)/USGS. Anualmente o Landsat gera cerca de 0.25 *petabyte* de volume de dados e com o aumento dos dados gerado pelo programa Copernicus (Programa Europeu de Observação da Terra), desenvolvido pela ESA (*European Space Agency*), Soille et al. (2018) fez uma estimativa até 2019 sobre a geração de dados de todas as missões Sentinel, e volume de dados gerados ultrapassou *petabytes*.

Figura 2.2 - Estimativa do volume de dados livres por equipamento.



Fonte: Soille et al. (2018).

Esta crescente disponibilidade de dados traz diversos benefícios, como por exemplo, a diversidade de imagens com diferentes resoluções espaciais, radiométricas e temporais. Atualmente é possível avaliar e monitorar as mudanças no uso e cobertura

da terra em várias escalas espaciais e temporais. Isso possibilita aos pesquisadores construir modelos de mudanças de UCT e fornecer um suporte importante no monitoramento da cobertura terrestre ao longo do tempo (MACIEL et al., 2019).

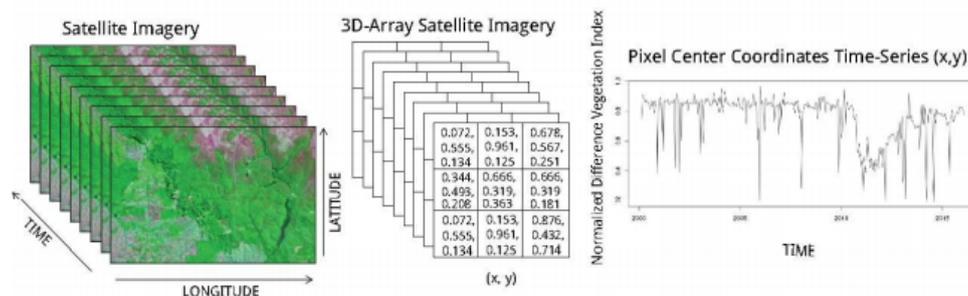
## 2.2 Geração de dados de uso e cobertura da terra

Além da evolução da tecnologia de aquisição dos dados, existe um grande esforço no desenvolvimento de métodos automáticos de classificação. Uma grande variedade de métodos de classificação supervisionados e não supervisionados são utilizados na geração de mapas de uso e cobertura da Terra. Recentemente, algoritmos de ML, que incluem árvores de classificação, Redes Neurais Artificiais e métodos de DL têm-se mostrado superiores a classificadores convencionais como o *Maximum Likelihood*, principalmente no contexto de grandes volumes de dados (ROGAN et al., 2003; ROGAN et al., 2008; DEFRIES; CHAN, 2000).

O uso de séries temporais derivadas de satélite de observação da Terra é uma das principais tendências de pesquisa na geração de mapas classificados de uso e cobertura da terra e na detecção de mudanças (PICOLI et al., 2018). Pelo fato dos satélites possuírem um período de revisita, as observações de um mesmo local podem ser organizadas nos respectivos intervalos que elas foram obtidas, de modo que seja mapeada em uma matriz tridimensional (espaço e tempo) (VINHAS et al., 2017).

Cada coordenada (x, y) em tempos consecutivos (t1, t2, ...) compõe uma série temporal de imagens de satélites. Na Figura 2.3 é apresentado um conjunto de imagens de satélite ordenado no espaço e tempo e sua respectiva matriz tridimensional. Dada uma coordenada de latitude e longitude (x,y) é possível extrair os valores dos pixels ao longo do tempo.

Figura 2.3 - Séries Temporais extraídas de imagens de sensoriamento remoto.



Fonte: Vinhas et al. (2017).

Conforme Picoli et al. (2018) apresentam, as séries temporais de sensoriamento remoto apresentam uma melhor performance conforme a resolução temporal do conjunto de dados aumenta. Ou seja, a relação temporal apresenta muita influência, capturando mudanças mais sutis de transição de uma cobertura para outra. As séries temporais de índices de vegetação, derivadas de informações espectrais de imagens de satélites de observação da Terra, como NDVI e EVI, são amplamente utilizadas.

Aguiar et al. (2010) identificaram áreas de pastagens e seus níveis de degradação no estado do Mato Grosso do Sul usando séries temporais NDVI (*Normalized Difference Vegetation Index*) e um classificador J48 com técnicas de *wavelet*. Arvor et al. (2011a) usaram séries temporais EVI (*Enhanced Vegetation Index*) para quantificar a evolução de áreas agrícolas de 2000 a 2006 no estado do Mato Grosso. Spera et al. (2014) utilizaram séries temporais EVI para examinar padrões de expansão de culturas agrícolas entre os anos de 2001 e 2011 no Estado do Mato Grosso usando algoritmos de árvore de decisão.

Estudos de séries temporais derivadas de imagens de sensoriamento remoto exige de uma infra-estrutura específica para armazenamento, gerenciamento, distribuição e processamento dos dados espaço-temporais (PICOLI et al., 2018). O Cubo de Dados de Observação da Terra (CDOBT) surgiram como um novo paradigma, sendo explorada em diversas pesquisas como uma possível solução para essas lacunas (GIULIANI et al., 2019). As Agências Espaciais têm buscado aplicar o paradigma de CDOBT como uma nova infraestrutura para compartilhar e gerenciar serviços e recursos de dados de OT. Dhu et al. (2017), Giuliani et al. (2017), Câmara et al. (2018) propõem implementações do paradigma de CDOBT. Este tipo de infraestrutura facilita o acesso aos dados e aplicações, sendo uma possível solução no desenvolvimento de sistemas operacionais que visam o monitoramento constante da superfície da Terra. As iniciativas de CDOBT lidam com diversos desafios para obter um cubo de dados organizado. Ao se tratar de imagens de SR, cada sensor trabalha em diferentes resoluções espaciais, temporais e radiométricas. Desta forma, é um desafio a produção de cubos de dados harmonizados que combinam diferentes sensores.

### **2.3 Trajetórias de uso e cobertura da terra**

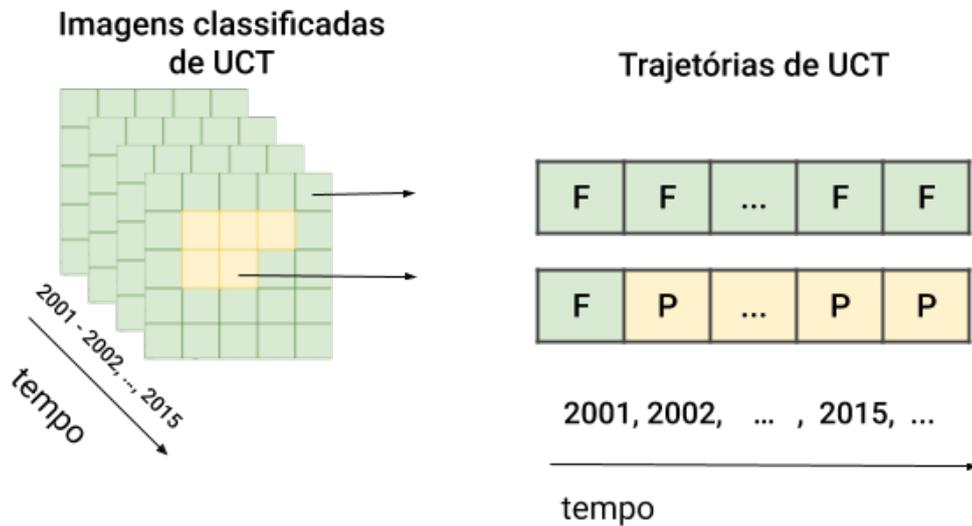
De maneira análoga às séries temporais de imagens de sensoriamento remoto, as trajetórias de UCT também são construídas a partir de um conjunto de imagens classificadas organizadas temporalmente. As trajetórias de UCT possuem diversas definições que variam conforme o autor, ou o propósito do estudo.

No trabalho de [Mertens e Lambin \(2000\)](#), os autores defendem que estudos anteriores apresentavam a trajetória apenas como uma conversão de uma categoria UCT para outra, ignorando a complexidade das mudanças no tempo. Desta forma, em seu trabalho eles definem o termo trajetória de uso e cobertura da terra como uma sucessão de tipos da terra dada uma determinada unidade amostral em mais de duas observações.

Segundo [Mena \(2008\)](#) as trajetórias de uso e cobertura da terra são a sequência temporal das classes de uso ou cobertura da terra no nível de pixel, descritas através de imagens classificadas reunidas em uma série temporal. E, conforme apresentado em [Wang et al. \(2012\)](#), trajetória pode ser expressa em códigos de vários tipos, como valores numéricos, letras, entre outros para representar os tipos de UCT para cada localização espacial.

O pixel na imagem classificada representa uma localização espacial e a classe de UCT. O intervalo temporal está associado à sequência de imagens classificadas de maneira ordenada. Também podem ocorrer casos em que cada banda de uma imagem classificada está associada a determinado tempo. A [Figura 2.4](#) apresenta um exemplo de representação de trajetórias de UCT extraídas de imagens classificadas ordenadas temporalmente. A letra F representa a classe FLORESTA, e a letra P representa a classe PASTO. Na [Figura 2.4](#), são apresentadas duas trajetórias: a primeira apresenta uma localização que não teve alterações ao longo do tempo. A segunda trajetória representa uma localização que em um primeiro instante de tempo era FLORESTA e nos instantes seguintes apareceu como Pasto.

Figura 2.4 - Exemplo de representação de Trajetórias de UCT extraídas de imagens classificadas.



Fonte: Adaptado de Maciel (2017).

Outro exemplo de representação é abordado no trabalho de Maciel (2017), onde o conjunto de trajetórias de UCT é apresentado como uma tabela, em que cada linha corresponde ao local (latitude, longitude) com sua classe de UCT e uma data de início e término.

Aplicações de trajetórias nas análises de mudanças de UCT têm ganhado atenção em diversas pesquisas (WANG et al., 2012; WANG et al., 2013; ZOMLOT et al., 2017; HUNAG et al., 2012; MACIEL et al., 2019). No estudo de Wang et al. (2012) foi desenvolvida uma metodologia que analisa dois aspectos de mudança. O primeiro realizou um análise de trajetórias de imagens binárias sobrepostas por diferentes fatores geográficos naturais. Nesta metodologia, o trabalho utilizou uma métrica de Intensidade de Mudança Relativa (RCI) para realizar comparações. A segunda metodologia utilizava métricas de padrão de trajetórias nas áreas de estudo.

Wang et al. (2013) trabalharam na detecção de mudanças na cobertura da terra com base em imagens de média e alta resolução espacial e análise de trajetória temporal. No trabalho de Hunag et al. (2012), o objetivo foi quantificar o padrão espaço-temporal das florestas e as trajetórias de mudança florestal.

O trabalho de Liu e Zhou (2004) apresentou uma abordagem onde utiliza-se as mu-

danças nas trajetórias de UCT como uma comparação pós-classificação de imagens Landsat classificadas. Como conclusão apresenta que ainda existe um grau de incerteza nesse tipo de avaliação, mas a abordagem oferece uma alternativa interessante quando a avaliação simultânea *ground truthing* não está disponível.

## 2.4 Sistemas de classificação

Cada projeto que gera informações sobre o uso e cobertura da terra desenvolve seu próprio sistema de classificação, com diferentes nomenclatura e significados das classes utilizadas para apresentar informações sobre diferentes classes cobertura da terra.

A classificação é o mecanismo em um sistema de classificação para oficializar determinada biofísica da superfície da terra ou sua utilização. A classificação é feita com base em um conjunto de regras que produz classes e subclasses, que mapeiam todo o domínio da informação da terra (JANSEN; GREGORIO, 2000).

Conforme apresentado em Giri (2016) e Jansen e Gregorio (2000), um sistema de classificação deve exibir as seguintes propriedades:

- uso de princípios classificatórios consistentes, únicos e aplicados sistematicamente;
- adaptado para descrever completamente toda a gama de tipos de recursos e seus detalhes;
- completude, fornecendo cobertura total do mundo que descreve;
- as classes derivadas são todas únicas, mutuamente exclusivas e inequívocas.

Anderson et al. (1976) apresentaram legendas de classificação que definiam categorias de uso e coberturas do solos derivadas de produtos de sensoriamento remoto. Este trabalho tornou-se padrão para o mapeamento da cobertura da terra.

A Comissão Europeia implementou o programa CORINE (*Coordination of Information on the Environment*). Este programa utiliza dados do Landsat e SPOT para mapear a cobertura da terra e fornece dados contínuos para a maioria da União Europeia. O CORINE desenvolveu uma hierarquia de classes para gerar um sistema de classificação para atender o propósito do programa (GIRI, 2016).

A medida que mais dados de uso e cobertura da terra são disponibilizados, há também maiores variações nas definições de classes destes mapas, surgindo obstáculos significantes no uso conjunto desses dados. Existem diversos estudos e técnicas que visam verificar a similaridade entre classes para obter sistemas de classificação harmonizados e assim realizar análises de diversos produtos de mapeamento. E deste modo, possibilitar comparações de diversas fontes de informações.

O LCCS (*Land Cover Classification System*) foi um passo importante dado pela FAO (*Food and Agriculture Organization*), representando o primeiro avanço desde o sistema do Anderson et al. (1976). LCCS foi publicado nos anos 2000 para atender à necessidade de uma padronização de sistema de classificação global de referência (JANSEN; GREGORIO, 2000). O sistema utiliza um conjunto de critérios independentes que permitem correlação com classificações e legendas existentes, fornecendo uma ferramenta poderosa para o desenvolvimento de produtos de cobertura da terra consistentes e comparáveis.

Conforme Jansen e Gregorio (2000) apresentam, a FAO enviou ao Comitê Técnico da ISO 211 o LCCS, como estabelecimento de um padrão internacional para sistemas de classificação de cobertura da terra. Como resultado, foi desenvolvido dois padrões: o **ISO 19144-1 Classification Systems**, para tratar de sistemas de classificação em geral e o **ISO 19144-2 Land Cover Meta Language**, expresso como um metamodelo UML, permite que diferentes sistemas de classificação de cobertura da terra sejam descritos. Ambos aprovados como padrões internacionais da ISO em 2012.

Foi desenvolvida a ferramenta LCML (*Land Cover Meta Language*) com o objetivo de produzir conjuntos de dados globais, regionais e nacionais que possam ser combinados em diferentes escalas, nível de detalhe localização geográfica. A ferramenta LCML fornece uma estrutura de regras que apresenta condições exclusivas para a criação de legendas (GREGORIO, 2016).

Apesar de não existir um consenso, ou nenhuma regra específica de harmonização, essas e diversas outras iniciativas proporcionaram o uso sinérgico dos dados, possibilitando encontrar a compatibilidade de sistemas de classificação e suas classes. Os sistemas de classificação normalmente são desenvolvidos com base no objetivo de um estudo. Desta forma, o sistema de classificação se torna subjetivo. Podem ocorrer casos onde uma mesma classe possui significados diferentes entre dois ou mais sistemas de classificação. Neste sentido, medir a similaridade semântica entre as categorias é muito necessário (FENG; FLEWELLING, 2004).

Várias abordagens para calcular a similaridade podem ser encontradas na literatura. Fornecendo informações importantes para lidar com o problema de utilizar um conjunto de dados de UCT existente para alguma outra aplicação específica (FENG; FLEWELLING, 2004).

No cenário nacional, muitas pesquisas fazem o uso do Manual Técnico do IBGE (Instituto Brasileiro de Geografia e Estatística) para estruturas um sistema de classificação de uso e cobertura da terra em seus trabalhos. A ideia central deste manual é servir como um referencial inicial para mapeamento e produção de informações de uso e cobertura da terra de forma a manter uma uniformidade em trabalhos situados nestes contexto (IBGE, 1999).

## **2.5 Programas de monitoramento ambiental nacional**

### **2.5.1 PRODES**

O PRODES é um projeto desenvolvido pelo INPE em colaboração com MMA e o IBAMA (Instituto Brasileiro do Meio Ambiente e dos Recursos Naturais Renováveis), que realiza o monitoramento de desmatamento por corte raso (retirada completa da cobertura florestal) na Amazônia Legal. O PRODES fornece informações anuais de desmatamento na Amazônia Legal desde 1988, auxiliando o governo na tomada de decisões em políticas públicas (INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS - INPE, 2018).

Para realizar o monitoramento esse projeto utiliza imagens de satélites Landsat-5, Landsat-7, Landsat-8, CBERS-2, CBERS-2B, Resourcesat-1 e UK2-DMC (INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS - INPE, 2018). As taxas são estimadas com base no incremento do desmatamento identificado no ano, ou seja, uma vez uma região identificada como desmatamento, essa área não é contabilizada nos anos posteriores.

São disponibilizados à comunidade todos os dados gerados pelo programa, na forma tabular e mapas vetoriais. Também são disponibilizadas as imagens de satélites utilizadas no mapeamento. Os dados vetoriais contém as informações referentes a classe principal atribuída à feição, são elas: Floresta, Não-Floresta, extensão desmatada (acumulado dos desmatamentos dos anos anteriores), desmatamento do ano do mapeamento, Hidrografia, Nuvem e Resíduo (CÂMARA et al., 2013).

### 2.5.2 DETER

O DETER é um sistema de alerta desenvolvido desde maio de 2004. Ele surgiu como um programa complementar ao PRODES, para suprir a necessidade de ações de prevenção e fiscalização em tempo real do desmatamento. São produzidos diariamente dados vetoriais com tempo de observação associado. Os dados produzidos pelo DETER são enviados ao IBAMA, visando apoiar a vigilância e controle do desmatamento. Para isto, são mapeadas áreas maiores que 25 hectares onde tenha ocorrido alguma mudança, como o corte raso ou outro tipo de degradação. Para realizar este mapeamento são utilizados os dados do sensor MODIS do satélite Terra (INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS - INPE, 2019).

Devido a alterações no padrão de desmatamento, tornou-se necessário o desenvolvimento de um novo projeto que atendesse os novos padrões, pois a maioria dos polígonos de desmatamento mapeado pelo PRODES possui área unitária menor que 25 hectares.

O DETER-B surgiu para preencher esta lacuna mapeando a cobertura florestal com área mínima próxima a 1 hectare. O DETER-B utiliza os dados do sensor WFI do satélite CBERS-4 e AWiFS do satélite Resourcesat-2, que possuem respectivamente uma resolução espacial de 64 e 56 metros (INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS - INPE, 2019). Neste projeto são mapeadas as classes: desmatamento com solo exposto, desmatamento com vegetação, mineração, degradação, cicatriz de incêndio florestal, corte seletivo desordenado e geométrico.

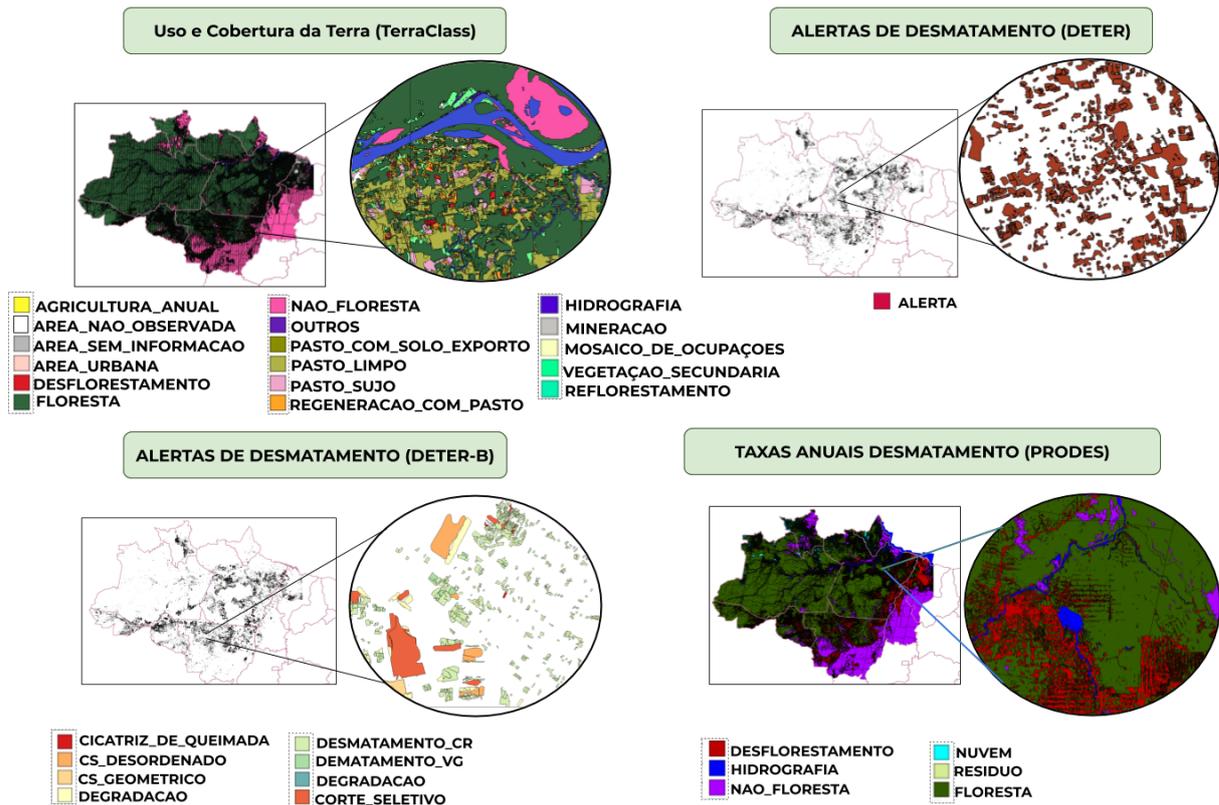
### 2.5.3 TerraClass

O Projeto TerraClass desenvolvido pelo INPE em parceria com a Embrapa, classifica o uso e cobertura da terra das áreas de desmatamento obtidas pelo PRODES. O objetivo é investigar sobre a dinâmica do desmatamento na região da Amazônia Legal, ou seja, investigar para qual finalidade as áreas são desmatadas com o intuito de obter um melhor entendimento do uso e cobertura da terra nesta região (ALMEIDA et al., 2016).

O Programa TerraClass produz mapas temáticos que apresentam as seguintes classes em sua primeira versão: Agricultura anual, Área não observada, Área urbana, Mineração, Mosaico de ocupações, Pasto com solo exposto, Pasto limpo, Pasto sujo, Regeneração com pasto, Reflorestamento, Vegetação secundária, e Outros. Os dados resultantes estão disponíveis para comunidade em formato matricial e vetorial, com

os mapeamentos dos anos de: 2004, 2008, 2010, 2012 e 2014 (INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS (INPE), 2014). A Figura 2.5 apresenta uma visão geral dos programas descritos.

Figura 2.5 - Programas de Monitoramento Ambiental do INPE.



Fonte: Produção da Autora.

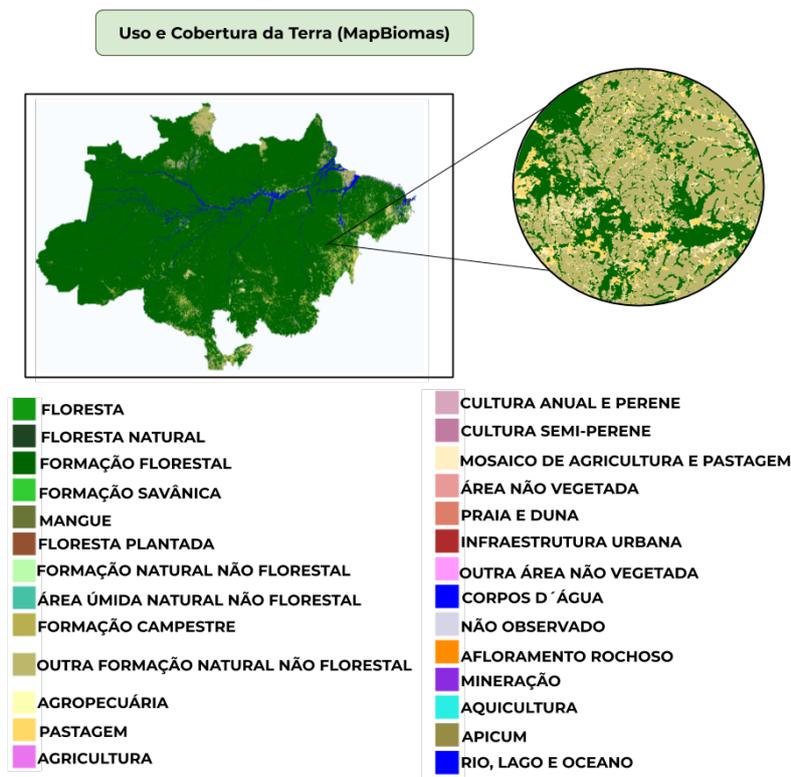
#### 2.5.4 MapBiomass

O Projeto de Mapeamento Anual da Cobertura e Uso do Solo no Brasil (MapBiomass) foi lançado em 2015 com o objetivo de contribuir para entender a dinâmica do uso e cobertura da terra no Brasil (SOUZA et al., 2020). Os dados produzidos por este projeto são divididos em Coleções. Até a data deste trabalho, a coleção de dados encontra-se na versão 5. As coleções 3, 3.1 possuem dados de 1985 até 2017 e a coleção 4 e 4.1 possui uma série histórica de 1985 até 2018. A coleção 5 possui

dados de 1985 até 2019.

Este projeto possui como metodologia a classificação automática pixel a pixel de imagens do *Landsat Data Archive* (LDA), utilizando métodos de aprendizado de máquina, como *Random Forest* e *Deep Learning*. Todo o processo é feito na plataforma do Google Earth Engine. A Figura 2.6 apresenta uma visão geral dos dados da coleção 3.1 do MapBiomias.

Figura 2.6 - Projeto MapBiomias coleção 3 da série anual de mapas de cobertura e uso da terra do Brasil do bioma Amazônia.



Fonte: Produção da Autora.

## 2.6 Serviços Web Geospaciais

Os serviços web geospaciais são uma importante ferramenta para a interoperabilidade, integração e compartilhamento de dados geospaciais (VINHAS et al., 2017). Eles trazem novas oportunidades para acessar grandes volumes de dados de ob-

servação da Terra através da Internet e permitem o processamento dos dados no servidor. A OGC (*Open Geospatial Consortium*) desempenha um papel importante neste cenário. Ela é um consórcio internacional de empresas, agências governamentais, organizações de pesquisa e universidades, que desenvolve especificações para facilitar a troca de informações geoespaciais. A OGC é responsável por definir padrões de serviços da web para visualizar, disseminar e processar dados geoespaciais (VINHAS et al., 2017), entre eles o **WMS** (*Web Map Service*), **WFS** (*Web Feature Service*), **WCS** (*Web Coverage Service*), **WCPS** (*Web Coverage Processing Service*) e **WPS** (*Web Processing Service*).

Vários produtos de uso e cobertura da terra são disponibilizados por meio desses serviços para facilitar o acesso aos dados geoespaciais. Os dados geoespaciais são representados em estrutura matricial, conhecida como *raster*, ou na estrutura vetorial.

A Tabela 2.1 apresenta os dados de geoespaciais suportados pelas especificação WCS, WMS e WCS, e os formatos suportados na resposta da requisição pelo cliente (WAGEMANN et al., 2018; BEST et al., 2007).

Tabela 2.1 - Tipos de Dados dos Padrões Serviços Web da OGC.

Padrão	Tipos de dados de entrada	Tipos de dados de saída
WMS	Mapas(Raster e Vetorial)	PNG, GIF, TIFF, SVG.
WFS	Dados Vetoriais (pontos, linhas, polígonos)	XML (GML),Shapefile,JSON,CSV.
WCS	Raster	JPEG, PNG,GeoTiff, ArcGrid.

O **WMS** define uma interface HTTP para solicitar imagens de mapas com informações geográficas (COMMITTEES, 2020c). Dada uma área de interesse, esta solicitação retorna o mapa, por exemplo no formato JPEG, que pode ser exibido em diversas aplicações.

O **WFS** define a forma como as informações geográficas vetoriais são criadas, modificadas e acessadas na Internet. Para isto são definidas onze operações que recuperam e descrevem as informações nos formatos XML e GML (*Geography Markup Language*) ou em outros formatos como o GeoJSON (COMMITTEES, 2020b).

O **WCS** oferece suporte para o intercâmbio de dados geoespaciais do tipo *coverage*. E ao contrário do WMS que retorna a representação em forma de imagens, o WCS trabalha diretamente com o dado. A interface WCS especifica três operações que

podem ser chamadas por um cliente WCS e executadas por um servidor WCS: `GetCapabilities`, `DescribeCoverage` e `GetCoverage` (COMMITTEES, 2020a).

Como o uso de séries temporais de sensoriamento remoto tem crescido, surgiu a necessidade de um novo serviço para lidar com a extração de informações no formato *time-first, space-later*. Neste sentido, foi proposto e implementado o **WTSS** (*Web Time Series Service*), um serviço web para o manuseio de séries temporais de imagens de sensoriamento remoto (QUEIROZ et al., 2015; VINHAS et al., 2017). Esse trabalho baseia-se na hipótese de que um serviço construído especificamente para recuperação de séries temporais é mais efetivo para acessar e recuperar essa informação do que serviços gerais baseados no padrão WCS.

A API definida por esse serviço é composta por três operações: `list_coverages`, `describe_coverage` e a operação `time_series`. A operação `time_series` é responsável por retornar uma série temporal a partir de uma dada localização no espaço e um intervalo de tempo. Para cada série temporal que se deseja obter devem ser especificados: o nome da *coverage*, os atributos de interesse para os quais desejam-se as séries de valores, a localização no espaço definido pelos parâmetros latitude e longitude em grau decimal adotando-se o Datum WGS84 (código EPSG 4326). E o intervalo de tempo (data inicial e final). Para realizar o intercâmbio de informações, o WTSS utiliza o formato JSON (*JavaScript Object Notation*) (QUEIROZ et al., 2015).



### 3 PLATAFORMA PARA ACESSO E ANÁLISE DE TRAJETÓRIAS DE USO E COBERTURA DA TERRA

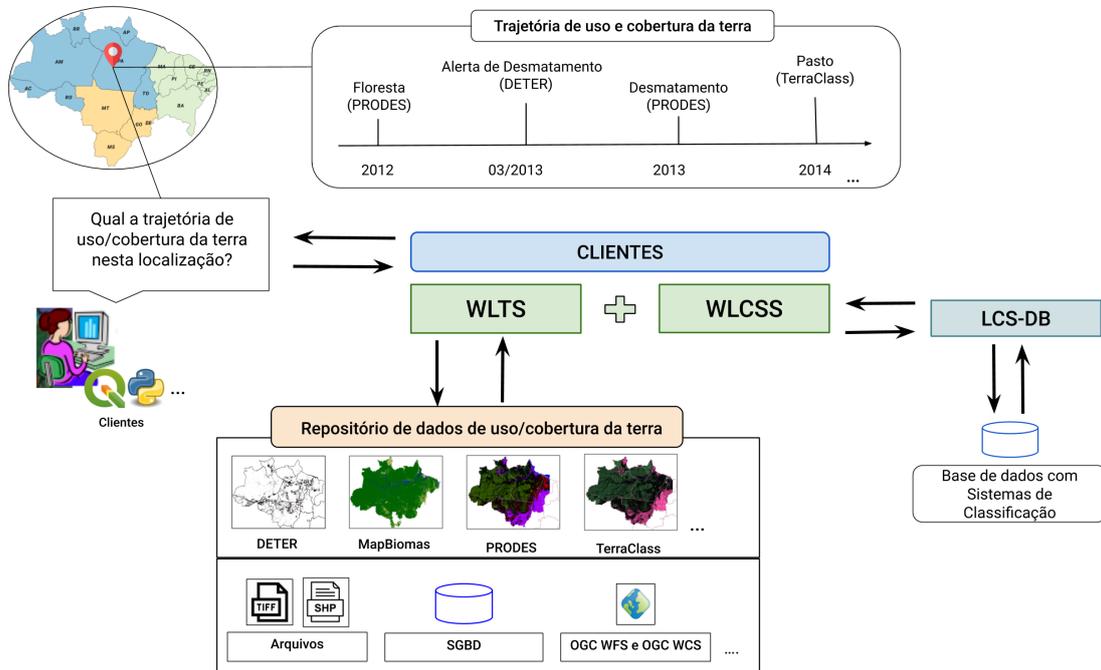
Neste trabalho é proposta uma Plataforma para Acesso e Análise de Trajetórias de Uso e Cobertura da Terra que inclui dois novos serviços web - o *Web Land Trajectory Service* (WLTS) e o *Web Land Classification System Service* (WLCSS), além de um banco de dados contendo coleções correspondentes aos principais produtos de mapeamento de UCT para o Brasil. O principal objetivo dessa plataforma é facilitar o uso dessas coleções, simplificando o acesso e a análise de trajetórias de UCT, em suporte às atividades de pesquisa e tomadas de decisão.

Através de uma camada de abstração de dados que incorpora o conceito de trajetórias de UCT, essa plataforma busca resolver as questões de interoperabilidade semântica e as diferenças nas representações espaciais e temporais dos dados fornecidos por ela. No que segue, será mostrado como a disponibilidade desses serviços e de bibliotecas para estes, em linguagens de script como Python e R, fornece um ambiente para que os usuários explorem os dados sem a necessidade de construir consultas complexas ou de conhecer detalhes técnicos dos formatos utilizados pelas diversas fontes de dados que compõem o banco de dados da plataforma.

#### 3.1 Visão Geral da Plataforma

A Figura 3.1 apresenta a visão geral da plataforma proposta sob a ótica de um usuário, que requisita uma trajetória de UCT associada a uma dada localização no espaço geográfico.

Figura 3.1 - Visão geral da plataforma para acesso e análise de trajetórias de uso e cobertura da terra.



Fonte: Produção da Autora.

Como pode ser observado na Figura 3.1, a plataforma possui cinco componentes principais:

- **LCS-DB:** fornece um modelo de dados capaz de representar os diversos sistemas de classificação, hierárquicos ou não-hierárquicos, e suas respectivas classes. Seu objetivo é normalizar a forma de representação empregada nas diversas coleções de dados de UCT disponíveis no repositório de dados da plataforma.
- **WLCSS:** serviço web, que disponibiliza as informações sobre os sistemas de classificação através de uma API especializada, capaz de fornecer: a lista de sistemas de classificação registrados na plataforma, a lista de classes de cada sistema, possíveis associações entre classes, e a simbologia empregada em cada sistema de classificação.
- **WLTS:** serviço web com o propósito de facilitar o acesso às várias coleções

de dados de UCT através de uma API projetada com base no conceito de trajetórias de UCT. A abordagem do WLTS é usar um modelo de dados que define um conjunto mínimo de informações temporais e espaciais para representar diferentes fontes e tipos de dados. O WLTS pode ser usado em diversas aplicações, como na validação de conjuntos de dados de cobertura do solo, na seleção de amostras de treinamento para dar suporte aos algoritmos de ML supervisionados usados na geração de novos mapas de classificação.

- **Repositório de dados:** conjunto de dados de UCT provenientes de diversos projetos: PRODES, DETER, TerraClass, Mapbiomas.
- **Clientes:** implementações do serviço em linguagens de script como Python e R para suporte à construção de aplicações e uso em ambientes de computação interativa como IPython/Jupyter (PÉREZ; GRANGER, 2007).

### 3.2 *Land Classification System Database Model* LCS-DB

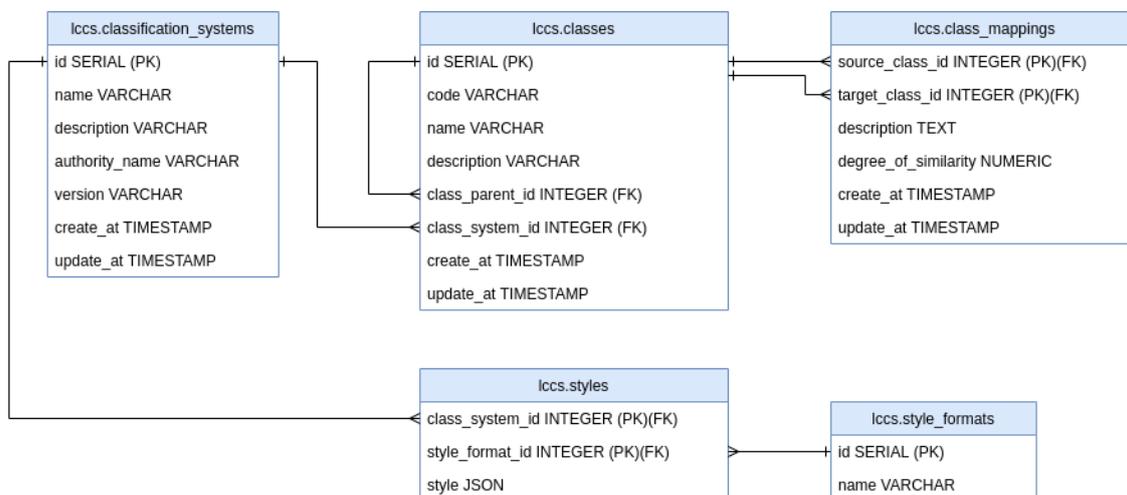
Para trabalhar com a reusabilidade dos dados, é importante realizar a harmonização dos sistemas de classificação. O modelo de dados proposto no LCS-DB e o serviço WLCSS foram projetados considerando essas questões, entretanto eles não visam resolver problemas de interoperabilidade semântica, fornecendo mapeamentos entre os sistemas de classificação, mas fornecem um ambiente de apoio aos pesquisadores e especialistas para ajudar na resolução dessas lacunas.

O LCS-DB fornece suporte à:

- Sistemas de classificação hierárquicos ou não-hierárquicos;
- Relacionamentos entre sistemas de classificação, com medidas de similaridade entre as classes;
- Estilo empregado nas legendas de cada sistema.

A Figura 3.2 apresenta o modelo de dados proposto no LCS-DB.

Figura 3.2 - Modelo de dados do LCS-DB.



Fonte: Produção da Autora.

Todas as entidades possuem os atributos *create\_at* e *update\_at*, para obter um controle sobre a manipulação dos dados no banco.

As entidades **lccs.classification\_systems** e **lccs.classes** são o tema central deste modelo de dados. Todos os outros conceitos capturados nesse diagrama partem da entidade **lccs.classification\_systems** na parte superior esquerda. Essa entidade representa os sistemas de classificação. Nela é possível definir um nome para o sistema, uma descrição, a versão do sistema e a autoridade responsável (ou criadora) pelo sistema de classificação.

As classes são definidas na entidade **lccs.classes**. Para suportar sistemas de classificação hierárquicos, a entidade **lccs.classes** possui um auto relacionamento, definido pelo atributo *class\_parent\_id*.

A entidade **lccs.class\_mappings** é responsável pelo suporte à harmonização entre sistemas de classificação. Nesta entidade são definidas as classes de origem (atributo *source\_class\_id*) do sistema de classificação A, e a classe destino de um sistema de classificação B.

Para exemplificar esta harmonização, suponha dois sistemas de classificação: PRODES e TerraClass inseridos no modelo LCS-DB. A Tabela 3.1 apresenta os dados ao

serem inseridos na tabela **lccs.classification\_systems**, com os atributos *create\_at* e *update\_at* omitidos.

Tabela 3.1 - Dados inseridos na tabela **lccs.classification\_systems** proveniente do LCS-DB.

<b>id</b>	<b>name</b>	<b>authority_name</b>	<b>description</b>	<b>version</b>
1	PRODES	INPE	Projeto Monitoramento Anual de Desmatamento	1
2	TerraClass	INPE	Mapeamento do Uso e Cobertura da Terra	1

Para estes sistemas de classificação, as classes apresentadas na Figura 3.2 foram inseridas na tabela **lccs.classes**. Alguns atributos foram omitidos.

Tabela 3.2 - Dados inseridos na tabela **lccs.classes** proveniente do LCS-DB.

<b>id</b>	<b>name</b>	<b>class_system_id</b>
1	Desflorestamento	1
2	Floresta	1
3	Hidrografia	1
4	Não Floresta	1
5	Nuvem	1
6	Resíduo	1
...	...	...
31	Agricultura Anual	2
32	Área Não Observada	2
33	Área sem Informação	2
34	Área Urbana	2
35	Desflorestamento	2
36	Floresta	2
37	Hidrografia	2
38	Mineração	2
39	Mosaico de Ocupações	2
40	Não Floresta	2
41	Outros	2
...	...	...

Para realizar a harmonização do sistema de classificação do projeto do TerraClass para o sistema de classificação do projeto PRODES, deve-se adicionar o *id* das classes nos atributos de *source\_class\_id* e *target\_class\_id* da tabela **lccs.class\_map**

**pings.** As classes de Agricultura Anual, Desflorestamento, Mineração do TerraClass são mapeadas para a classe Desmatamento do Prodes.

Tabela 3.3 - Dados inseridos para o mapeamento na tabela **lccs.classes** provenientes do modelo LCS-DB.

<b>source_class_id</b>	<b>target_class_id</b>
31	1
35	1
38	1
36	2
...	...

A entidade **lccs.style\_formats** permite cadastrar formatos de estilos (QGIS, TerraView, GeoServer). O arquivo de estilo em si fica armazenado na tabela **lccs.styles**. Desta forma, cada sistema de classificação pode ser associado a vários estilos.

### 3.3 *Web Land Classification System Service (WLCSS)*

O *Web Land Classification System Service* (WLCSS) foi projetado para lidar com representações do sistema de classes nos mapas de UCT. Além disso, pode facilitar o acesso das aplicações às simbologias dos mapas classificados, como por exemplo, do GeoServer e QGIS. O serviço WLCSS provê uma interface simples para utilizar o modelo de dados do LCS-DB e realizar consultas ao banco de dados.

A API definida por este serviço suporta as operações dos sistemas de classificação, classes, estilos e mapeamentos. Para isto, seis rotas foram definidas, que retornam como resposta um documento JSON. As rotas do WLCSS são navegáveis através de *links*.

Os *links* descrevem os relacionamentos no WLCSS. A estrutura de *links* é composta por *URLs* e relacionamentos estruturais (*rel*). Os tipos de *rel* incluem *root*, *parent*, *child*, *classification\_systems*, *classification\_system* e *mappings*. A Tabela 3.4 apresenta a descrição dos tipos de campos presentes nos *links*.

Tabela 3.4 - Campos presentes nos *links* do WLCSS.

Nome do Campo	Tipo	Descrição
<i>href</i>	<i>String</i>	REQUERIDO. O link real no formato de um URL.
<i>rel</i>	<i>String</i>	REQUERIDO. Relação entre o documento atual e o documento vinculado.
<i>type</i>	<i>String</i>	Tipo de mídia da entidade referenciada.
<i>title</i>	<i>String</i>	Título do <i>link</i> .

Fonte: Adaptado de [Foundation \(2020\)](#).

A seguir são apresentadas as especificações das operações suportadas pelo serviço WLCSS para lidar com os Sistemas de Classificação, Classes, Mapeamentos e Estilos.

### Sistemas de Classificação

`/classification_systems`: Recupera a lista de nomes de todos os sistemas de classificação disponíveis em uma instância do serviço WLCSS. Nenhum parâmetro é passado para esta operação. A Listagem 1 apresenta um exemplo de retorno dessa operação.

---

```

1 {
2   "links": [
3     {
4       "href": "http://localhost/classification_system",
5       "rel": "child",
6       "title": "Child"
7     }
8   ]
9 }
```

---

Listagem 1 - Definição de retorno da operação `classification_systems`.

`/classification_systems/system_id`: Dado o identificador (nome) de um sistema de classificação, essa operação retorna as informações do sistema de classificação. Este identificador é um parâmetro obrigatório, representado pelo `system_id`. Essa operação retorna as seguintes informações apresentadas na Listagem 2.

---

```

1 {
2   "id": "number",
3   "name": "string",
4   "authority_name": "string",
5   "description": "string",
6   "version": "string",
7   "styles": [
8     {
9       "href": "http://localhost/system_id/styles/style_format",
10      "rel": "item",
11      "title": "Styles"
12    }
13  ],
14  "links": [
15    {
16      "href": "http://localhost/",
17      "rel": "item"
18    }
19  ]
20 }

```

---

Listagem 2 - Definição de retorno da operação que descreve um sistema de classificação.

## Classes

`/classification_systems/system_id/classes`: Retorna todas as classes de um sistema de classificação. Esta operação também requer o Identificador (nome) do sistema de classificação.

Essa operação segue o mesmo padrão de retorno apresentado na Listagem 1. Um *array* que contém os links para todas as classes do sistema de classificação desejado.

`/classification_systems/system_id/classes/class_id`: Busca a classe de um sistema de classificação. A Tabela 3.5 apresenta os parâmetros requeridos dessa operação.

Tabela 3.5 - Parâmetros da operação que retorna informações sobre uma única classe.

Parâmetro	Descrição
system_id	Identificador (nome) de um sistema de classificação.
class_id	Identificador (nome) da classe.

A Listagem 3 apresenta um exemplo de resposta para essa operação.

---

```
1 {
2   "id": "number",
3   "name": "string",
4   "code": "string",
5   "description": "string",
6   "parent": {},
7   "links": [
8     {
9       "href": "http://localhost/classification_system",
10      "rel": "child",
11      "title": "Child"
12    }
13  ]
14 }
```

---

Listagem 3 - Definição de retorno da operação que busca uma única classe.

## Mapeamentos

`/mappings/system_id_source`: Recupera os mapeamentos disponíveis para um sistema de classificação. Requer como parâmetro o identificador (nome) de um sistema de classificação. Essa operação segue o mesmo padrão de retorno apresentado na Listagem 1. É retornado um *array links* que contém os links de todos os mapeamentos disponíveis.

`lccs/mappings/system_id_source/system_id_target`: Recupera os mapeamentos entre dois sistemas de classificação. A Tabela 3.6 apresenta os parâmetros requeridos dessa operação.

Tabela 3.6 - Parâmetros da operação com os mapeamentos de sistemas de classificação.

Parâmetro	Descrição
<code>system_id_source</code>	Identificador (nome) de um sistema de classificação de origem.
<code>system_id_target</code>	Identificador (nome) de um sistema de classificação de destino.

## Estilos

`/classification_systems/system_id/styles`: Busca os estilos de um sistema de classificação. Esta operação requer como parâmetros o identificador (nome) de um sistema de classificação. Essa operação segue o mesmo padrão de retorno apresentado na Listagem 1, onde tem-se os *links* com todos os estilos disponíveis para um dado sistema de classificação. Para cada sistema de classificação são apresentados os estilos cadastrados na tabela `style_format` presente no modelo do LCS-DB (3.2).

`/classification_systems/system_id/styles/style_format`: Busca o estilo pertencente a um sistema de classificação e um formato de estilo. Desta forma, requer como parâmetro o identificador (nome) de um sistema de classificação e o identificador (nome) de um formato de estilo. A operação retorna o arquivo de estilo em si. O usuário deve escolher o local para salvar o arquivo em sua máquina e então o *download* é realizado.

### 3.4 Web Land Trajectory Service (WLTS)

#### 3.4.1 Definições

O WLTS traz o conceito de trajetória de uso e cobertura da terra como uma abstração de alto nível, onde é possível expressar mudanças ao longo do tempo de maneira categórica. Cada trajetória representa as classes de uso e cobertura da terra das respectivas coleções, de uma localização espacial específica  $(x, y)$ . Cada elemento de uma trajetória é representado por uma observação em uma data específica, sendo a trajetória uma ordenação temporal destes elementos. As classes de UCT atendem à definição específica de cada fonte de dados. A trajetória de UCT também traz a *collection* como informação a cada elemento da trajetória, especificando qual a fonte de dados relacionada.

Neste trabalho, *collection*, *class* e *trajectory* são definidos como:

**Definição 1.** *Collection* ( $C$ ) refere-se ao conjunto de dados de uma fonte (base de dados) específica, seja o conjunto formado por uma estrutura vetorial ou matricial (*raster*). Uma *Collection* possui um intervalo temporal delimitado pelos tempos  $(t_{min}, t_{max})$ . Desta forma, cada *Collection* tem um atributo temporal associado, que está alinhado conforme a granularidade de tempo de cada projeto que disponibiliza a *Collection*.

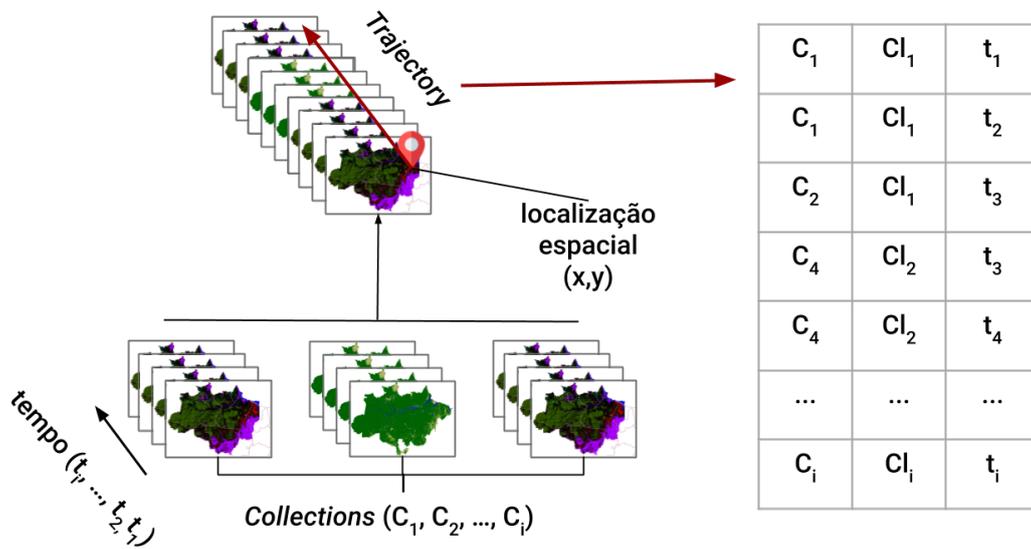
**Definição 2.** *Class* ( $Cl$ ) É o rótulo associado a um dado, que corresponde aos

tipos específicos de uso ou cobertura do solo, definido pelo sistema de classificação da fonte do dado. Uma *Collection* é constituída por um conjunto de *Class*. Desta forma,  $\forall Cl_i, Cl_i \in C$ .

**Definição 3.** *Trajectory* (Trj) Dada uma localização espacial  $(x, y)$ , uma trajetória *Trj* é representada por um conjunto de observações  $\tau$  na forma  $\{(C_1, Cl_1, t_1), \dots, (C_i, Cl_i, t_i)\}$ . Cada observação do conjunto  $\tau$  contém a classe (*C*) de uso e cobertura da terra e coleção (*Cl*) de uma localização  $x, y$  no tempo  $t$ . Temporalmente um trajetória é delimitada por  $(t_{min}, t_{max})$  que são os instantes de tempo mínimo e máximo associados a uma (*C*). Desta forma, uma trajetória *Trj* pode ser representado como  $\{(C_1, Cl_1, t_{min}), (C_n, Cl_n, t_{max})\}$ , dado que todas as observações de *Trj* satisfazem  $t_{min} < t < t_{max}$ .

A Figura 3.3 apresenta uma representação das definições apresentadas nesta Seção.

Figura 3.3 - Representação da definição de *Trajectory* no contexto do WLTS.

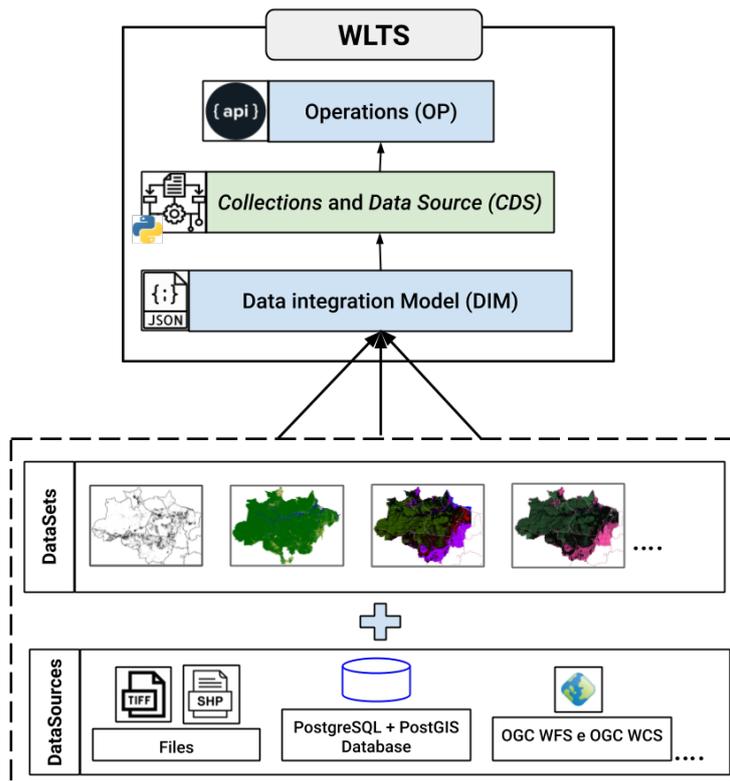


Fonte: Produção da Autora.

### 3.4.2 Arquitetura

Para o desenvolvimento do WLTS foi proposta uma arquitetura composta por três camadas, *Operations* (OP), *Collections e Data Source* (CDS) e *Data Integration Model* (DIM), como mostra a Figura 3.4.

Figura 3.4 - Arquitetura do WLTS.



Fonte: Produção da Autora.

O DIM é responsável por definir um conjunto mínimo de informações espaciais e temporais dos dados de UCT. As informações dos *datasets* são informadas com base nos tipos definidos pelo DIM, *Feature Collection* e *Image Collection*. Informações sobre fontes de dados (*datasource*) também devem ser informadas ao modelo com base no tipo *DataSource* do DIM.

### 3.4.3 Modelo de Integração de Dados

Para acessar as trajetórias de UCT de dados espaço-temporais provenientes de diferentes fontes, propomos como abordagem a utilização de um modelo de dados capaz de descrever diferentes fontes de dados que disponibilizam informações de UCT: Modelo de Integração de Dados, do inglês, *Data Integration Model* (DIM).

A premissa do DIM é que ao utilizar informações que descrevem como os *datasets* são armazenados e fornecidos, é possível realizar a integração dos dados, possibilitando resolver as questões de diferenças de estruturas de armazenamento, espaciais e temporais. Desta forma, as informações passadas a este modelo permite que a camada CDS consiga acessar diferentes tipos e fontes de dados espaço-temporais para realizar a extração de trajetórias UCT. É proposto expressar essas informações do DIM utilizando o formato JSON<sup>1</sup>.

O DIM define um conjunto mínimo de informações espaciais e temporais para fornecer um modelo genérico que suporte estruturas vetoriais e matriciais. São definidos no DIM três tipos de objetos: **datasources**, **feature\_collection** e **image\_collection**. Uma visão geral do DIM encontra-se no diagrama apresentado no Anexo A.2.

Normalmente, cada projeto que fornece mapas classificados de UCT disponibiliza suas informações através de diversas fontes de dados. São utilizados como disseminação arquivos no formato *ShapeFile*, *GeoTiff*, ou via serviços WFS, WCS, WMS. O **datasource** contém informações sobre essas fontes.

Definimos **datasource** como um tipo que possui como propriedade um *array*. Esse *array* é composto por ou mais objetos do tipo **datasource**, que podem incluir, mas não se limitar a: **webservice\_source**, **files\_sources**, **dbms\_sources**. Desta forma, cada tipo específico de **datasource** é descrito por um conjunto específico de atributos, necessários para realizar sua conexão. O **webservice\_source** descreve os *datasets* provenientes dos serviços OGC WFS e OGC WCS.

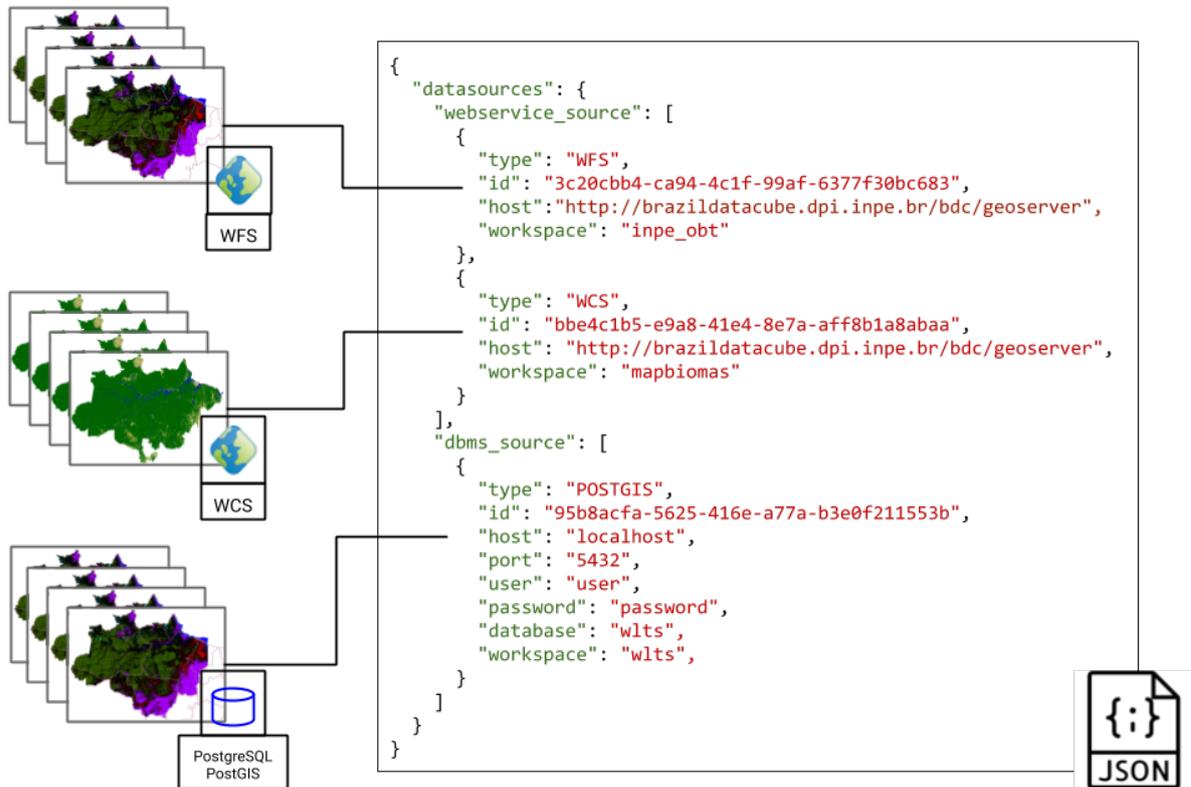
No objeto **webservice\_source**, deve ser informado o seu tipo de padrão (WFS ou WCS), o endereço de acesso, composto pelo *host*, *port*, *workspace*, e se existir usuário e senha. Cada **datasource** possui um identificador único que é utilizado para realizar a ligação com as *Collections*. A Figura 3.5 apresenta um exemplo de como as informações de cada tipo de **datasource** é preenchido. Neste exemplo,

---

<sup>1</sup><https://www.json.org/>

temos dois datasources do tipo `webservice_source` e um do tipo `dbms_source`.

Figura 3.5 - Exemplos de DataSource preenchidos no DIM.



Fonte: Produção da Autora.

Portanto o DIM tem a função de descrever como as `feature_collection` e `image_collection` são armazenadas e fornecidas pelos `datasources`. A `feature_collection` no DIM refere-se a uma ou mais estruturas vetoriais. A estrutura vetorial é composta por um ou mais pares de coordenadas que formam os elementos vetoriais pontos, linhas e polígonos, representam feições (*features*) únicas localizadas no espaço. Já uma `image_collection` refere-se a um conjunto de dados em estrutura matricial, ordenadas temporalmente. Os objetos `feature_collection` e `image_collection` possuem uma ou mais conjunto de dados.

É proposto um conjunto de atributos no DIM, necessários para descrever as *Collections*. De maneira geral, as seguintes informações devem ser informadas nos arquivos

nos objetos `feature_collection` e `image_collection`:

- Qual o tipo da `Collection`: Representação do tipo de dado, podendo ser `image_collection` ou `feature_collection`.
- O `datasource` proveniente: Deve ser passado como informação das `collection` o `datasource` proveniente.
- Propriedades temporais: A propriedade temporal deve conter o instante de tempo (por exemplo `DATE`), ou ele pode armazenar tempos como textos (`STRING`). Deve ser informada a resolução temporal, composta pela unidade (`unit`), e valor (`value`). A resolução temporal é definida pela frequência temporal da `Collection`.
- A propriedade espacial: Define o sistema de referência espacial e o tipo de representação geométrica do dado.
- Sistema de classificação: O identificador correspondente que foi previamente definido em um objeto `datasource`. Dever ser informado qual a propriedade representa a classe nesta `collection`.

Para o sistema de classificação, foram abordados alguns casos. Normalmente os *datasets* possuem as classes de UCT junto aos seus dados. Nos dados vetoriais, por exemplo, as classes podem estar definidas em uma coluna desse dado, em formato string ou podem conter um identificador (id) que é chave estrangeira para outra tabela.

Podem existir casos em que o sistema de classificação com as respectivas classes dos dados estão em um *datasource* separado. Neste caso devemos informar como acessar esse *datasource*. Um objeto *datasource* deve ser preenchido e seu identificador deve ser informado à coleção. Também deve ser informado como as propriedade das classes estão definidas. Essa situação pode ocorrer por exemplo com um dado matricial, onde o valor de cada pixel da matriz está associado a uma determinada classe. Esses casos permitem também que o WLTS realize a integração com o LCS-DB para consultar as classes de cada coleção.

#### 3.4.4 *Collections e Data Source*

Os dados informados no DIM nos tipos `datasources`, `feature_collections` e `image_collections` são materializados em classes na camada CDS do WLTS. O diagrama de classe da camada CDS do WLTS pode ser visualizado no Anexo [A.1](#).

Na camada CDS o WLTS possui dois tipos de classes abstratas: **Collections** e **DataSources**. As classes **FeatureCollection** e **ImageCollection** são especializações da classe **Collections**. Estas classes possuem funções que permitem organizar as informações que possibilitam acessar e retornar as trajetórias.

O WLTS foi modelado para ser extensível, onde um novo tipo de **datasource**, proveniente do DMI, pode ser adicionado ao serviço sem a necessidade de grandes alterações no código fonte. Para novos **datasource** serem adicionados, estes devem constar no DMI, e sua implementação deve ser realizada na camada CDS. Os novos **datasources** devem herdar da classe **DataSources** e implementar seus métodos.

O acesso e consulta dos dados ficam atribuídos às implementações da classe abstrata **DataSources**. Pois cada tipo de fonte de dados possui suas peculiaridade para acessar e retornar informações. O **WFSDataSource** utiliza a operação *GetFeature* para compor o retorno de trajetórias, enquanto o **WCSDataSource** utiliza o *GetCoverage*.

### 3.4.5 Operations

A camada *Operations* contém as operações disponíveis na API do WLTS. O WLTS adota o formato JSON para recuperar as informações. A API definida pelo serviço WLTS é composta por três operações. A seguir é apresentada a especificação das operações suportadas pelo WLTS com a descrição dessas operações, suas respostas e parâmetros.

**list\_collections:** Esta operação recupera a lista de *collections* disponíveis no serviço do WLTS.

Como resposta, esta operação retorna um *array* de *string*. Este *array* é composto pelos nomes da *collections* disponíveis no serviço WLTS. A Listagem 4 apresenta como o retorno desta operação está definida.

---

```
1 {  
2   "collections": [ "string" ]  
3 }
```

---

Listagem 4 - Definição de retorno da operação `list_collections`.

**describe\_collection:** Recupera os metadados de uma determinada *collection*. A

Tabela 3.7 apresenta os parâmetros suportados por esta operação.

Tabela 3.7 - Parâmetros da operação `describe_collections` no WLTS.

Parâmetro	Descrição	Requerido
<code>collection_id</code>	Identificador (nome) da coleção	Sim

Como resposta esta operação retorna um arquivo JSON com as informações de uma coleção (informada no parâmetro `collection_id`). A Listagem 5 apresenta a definição do retorno desta operação

```
1 {  
2   "name": "string",  
3   "description": "string",  
4   "detail": "string",  
5   "collection_type": "string",  
6   "resolution_unit": {  
7     "unit": "string",  
8     "value": 0  
9   },  
10  "period": {  
11    "start_date": "string",  
12    "end_date": "string"  
13  },  
14  "spatial_extent": {  
15    "xmin": 0,  
16    "xmax": 0,  
17    "ymin": 0,  
18    "ymax": 0  
19  }  
20 }
```

Listagem 5 - Definição de retorno da operação `describe_collection`.

## **trajectory**

Recupera as trajetórias UCT a partir das coleções de dados em uma localização no espaço. A resposta da operação *trajectory* é um documento JSON contendo duas partes: *query* e *result*. A chave *query* contém o contexto da consulta realizada. Na chave *result* encontra-se a trajetória de UCT para a localização informada. A Tabela 3.8 apresenta os parâmetros dessa operação.

Tabela 3.8 - Parâmetros da operação trajectory no WLTS.

Parâmetro	Descrição	Requerido
collections	Lista de identificadores (nome) de coleções delimitadas por vírgulas	Não
longitude	Coordenada de longitude	Sim
latitude	Coordenada de latitude	Sim
start_date	Data inicial da trajetória	Não
end_date	Data final da trajetória	Não

A Listagem 6 apresenta como é o arquivo de retorno da operação trajectory.

---

```

1 {
2   "result": {
3     "trajectory": [
4       {
5         "class": "string",
6         "collection": "string",
7         "date": "string"
8       }
9     ]
10  },
11  "query": {
12    "collections": [
13      "string"
14    ],
15    "longitude": 0,
16    "latitude": 0,
17    "start_date": "string",
18    "end_date": "string"
19  }
20 }
```

---

Listagem 6 - Definição de retorno da operação trajectory.

### 3.5 Implementação

O desenvolvimento destas ferramentas considerou um conjunto de tecnologias dentro de um cenário tecnológico de serviços Web. A linguagem Python<sup>2</sup> foi adotada como

<sup>2</sup><https://www.python.org/>

linguagem de programação primária de codificação. Python é um linguagem de alto nível amplamente utilizada na comunidade de SR. Considerando o cenário Web foram utilizados dois *frameworks* Flask<sup>3</sup> e Flask SQLALchemy<sup>4</sup>. Para auxiliar na construção de linhas de comando, foi utilizado o pacote Click<sup>5</sup>.

O GitHub<sup>6</sup> foi adotado para disponibilizar os pacotes e códigos desenvolvidos neste trabalho. O GitHub é uma das principais ferramentas para trabalhar com versionamento e repositório de códigos, além de possibilitar a integração de diversas tecnologias. Os repositórios de códigos disponíveis no GitHub podem ser visualizados nos Anexos C.1, C.2, C.3.

Travis CI<sup>7</sup> é um pacote de integração contínua de código-fonte que fornece um ambiente de execução de testes. E o COVERALLS<sup>8</sup> é um pacote para geração de informações e estatísticas sobre cobertura dos testes do ambiente Travis CI. Travis CI e COVERALLS são tecnologias integradas ao GitHub, que fornecem gratuitamente integração contínua para os repositórios públicos.

As especificações dos serviços WLCSS e WLTS foram definidas via *OpenAPI-Specification*<sup>9</sup>. E podem ser encontradas nos Anexos B.2 e B.1.

Para a geração automática de documentação foi utilizada a biblioteca Sphinx<sup>10</sup>. O Jupyter<sup>11</sup> foi adotado como ferramenta para apresentação de um modelo de documentação onde são apresentados exemplos de códigos das ferramentas desenvolvidas.

Todos os repositórios de códigos e documentação desenvolvidos neste trabalho podem ser encontrados no GitHub do Projeto Brazil Data Cube<sup>12</sup>.

---

<sup>3</sup><https://flask.palletsprojects.com/en/1.1.x/>

<sup>4</sup><https://flask-sqlalchemy.palletsprojects.com/en/2.x/>

<sup>5</sup><https://click.palletsprojects.com/en/7.x/>

<sup>6</sup><https://github.com/>

<sup>7</sup><https://travis-ci.org/>

<sup>8</sup><https://coveralls.io/>

<sup>9</sup><https://github.com/OAI/OpenAPI-Specification>

<sup>10</sup><https://www.sphinx-doc.org/en/master/>

<sup>11</sup><https://jupyter.org/>

<sup>12</sup><https://github.com/brazil-data-cube>



## 4 ESTUDO DE CASO E RESULTADOS

O modelo de integração de dados desenvolvido fornece uma abordagem que permite aos usuários do serviço adicionar variados tipos de dados e fontes, conforme a necessidade de cada aplicação em que se deseja utilizar o WLTS. O serviço WLCSS e o modelo de dado LCS-DB compõem um ambiente de apoio à resolução de questões de interoperabilidade semântica, questão importante quando se deseja trabalhar em análise ou reusos de diferentes fontes de dados.

### 4.1 Estudo de caso 1

Como prova de conceito, utilizamos os dados dos projetos PRODES, DETER, TerraClass e MapBiomas, dos biomas Amazônia Legal e Cerrado. Com exceção do TerraClass que possui dados publicados, até o momento, apenas para o bioma Amazônia. Esses projetos possuem um acervo de dados de UCT consolidados, com informações importantes para análises de trajetórias de UCT.

Os dados disponibilizados por esses projetos representam as variabilidades semânticas, espaciais e temporais que a plataforma visa resolver. Neste sentido, foi possível ilustrar o uso do DIM (*Data Integration Model*) para descrever as *collections* e *datasets* e, posteriormente, o mapeamento nos tipos de dados da camada CDS (*Collections and Data Sources*).

Os dados espaciais vetoriais dos projetos PRODES, DETER e TerraClass foram organizados e armazenados no Sistema Gerenciador de Bancos de Dados (SGBD) PostgreSQL com a extensão PostGIS. Essas coleções são disponibilizadas através da implementação WFS do aplicativo GeoServer. Os dados matriciais do projeto MapBiomas foram disponibilizados através da implementação WCS do aplicativo Geoserver. A Tabela 4.1 apresenta algumas das variabilidades destes projetos.

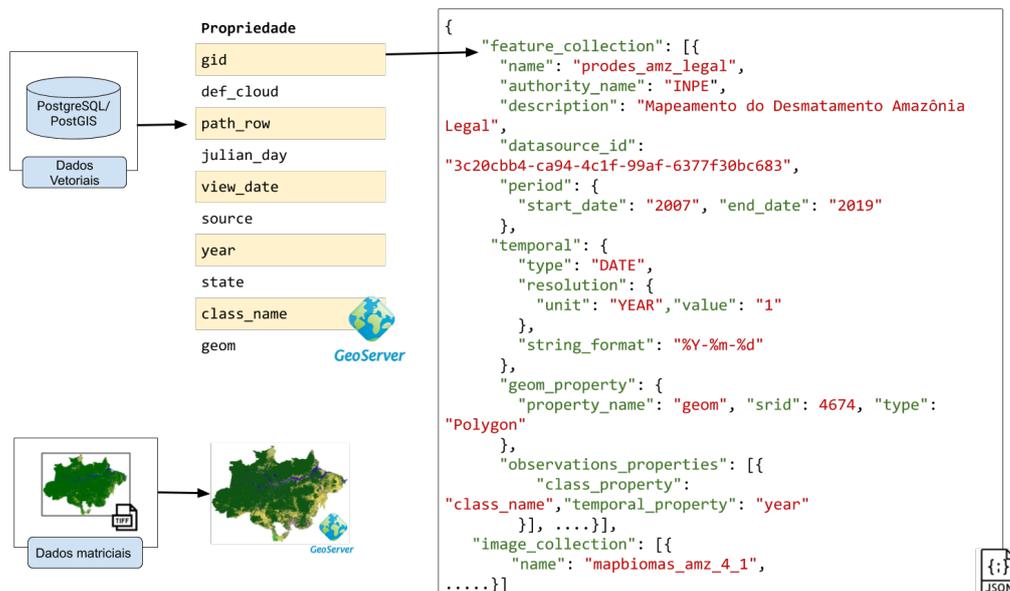
Tabela 4.1 - Informações dos *datasets* utilizados no estudo de caso.

<b>Projeto</b>	<b>Estrutura de Armazenamento</b>	<b>Resolução Temporal</b>	<b>Web Service Data Source</b>
PRODES	Vetorial	Anual	OGC WFS
DETER	Vetorial	Diário	OGC WFS
TerraClass	Vetorial	BiAnual	OGC WFS
MapBiomas	Raster	Anual	OGC WCS

Para que estes dados fiquem disponíveis via serviço WLTS para a análise de trajetórias de UCT, deve-se preencher as informações do DIM. Para as informações de Data Sources derivados de Web Services deve ser preenchida as informações de type (tipo do serviço OGC WFS ou WCS), o identificador (id) do *datasource*, a url (host) de acesso ao dado, e o *workspace* em que o dado se encontra no aplicativo do Geoserver. Caso existam, o usuário e senha também devem ser informados.

Cada bioma dos respectivos *datasets* é inserido no DIM como uma *collection*. A Figura 4.1 mostra um exemplo da estrutura de um dado vetorial armazenado na ferramenta GeoServer e as principais informações do DIM que a descreve. Nesta Figura, o conjunto de dados é proveniente do PRODES, nomeada de `prodes_amz_legal`. Esse *dataset* possui a coluna *year*, que contém o ano a que se refere a feição. As propriedades temporais para este caso são *type* DATE, com *resolution* YEAR e *value* 1.

Figura 4.1 - Exemplo do modelo proposto para integração de dados: DIM.



Fonte: Produção da Autora.

Os arquivos JSON preenchidos com base no DIM, descrevem os *datasets* e *datasources*, que são mapeados nos tipos de dados da camada CDS do WLTS para acessar as trajetórias de UCT destes acervos de dados. Após preencher o DIM, e uma ins-

tância do serviço WLTS estar disponível, o DIM fica transparente ao usuário que irá utilizar o serviço. Para visualizar essas informações, o usuário pode acessar a rota de `describe_collection`, e irá ter acesso às informações mínimas espaciais e temporais que foram preenchidas no DMI, como o exemplo simplificado de metadados apresentado na Figura 4.2.

Tabela 4.2 - Metadado das Collections no WLTS.

<b>collection</b>	<b>type</b>	<b>start_date</b>	<b>end_date</b>	<b>resolution</b>
deter_amz_legal	Feature	2016-08-02	2020-06-18	unit: DAY e value: 1
deter_cerrado	Feature	2018-05-01	2020-06-18	unit: DAY e value: 1
prodes_amz_legal	Feature	2007	2019	unit: YEAR e value: 1
prodes_cerrado	Feature	2000	2019	unit: YEAR e value: 1
terraclass_amz	Feature	2004	2014	unit: YEAR e value: 2
mapbiomas_amz_4_1	Image	2007	2018	unit: YEAR e value: 1
mapbiomas_cerrado_4_1	Image	2007	2018	unit: YEAR e value: 1

Para facilitar a utilização do WLTS e WLCSS, dois clientes em Python foram desenvolvidos para os respectivos serviços: `wlts.py` e `lccs.py`. O cliente em Python do serviço WLTS pode ser acessado utilizando a linha de comando. Ao executar o comando `wlts-cli` são apresentadas algumas informações de ajuda para os usuários, como apresenta a Listagem 7. Desta forma é possível listar as *collections* e acessar as trajetórias de UCT por meio de interface de linha de comando informando o endereço (url) do serviço do WLTS que deseja-se utilizar.

---

```
1 Usage: wlts-cli [OPTIONS] COMMAND [ARGS]...
2
3   WLTS on command line.
4
5 Options:
6   --url TEXT  The WLTS server address (an URL).
7   --help      Show this message and exit.
8
9 Commands:
10  collections  Return the list of available collections in the service...
11  trajectory   Return the trajectory associated to the location.
```

---

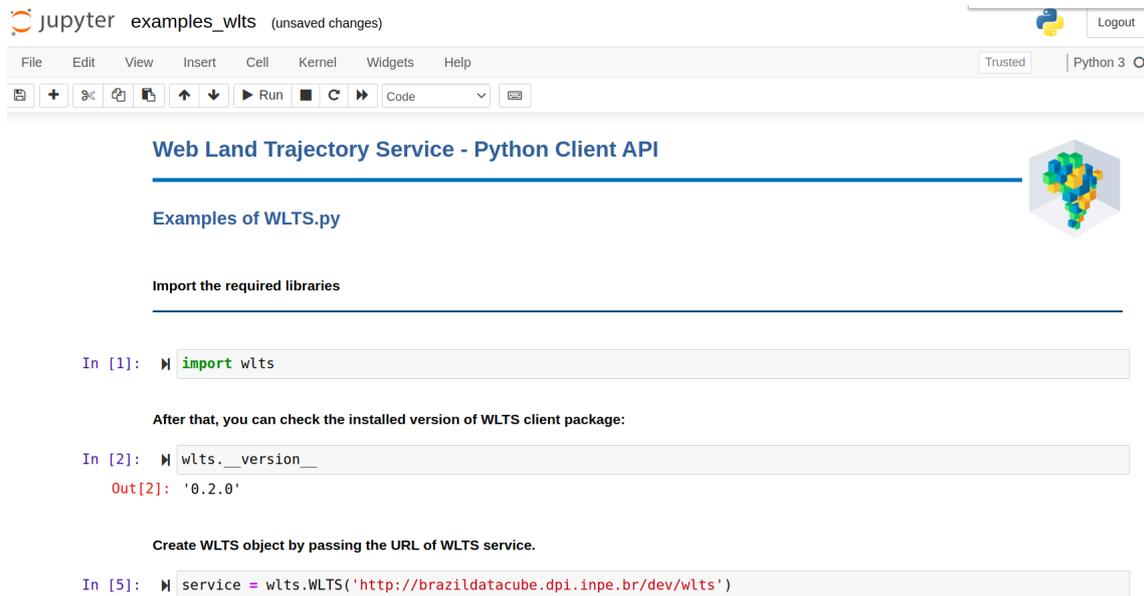
#### Listagem 7 - Linha de comando do wlts-cli do wlts.py.

A Figura 4.2 apresenta, através de um notebook Jupyter, como realizar a importação da biblioteca cliente `wlts.py` e realizar a verificação da versão instalada. Como apresentado na figura, para o `wlts.py` ter disponível as operações do WLTS, deve ser fornecido como parâmetro ao `wlts.WLTS` a URL do WLTS que deseja-se utilizar. Neste exemplo utilizamos o endereço do WLTS<sup>1</sup> disponível pelo BDC.

---

<sup>1</sup><http://brazildatacube.dpi.inpe.br/dev/wlts>

Figura 4.2 - Exemplo de como importar a biblioteca cliente `wlts.py`, através do Jupyter Notebook.



Fonte: Produção da Autora.

O `wlts.py` disponibiliza em alto nível as três operações do serviço WLTS. No exemplo abaixo são apresentadas todas as coleções utilizadas no estudo de caso. Para isto é utilizada a função `list_collection` do biblioteca cliente. A Listagem 8 apresenta a chamada (linha 1) e retorno desta função (linha 3 a 6). Esta operação retorna um dicionário que contém uma lista com os nomes das coleções disponíveis no WLTS. Os nomes das collections são utilizadas como parâmetros para outras operações do WLTS.

```
1 wlts_ob.list_collections()  
2 {  
3   'collections': ['deter_amz_legal', 'deter_cerrado',  
4   'prodes_cerrado', 'terraclass_amz', 'prodes_amz_legal', 'mapbiomas_amz_4_1',  
5   ↵ 'mapbiomas_cerrado_4_1' ]  
6 }
```

Listagem 8 - Listar as collections usando `wlts.py`

As informações espaciais e temporais das coleções, que foram definidas no DMI, podem ser vistas por meio da operação `describe_collection`. Nesta operação é necessário passar o identificador (nome) da `collection` que deseja-se obter as informações, definido pelo parâmetro `collection_id`. A Listagem 9 apresenta os metadados da coleção do `prodes_cerrado` no WLTS. Conforme pode ser observado, os metadados incluem uma descrição e detalhes da `collection`, as informações de extensão espacial e temporal. Também é informada a resolução temporal do dado, composta pelas informações de unidade e valor.

---

```
1 wlts_ob.describe_collection(collection_id="prodes_cerrado")
2 {
3   'collection_type': 'Feature',
4   'description': 'Desmatamento Cerrado',
5   'detail': 'http://www.obt.inpe.br/OBT/assuntos/programas/amazonia/prodes',
6   'name': 'prodes_cerrado',
7   'period': {'end_date': '2019', 'start_date': '2000'},
8   'resolution_unit': {'unit': 'YEAR', 'value': '1'},
9   'spatial_extent': {'xmax': -24.6851196289062,
10  'xmin': -60.0939559936523,
11  'ymax': -2.32670903205872,
12  'ymin': -41.5224685668945}
13 }
```

---

Listagem 9 - Retorno de metadados de uma coleção usando `wlts.py`

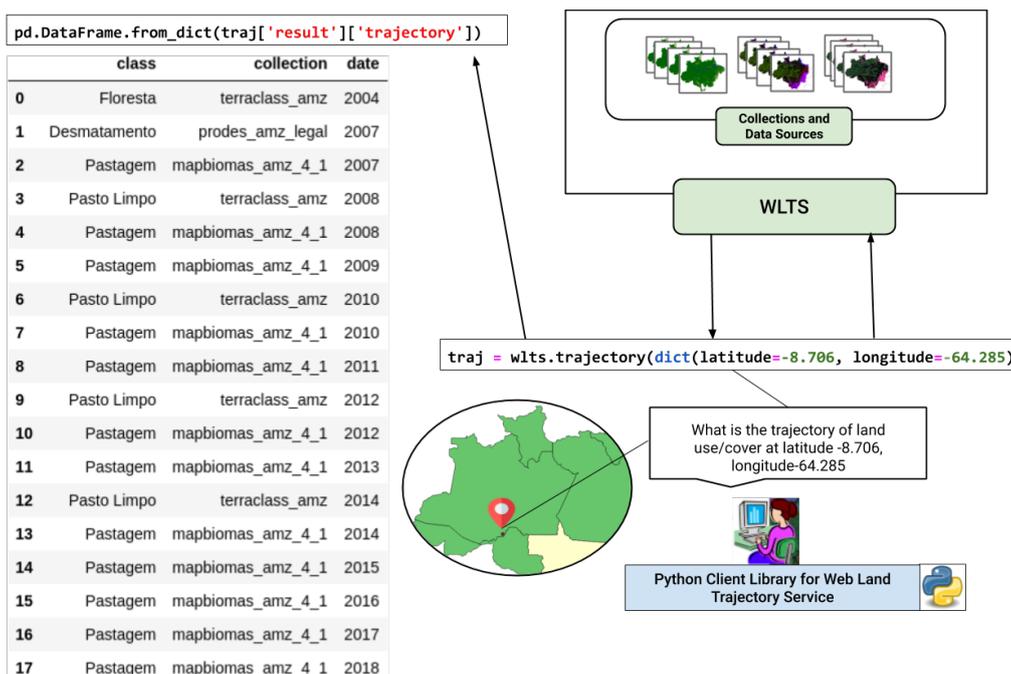
A trajetória de UCT pode ser obtida para um ponto específico indicado pelos parâmetros latitude e longitude em grau decimal adotando-se o Datum WGS84 (EPSG 4326). A Figura 4.3 apresenta um exemplo de trajetória de UCT recuperada para a localização de latitude -8.706 e longitude -64.285.

A localização é passada como parâmetro, em forma de dicionário, para a função `trajectory`. Essa operação retorna um dicionário com duas partes a `query` e o `result`. A `query` retorna o contexto da consulta realizada, ou seja, os parâmetros passados para operação. O `result` contém a chave `trajectory` em que se encontra a trajetória de UCT para a localização informada.

O retorno da função `trajectory` é armazenado na variável `traj`. Neste exemplo

usamos a biblioteca do Python Pandas<sup>2</sup> para apresentar o resultado da trajetória como um *dataframe*, um tipo de estrutura de dados, que contém eixos rotulados: as linhas e colunas.

Figura 4.3 - Trajetória de UCT usando a biblioteca cliente *wlts.py*.



Fonte: Produção da Autora.

Uma consideração importante a ser dada sobre o dado do PRODES, disponível pelo WLTS, é que até 2007 o dado apresenta um acumulado dos desmatamentos dos anos anteriores. E uma vez uma região identificada como Desmatamento, ela entra na máscara do PRODES e não é analisada novamente.

A trajetória de UCT resultante, apresentada na Figura 4.3, apresenta a localidade como Floresta em 2004 no TerraClass. Em 2007 é apresentada como Desmatamento pelo PRODES, mas isto não significa necessariamente que essa localidade foi desmatada nesse ano, devido ao fato de que até 2007 o projeto apresenta o acumulado dos anos anteriores.

<sup>2</sup><https://pandas.pydata.org/>

A partir do momento em que uma região é apresentada como Desmatamento pelo PRODES, esta informação não é mais apresentada por essa coleção. Em 2007, o MapBiomias classificou essa região como Pastagem, e em 2008 o TerraClass classificou como Pasto Limpo. No anos seguintes, as classificações são mantidas em ambos os projetos.

A Figura 4.4 apresenta duas trajetórias retornadas pelo WLTS. Essas trajetórias ilustram que os projetos que constituem a trajetória possuem concordância de classificação para essas localizações. Além de indicar uma área de interesse para ser usada como amostra representativa para ser utilizada como dado de treinamento para geração de novos mapas classificados de UCT, essas trajetórias também apresentam outros potenciais de uso do WLTS: identificar trajetórias com transições de classes impossíveis ou improváveis e utilização do WLTS para auxiliar na análise da consistência dos dados de um mesmo projeto ou entre projetos.

Figura 4.4 - DataFrame com Trajetórias de UCT usando a biblioteca cliente `wlts.py`, através do *Jupyter Notebook*.

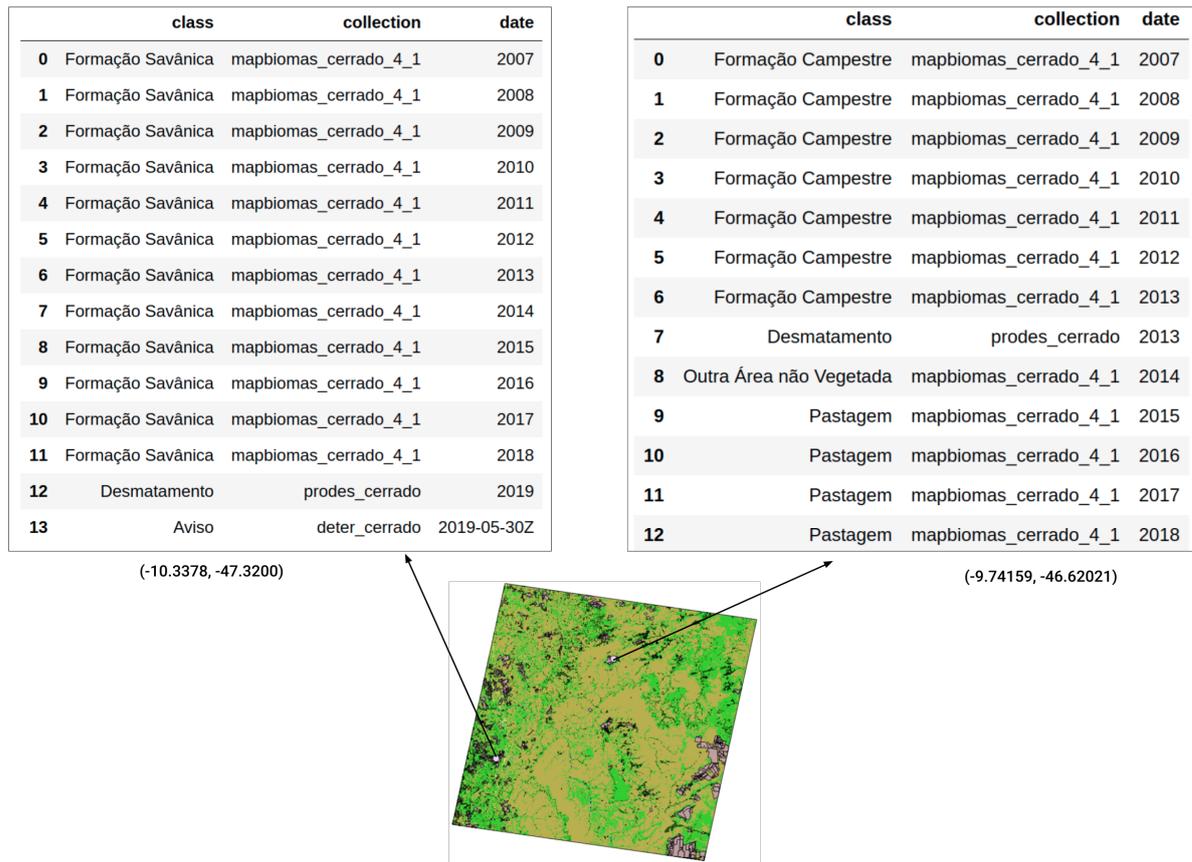
	class	collection	date		class	collection	date
0	Pasto Limpo	terraclass_amz	2004	0	Hidrografia	terraclass_amz	2004
1	Desmatamento	prodes_amz_legal	2007	1	Hidrografia	prodes_amz_legal	2007
2	Pastagem	mapbiomas_amz_4_1	2007	2	Rio, Lago e Oceano	mapbiomas_amz_4_1	2007
3	Pasto Limpo	terraclass_amz	2008	3	Hidrografia	terraclass_amz	2008
4	Pastagem	mapbiomas_amz_4_1	2008	4	Rio, Lago e Oceano	mapbiomas_amz_4_1	2008
5	Pastagem	mapbiomas_amz_4_1	2009	5	Rio, Lago e Oceano	mapbiomas_amz_4_1	2009
6	Pasto Limpo	terraclass_amz	2010	6	Hidrografia	terraclass_amz	2010
7	Pastagem	mapbiomas_amz_4_1	2010	7	Rio, Lago e Oceano	mapbiomas_amz_4_1	2010
8	Pastagem	mapbiomas_amz_4_1	2011	8	Rio, Lago e Oceano	mapbiomas_amz_4_1	2011
9	Pasto Limpo	terraclass_amz	2012	9	Hidrografia	terraclass_amz	2012
10	Pastagem	mapbiomas_amz_4_1	2012	10	Rio, Lago e Oceano	mapbiomas_amz_4_1	2012
11	Pastagem	mapbiomas_amz_4_1	2013	11	Rio, Lago e Oceano	mapbiomas_amz_4_1	2013
12	Pasto Limpo	terraclass_amz	2014	12	Hidrografia	terraclass_amz	2014
13	Pastagem	mapbiomas_amz_4_1	2014	13	Rio, Lago e Oceano	mapbiomas_amz_4_1	2014
14	Pastagem	mapbiomas_amz_4_1	2015	14	Rio, Lago e Oceano	mapbiomas_amz_4_1	2015
15	Pastagem	mapbiomas_amz_4_1	2016	15	Rio, Lago e Oceano	mapbiomas_amz_4_1	2016
16	Cultura Anual e Perene	mapbiomas_amz_4_1	2017	16	Rio, Lago e Oceano	mapbiomas_amz_4_1	2017
17	Cultura Anual e Perene	mapbiomas_amz_4_1	2018	17	Rio, Lago e Oceano	mapbiomas_amz_4_1	2018

A esquerda é apresentada trajetória de UCT para localização latitude -10.710 e longitude -55.612. A direita latitude -4.090 e longitude -63.353.

Fonte: Produção da Autora.

A Figura 4.5 apresenta exemplos de trajetória para a região do cerrado. Para esta consulta, foi fornecido como parâmetros as *collections* `mapbiomas_cerrado_4_1`, `prodes_cerrado` e `deter_cerrado`.

Figura 4.5 - Exemplos de Trajetórias para a região do Cerrado.



A esquerda é apresentada trajetória de UCT para localização latitude -10.3378 e longitude -47.3200. A direita latitude -9.74159 e longitude -46.62021.

Fonte: Produção da Autora.

Para o serviço WLCSS foi desenvolvido o cliente em python denominado `lccs.py`. A inicialização do `lccs.py` segue a mesma estrutura do `wlts.py`, apresentada anteriormente. Após importar a biblioteca, utilizando o comando `import` do python, basta informar a url do serviço que deseja-se utilizar ao objeto `lccs`: `service = lccs.lccs('http://localhost:5000')`. Desta forma podemos utilizar a va-

riável `service` para acessar as funções deste cliente. Para retornar a lista com os nomes dos sistemas de classificação disponíveis no WLCSS, basta executar `service.classification_system`.

A Listagem 10 ilustra como ter acesso ao sistemas de classificação e suas respectivas classes e estilos. A linha 1 apresenta como recuperar os metadados do sistema de classificação do PRODES. Na linha 17 é apresentado como recuperar as classes deste sistema de classificação.

---

```
1 prodes = lccs_service.classification_system(system_id='PRODES')
2 print(prodes)
3
4 {'name': 'PRODES', 'id': 4, 'version': '1.0', 'authority_name': 'INPE',
5  'links': [{ 'rel': 'self',
6              'href': 'http://localhost:5000/classification_systems/PRODES'},
7            { 'rel': 'classes',
8              'href': 'http://localhost:5000/classification_systems/PRODES/classes'},
9            { 'rel': 'parent',
10             'href': 'http://localhost:5000/classification_systems'},
11            { 'rel': 'root', 'href': 'http://localhost:5000/' }]},
12 'description': 'Sistema de Classificação Anual de Desmatamento',
13 'styles': [{ 'rel': 'child',
14              'href': 'http://localhost:5000/classification_systems/PRODES/styles/',
15              'title': 'Styles' }]}
16
17 for classe in prodes.classes():
18     print(classe['name'])
19
20 Floresta
21 Hidrografia
22 Nao Floresta
23 Nuvem
24 Residuo
25 Desmatamento
```

---

Listagem 10 - Acessando os metadados e classes do sistema PRODES

O `lccs.py` possui a função `mapping`, em que dado um sistema de classificação de origem e um de destino (parâmetros `system_id_source` e `system_id_target` respectivamente), essa função retorna o mapeamento entre os sistemas. O retorno desta

função é definido por uma lista onde podemos acessar os mapeamentos correspondente entre as classes.

A lista com o mapeamento entre as classes pode ser utilizada como informação para se buscar a classe de origem e de destino e assim retornar uma trajetórias harmonizada dos projetos, realizando uma integração dos clientes do WLTS e WCLSS. A Figura 4.6 apresenta um exemplo de trajetória harmonizada entre os projetos TerraClass e Prodes. Neste exemplo, foram harmonizadas as legendas do TerraClass para as legendas do PRODES.

Figura 4.6 - Trajetória Harmonizada.



Fonte: Produção da Autora.

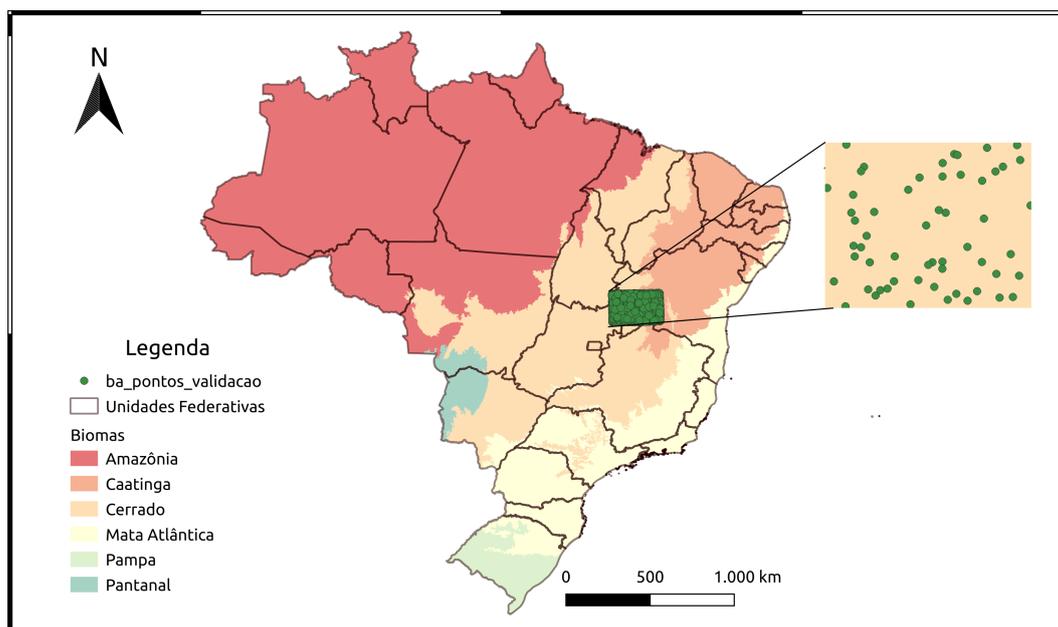
## 4.2 Estudo de caso 2

Uma vertente em que a trajetória de UCT pode ser empregada é na validação de novos mapas classificados de UCT. Neste contexto, este estudo apresenta uma validação dos cubos de dados de UCT gerados pelo projeto *Brazil Data Cube*.

### 4.2.1 Área de Estudo

A área de estudo está localizada na região da Bahia. A Figura 4.7 apresenta as amostras utilizadas no estudo de caso.

Figura 4.7 - Área de estudo com as amostras utilizadas no estudo de caso 2.



Fonte: Produção da Autora.

#### 4.2.2 Dados

Os pontos utilizados neste estudo de caso foram geradas aleatoriamente, sendo 25000 pontos para a região do oeste da Bahia, como apresentado na Figura 4.7. Os pontos foram geradas usando o SIG QGIS com a ferramenta de *random points*, dado a geometria de interesse.

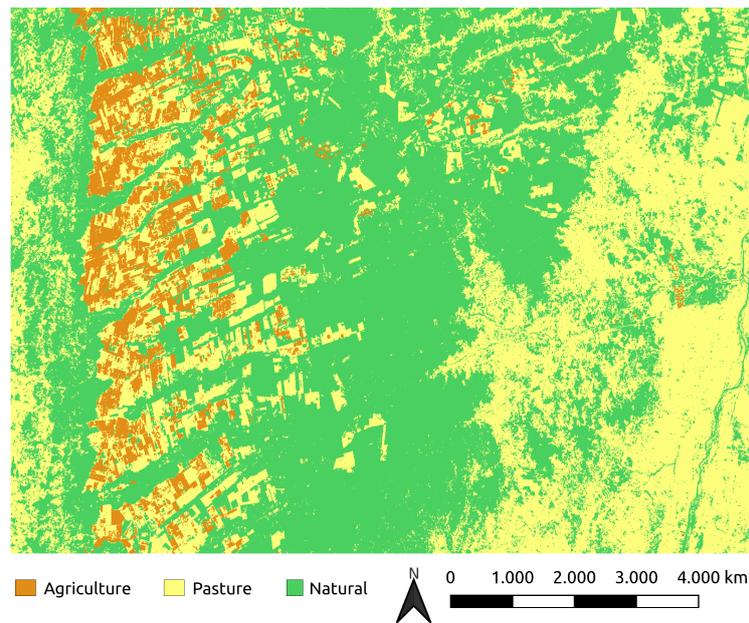
Os dados avaliados neste experimento são os cubos gerados pelo projeto do *Brazil Data Cube* (BDC). Entre os diversos cubos de dados gerados por este projeto, como os cubos do Landsat e Sentinel, o BDC fornece o cubo de dados de imagens do satélite Sino-Brasileiro de Recursos Terrestres (CBERS-4), uma parceira entre Brasil e China.

Para este experimento estamos analisando os produtos do BDC nomeados de C4\_64. Este produto contém os valores de reflectância de superfície do sensor *AWFI* de resolução espacial de 64 metros. Possui composição temporal mensal e de 16 dias. As coleções de cubos de 16 dias C4\_64\_16D\_MED e C4\_64\_16D\_STK e os cubos mensais C4\_64\_1M\_MED e C4\_64\_1M\_STK são geradas aplicando as funções de mediana e *stack*.

O período das imagens é de setembro a agosto dos anos de 2017 a 2018 e 2018 a 2019.

O trabalho de Picoli et al. (2020) apresenta o uso do cubo de dados CBERS-4 para mapear mudanças de UCT. Para realizar a classificação, os pesquisadores coletaram ao todo 922 amostras das classes *Agriculture* (242), *Natural Vegetation* (422) e *Pasture* (258). Os cubos foram classificados utilizando o algoritmo *Random Forest*. A Figura 4.8 apresenta a imagem do classificada do cubo C4\_64\_1M\_MED e sua respectiva legenda.

Figura 4.8 - Classificação *Random forest* do C4\_64\_16D\_MED.



Fonte: Produção da Autora.

Neste estudo de caso, foram comparados as classificações dos cubos CBERS do projeto BDC com os dados provenientes dos programas PRODES e MapBiomas para a região do cerrado. Para realizar o experimento foi utilizado o cliente em Python do WLTS, chamado `wlts.py`. Também foi desenvolvido um *script* em Python que utiliza a biblioteca do *Pandas* para realizar a leitura das localizações espaciais

no formato CSV para consultá-las no WLTS.

### 4.2.3 Resultados

O objetivo deste experimento foi verificar a utilização do WLTS como uma ferramenta analítica proposta dentro do cenário de validação de dados de uso e cobertura da terra. A acurácia das classificações dos cubos de dados CBERS pelo projeto BDC não será abordada.

Uma observação a ser destacada é que para este experimento não consideramos as discrepâncias entre os dados dos cubos C4\_64 e outros projetos com relação às classes de água e áreas não vegetais. Pois para realizar a classificação destes cubos, os autores utilizaram como estratégia de classificação dos cubos BDC em seu artigo, de não considerar água para classificação, devido o tipo de amostras disponíveis para classificação. A estratégia foi aplicar posteriormente uma máscara de água do projeto PRODES.

Além de ajudar na validação os dados, com informações em que pode-se extrair estatísticas de acerto entre os acervos de dados dos projetos, o WLTS fornece um ambiente exploratório que simplifica o acesso e análise sinérgico dos dados. Nas regiões em que ocorreram discordâncias entre o BDC e MapBiomass, pode-se buscar as trajetórias destas localidades, obtendo todo o benefício temporal que a ferramenta provê. Pode-se verificar, por exemplo o caso do ponto de latitude=-12.8292796 e longitude=-45.496841, onde ocorreram algumas discrepâncias entre as classificações dos cubos CBERS.

Figura 4.9 - Brazil Data Cube CBERS Versus MapBiomias.



Fonte: Produção da Autora.

A Figura 4.10 apresenta à esquerda um *dataframe* com a trajetória da localização (latitude=-12.8292796 e longitude=-45.496841) apresentada na Figura anterior. Pode ser visto que ocorreram variações entre os cubos de mediana e *stack*, e entre as classificações dos cubos CBERS e o do projeto Mapbiomas. Um exemplo onde ocorreu uma concordância entre as classificações é apresentado a direita da Figura 4.10. Esta trajetória é pertencente à localização de latitude = -13.012 e longitude -44.488. Para esta localidade, todos os cubos CBERS classificados indicaram *Natural* como classe. Para o projeto Mapbiomas foi indicado como Formação Savânica.

Figura 4.10 - Trajetória Brazil Data Cube CBERS Versus MapBiomias.

	class	collection	date
0	Formação Savânica	mapbiomas_cerrado_4_1	2007
1	Formação Savânica	mapbiomas_cerrado_4_1	2008
2	Formação Savânica	mapbiomas_cerrado_4_1	2009
3	Formação Savânica	mapbiomas_cerrado_4_1	2010
4	Formação Savânica	mapbiomas_cerrado_4_1	2011
5	Formação Savânica	mapbiomas_cerrado_4_1	2012
6	Formação Savânica	mapbiomas_cerrado_4_1	2013
7	Formação Savânica	mapbiomas_cerrado_4_1	2014
8	Desmatamento	prodes_cerrado	2015
9	Formação Savânica	mapbiomas_cerrado_4_1	2015
10	Pastagem	mapbiomas_cerrado_4_1	2016
11	Natural	c4_64_16d_med_bahia	2016-09-01
12	Pasture	c4_64_16d_stk_bahia	2016-09-01
13	Natural	c4_64_1m_med_bahia	2016-09-01
14	Agriculture	c4_64_1m_stk_bahia	2016-09-01
15	Pastagem	mapbiomas_cerrado_4_1	2017
16	Natural	c4_64_16d_med_bahia	2017-09-01
17	Agriculture	c4_64_16d_stk_bahia	2017-09-01
18	Natural	c4_64_1m_med_bahia	2017-09-01
19	Agriculture	c4_64_1m_stk_bahia	2017-09-01
20	Pastagem	mapbiomas_cerrado_4_1	2018
21	Natural	c4_64_16d_med_bahia	2018-09-01
22	Pasture	c4_64_16d_stk_bahia	2018-09-01
23	Pasture	c4_64_1m_med_bahia	2018-09-01
24	Natural	c4_64_1m_stk_bahia	2018-09-01

	class	collection	date
0	Formação Savânica	mapbiomas_cerrado_4_1	2007
1	Formação Savânica	mapbiomas_cerrado_4_1	2008
2	Formação Savânica	mapbiomas_cerrado_4_1	2009
3	Formação Savânica	mapbiomas_cerrado_4_1	2010
4	Formação Savânica	mapbiomas_cerrado_4_1	2011
5	Formação Savânica	mapbiomas_cerrado_4_1	2012
6	Formação Savânica	mapbiomas_cerrado_4_1	2013
7	Formação Savânica	mapbiomas_cerrado_4_1	2014
8	Formação Savânica	mapbiomas_cerrado_4_1	2015
9	Formação Savânica	mapbiomas_cerrado_4_1	2016
10	Natural	c4_64_16d_med_bahia	2016-09-01
11	Natural	c4_64_16d_stk_bahia	2016-09-01
12	Natural	c4_64_1m_med_bahia	2016-09-01
13	Natural	c4_64_1m_stk_bahia	2016-09-01
14	Formação Savânica	mapbiomas_cerrado_4_1	2017
15	Natural	c4_64_16d_med_bahia	2017-09-01
16	Natural	c4_64_16d_stk_bahia	2017-09-01
17	Natural	c4_64_1m_med_bahia	2017-09-01
18	Natural	c4_64_1m_stk_bahia	2017-09-01
19	Formação Savânica	mapbiomas_cerrado_4_1	2018
20	Natural	c4_64_16d_med_bahia	2018-09-01
21	Natural	c4_64_16d_stk_bahia	2018-09-01
22	Natural	c4_64_1m_med_bahia	2018-09-01
23	Natural	c4_64_1m_stk_bahia	2018-09-01

Fonte: Produção da Autora.

## 5 CONCLUSÕES

Neste trabalho foi desenvolvido uma plataforma para acesso e análise de trajetórias de uso e cobertura da terra. Esta plataforma possibilita a integração de diferentes tipos e fontes de dados, fornecendo um ambiente para análise exploratória de trajetórias de UCT ao acervo de dados de diversos projetos. A plataforma fornece uma semântica de alto nível, mais próximo do domínio de aplicação, e se mostra apropriada para especialistas realizarem análises em diversas vertentes.

Este trabalho apresentou os desafios na análise integrada de dados UCT provenientes de diferentes tipos de fontes de dados, e uma proposta para lidar com esses desafios. A proposta apresentada utiliza como abordagem um modelo de dados (DIM) capaz de descrever diferentes *datasets* e *datasources* de UCT. Com base neste modelo, o serviço WLTS é capaz de acessar as trajetórias de UCT de diferentes formatos e fontes de dados. Além disso, a plataforma possui um modelo de dados e um serviço para dar apoio à resolução de questões de interoperabilidade semântica entre os sistemas de classificação dos diversos projetos.

O trabalho demonstrou a utilização da plataforma para a análise exploratória de trajetórias UCT dos projetos PRODES, DETER, TerraClass, MapBiomas e BDC. O modelo de dados proposto mostrou-se capaz de descrever esses diferentes acervos de dados e para mapeá-los nos tipos de dados *Collections* e *DataSource* para realizar o acesso às trajetórias.

Os clientes em Python possibilitaram a interação entre os serviços, além de proporcionar a possibilidade de integração com outras bibliotecas do Python, como o Pandas. A plataforma pode ser integrada com séries temporais de imagens de sensoriamento remoto fornecidas, por exemplo pelo WTSS, o que complementa o ambiente para o suporte às atividades de pesquisa e tomadas de decisão.

Os códigos desenvolvidos neste trabalho, que incluem o modelo de dados, serviços e clientes são livres e abertos. Todos os repositórios de códigos estão disponível no GitHub<sup>1</sup> do projeto BDC. Os repositórios seguem um padrão de organização. Todos incluem documentação de configuração e instalação, exemplos de acesso via linha de comando e notebooks *Jupyter* com exemplos de utilização dos clientes. Os serviços WLTS e WLCSS também possuem repositórios dedicados a especificação do serviço utilizando *OpenAPI 3.0 specification* e sistema de construção de documentação baseado em *ReDoc*.

---

<sup>1</sup><https://github.com/brazil-data-cube>

Por fim, em relação aos trabalhos futuros pode-se levantar algumas melhorias e possibilidades de outros desenvolvimentos. O estudo de caso abordou a utilização da plataforma no viés de análise e de validação de mapas classificados de UCT. Objetivou-se abordar a plataforma no contexto de coleta em grandes escalas de amostras de UCT. Um possível estudo que pode ser desenvolvido é avaliar as implicações de se utilizar um conjunto de dados coletados pelo WLTS influencia nos modelos de *Machine Learning* para classificação de UCT em imagens de sensoriamento remoto. Este estudo pode avaliar as implicações que um conjunto de amostras de áreas inalteradas e validadas por projetos consolidados apresentam em um modelo de classificação automático.

Neste trabalho foram apresentados dois clientes desenvolvidos para os serviços WLTS e WLCSS. Espera-se desenvolver um *plugin* para o QGIS que utilize esses clientes. Este *plugin* poderá conter uma interface de visualização de trajetórias de UCT, que auxilie de maneira visual os pesquisadores e usuários em geral. Em relação a implementação dos serviços e pacotes da plataforma, as próximas versões podem incluir um sistema de *Cache* no componente do WLTS, para melhorar o tempo de respostas às consultas. Ainda em relação à implementação, melhorar o conjunto de testes unitários para os serviços e clientes da plataforma.

Além disso, o trabalho permitiu a identificação de uma nova capacidade de utilização da plataforma para acesso e análise de trajetórias de uso e cobertura da terra. Os resultados de trajetórias retornadas pelo WLTS podem ser utilizadas como um dado de entrada para um outro serviço ou ferramenta. Essa nova capacidade pode realizar o processamento das trajetórias de UCT para identificar padrões de transições entre outros aspectos.

## REFERÊNCIAS BIBLIOGRÁFICAS

- ABDI, A. M. Land cover and land use classification performance of machine learning algorithms in a boreal landscape using sentinel-2 data. **GIScience & Remote Sensing**, v. 57, n. 1, p. 1–20, 2020. Disponível em: <<https://doi.org/10.1080/15481603.2019.1650447>>. 3
- AGUIAR, D. A.; ADAMI, M.; SILVA, W. F.; RUDORFF, B. F. T.; MELLO, M. P.; SILVA, J. d. S. Vila da. In: IEEE INTERNATIONAL GEOSCIENCE AND REMOTE SENSING SYMPOSIUM, 2010. **Proceedings...** Honolulu, HI, USA: IEEE, 2010. p. 2123–2126. 1, 11
- ALMEIDA, C. A. d.; COUTINHO, A. C.; ESQUERDO, J. D. M.; ADAMI, M.; VENTURIERI, A.; DINIZ, C. G.; DESSAY, N.; DURIEUX, L.; GOMES, A. R. High spatial resolution land use and land cover mapping of the Brazilian Legal Amazon in 2008 using Landsat-5/TM and MODIS data. **Acta Amazonica**, v. 46, p. 291 – 302, 09 2016. ISSN 0044-5967. Disponível em: <[http://www.scielo.br/scielo.php?script=sci\\_arttext&pid=S0044-59672016000300291&nrm=iso](http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0044-59672016000300291&nrm=iso)>. Acesso em: 4 maio 2019. 17
- ANDERSON, R.; HARDY, E. E.; ROACH, J. T.; WITMER, R. E. **A land use and land cover classification system for use with remote sensor data**. [S.l.]: USGS, 1976. 34 p. 7, 14, 15
- ARINO, O.; GROSS, D.; RANERA, F.; LEROY, M.; BICHERON, P.; BROCKMAN, C.; DEFOURNY, P.; VANCUTSEM, C.; ACHARD, F.; DURIEUX, L.; BOURG, L.; LATHAM, J.; GREGORIO, A. D.; WITT, R.; HEROLD, M.; SAMBALE, J.; PLUMMER, S.; WEBER, J.-L. Globcover: Esa service for global land cover from meris. In: IEEE INTERNATIONAL GEOSCIENCE AND REMOTE SENSING SYMPOSIUM, 2007. **Proceedings...** Barcelona, Spain: IEEE, 2007. p. 2412–2415. 8
- ARVOR, D.; JONATHAN, M.; MEIRELLES, M. S. P.; DUBREUIL, V.; DURIEUX, L. Classification of modis evi time series for crop mapping in the state of Mato Grosso, Brazil. **International Journal of Remote Sensing**, v. 32, n. 22, p. 7847–7871, 2011. 11
- ARVOR, D.; SIMOES, M.; DUBREUIL, V.; BéGUÉ, A.; SHIMABUKURO, Y. Analyzing the agricultural transition in mato grosso, brazil, using satellite-derived indices. **Applied Geography**, v. 32, p. 702–713, 08 2011. 1

BEST, B. D.; HALPIN, P. N.; FUJIOKA, E.; READ, A. J.; QIAN, S. S.; HAZEN, L. J.; SCHICK, R. S. Geospatial web services within a scientific workflow: predicting marine mammal habitats in a dynamic environment. **Ecological Informatics**, v. 2, n. 3, p. 210 – 223, 2007. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1574954107000477>>. 20

CÂMARA, G.; ASSIS, L. F.; RIBEIRO, G.; FERREIRA, K. R.; LLAPA, E.; VINHAS, L. Big earth observation data analytics: Matching requirements to system architectures. In: ACM SIGSPATIAL INTERNATIONAL WORKSHOP ON ANALYTICS FOR BIG GEOSPATIAL DATA, 5., 2016. **Proceedings...** New York, NY, USA: ACM, 2016. p. 1–6. 1

CÂMARA, G.; QUEIROZ, G.; VINHAS, L.; FERREIRA, K.; SOUZA, R.; SIMOES, R.; LLAPA, E.; ASSIS, L. F.; IPIA, A. The e-sensing architecture for big earth observation data analysis. In: CONFERENCE ON BIG DATA FROM SPACE, 2017. **Proceedings...** Toulouse, France, 2018. 1, 2, 11

CÂMARA, G.; VALERIANO, D.; VIANEI, J.; MOTTA, M.; MAURANO, L. **Metodologia para o cálculo da taxa anual de desmatamento na Amazônia Legal**. 2013. Disponível em:[http://www.obt.inpe.br/OBT/assuntos/programas/amazonia/prodes/pdfs/metodologia\\_taxaprodes.pdf](http://www.obt.inpe.br/OBT/assuntos/programas/amazonia/prodes/pdfs/metodologia_taxaprodes.pdf). Acesso: 29 de mar. 2019. 16

CEOS, C. O. E. O. S. **The Earth observation handbook-satellite Earth observations in support of the sustainable development goals**. [S.l.: s.n.], 2018. 8

CHEN, J.; LI, S.; WU, H.; CHEN, X. Towards a collaborative global land cover information service. **International Journal of Digital Earth**, v. 10, n. 4, p. 356–370, 2017. Disponível em: <<https://doi.org/10.1080/17538947.2016.1267268>>. 3, 4

COMMITTEES, O. **Web coverage service**. 2020. Disponível em: <<https://www.ogc.org/standards/wcs>>. 21

\_\_\_\_\_. **Web feature service**. 2020. Disponível em: <<https://www.ogc.org/standards/wfs>>. 20

\_\_\_\_\_. **Web map service**. 2020. Disponível em: <<https://www.ogc.org/standards/wms>>. 20

DEFRIES, R.; CHAN, J. C.-W. Multiple criteria for evaluating machine learning algorithms for land cover classification from satellite data. **Remote Sensing of Environment**, v. 74, n. 3, p. 503 – 515, 2000. ISSN 0034-4257. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0034425700001425>>. 10

DHU, T.; DUNN, B.; LEWIS, B.; LYMBURNER, L.; MUELLER, N.; TELFER, E.; LEWIS, A.; MCINTYRE, A.; MINCHIN, S.; PHILLIPS, C. Digital earth Australia – unlocking new value from earth observation data. **Big Earth Data**, v. 1, n. 1-2, p. 64–74, 2017. Disponível em: <<https://doi.org/10.1080/20964471.2017.1402490>>. 11

FENG, C.-C.; FLEWELLING, D. Assessment of semantic similarity between land use/land cover classification systems. **Computers, Environment and Urban Systems**, v. 28, n. 3, p. 229 – 246, 2004. ISSN 0198-9715. GIS for Environmental Modeling. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0198971503000206>>. 15, 16

FOLEY, J. A.; DEFRIES, R.; ASNER, G.; BARFORD, C.; BONAN, G.; CARPENTER, S.; III, F. S. C.; COE, M.; DAILY, G.; GIBBS, H.; HELKOWSKI, J. H.; HOLLOWAY, T.; HOWARD, E. A.; KUCHARIK, C.; MONFREDA, C.; PATZ, J.; PRENTICE, I.; RAMANKUTTY, N.; SNYDER, P. K. Global consequences of land use. **Science (New York, N.Y.)**, v. 309, p. 570–4, 08 2005. 1

FOUNDATION, R. E. **The SpatioTemporal Asset Catalog (STAC) specification**. 2020. Disponível em: <https://github.com/radianteearth/stac-spec>. Acesso em: 26 set. 2020. 29

FRIEDL, M.; MCIVER, D.; HODGES, J.; ZHANG, X.; MUCHONEY, D.; STRAHLER, A.; WOODCOCK, C.; GOPAL, S.; SCHNEIDER, A.; COOPER, A.; BACCINI, A.; GAO, F.; SCHAAF, C. Global land cover mapping from modis: algorithms and early results. **Remote Sensing of Environment**, v. 83, n. 1, p. 287 – 302, 2002. ISSN 0034-4257. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0034425702000780>>. 8

GIRI, C. **Remote sensing of land use and land cover: principles and applications**. [S.l.: s.n.], 2016. 1-446 p. 8, 14

GIULIANI, G.; CAMARA, G.; KILLOUGH, B.; MINCHIN, S. Earth observation open science: enhancing reproducible science using data cubes. **Data**, v. 4, n. 4,

2019. ISSN 2306-5729. Disponível em:

<<https://www.mdpi.com/2306-5729/4/4/147>>. 11

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, 2017. Disponível em:

<<https://doi.org/10.1080/20964471.2017.1398903>>. 11

GÓMEZ, C.; WHITE, J. C.; WULDER, M. A. Optical remotely sensed time series data for land cover classification: a review. **ISPRS Journal of Photogrammetry and Remote Sensing**, v. 116, p. 55–72, 2016. 1

GREGORIO, A. D. **Land cover classification system: classification concepts. software version 3**. [s.n.], 2016. Disponível em:

<<http://www.fao.org/3/a-i5232e.pdf>>. 7, 15

HEROLD, M.; MAYAUX, P.; WOODCOCK, C.; BACCINI, A.; SCHMULLIUS, C. Some challenges in global land cover mapping: an assessment of agreement and accuracy in existing 1 km datasets. **Remote Sensing of Environment**, v. 112, p. 2538–2556, 04 2008. 3

HUNAG, F.; ZHANG, H.; WANG, P. Trajectory analysis of forest changes in northern area of changbai mountains, china from landsat tm image. **ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences**, v.39, n.B8, p. 479–484, 2012. 13

IBGE, I. B. de Geografia e E. **Manual técnico do uso da terra**. 1999.

<https://sentinel.esa.int/web/sentinel/missions/sentinel-2>. Acesso em: 10 out 2020. 16

INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS - INPE.

**Monitoramento da Cobertura Florestal da Amazônia por Satélites**. 2008.

Disponível em:<http://www.obt.inpe.br/OBT/assuntos/programas/amazonia/deter/pdfs/relatoriomonitoramento.pdf>. Acesso: 26 abr. 2019. 3

\_\_\_\_\_. **Monitoramento da floresta amazônica brasileira por satélite**. 2018.

Disponível em:

<http://www.obt.inpe.br/OBT/assuntos/programas/amazonia/prodes>. Acesso: 26 mar. 2019. 16

\_\_\_\_\_. **Deter**. 2019. Disponível

em:<http://www.obt.inpe.br/OBT/assuntos/programas/amazonia/deter>.

Acesso: 26 abr. 2019. 17

INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS (INPE). **TerraClass**.

2014. Disponível em:

[http://www.inpe.br/cra/projetos\\_pesquisas/dados\\_terraclass.php](http://www.inpe.br/cra/projetos_pesquisas/dados_terraclass.php). Acesso

em: 26 set. 2020. 18

JANSEN, L.; GREGORIO, A. **Land Cover Classification System (LCCS): classification concepts and user manual**. [S.l.: s.n.], 2000. 7, 14, 15

LEINENKUGEL, P.; DECK, R.; HUTH, J.; OTTINGER, M.; MACK, B. The potential of open geodata for automated large-scale land use and land cover classification. **Remote Sensing**, v. 11, p. 2249, 2019. 3

LIU, H.; ZHOU, Q. Accuracy analysis of remote sensing change detection by rule-based rationality evaluation with post-classification comparison.

**International Journal of Remote Sensing**, v. 25, p. 1037–1050, 03 2004. 13

LIU, J. Y.; ZHUANG, D. F.; LUO, D.; XIAO, X. Land-cover classification of China: integrated analysis of avhrr imagery and geophysical data. **International Journal of Remote Sensing**, v. 24, n. 12, p. 2485–2500, 2003. Disponível em:

<<https://doi.org/10.1080/01431160110115582>>. 8

LU, D.; MAUSEL, P.; BRONDÍZIO, E.; MORAN, E. Change detection techniques.

**International Journal of Remote Sensing**, v. 25, n. 12, p. 2365–2401, 2004. 1

MACIEL, A. M. **Spatiotemporal interval logic for reasoning about land use change dynamics**. 2017. 133 p. Tese (Doutorado em Computação Aplicada)

— Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2017.

Disponível em: <<http://urlib.net/8JMKD3MGP3W34P/3Q44DE8>>. 13

MACIEL, A. M.; CAMARA, G.; VINHAS, L.; PICOLI, M. C. A.; BEGOTTI, R. A.; ASSIS, L. F. F. G. de. A spatiotemporal calculus for reasoning about land-use trajectories.

**International Journal of Geographical Information Science**, v. 33, n. 1, p. 176–192, 2019. Disponível em:

<<https://doi.org/10.1080/13658816.2018.1520235>>. 10, 13

MAPBIOMAS. **MapBiomass general handbook. Algorithm theoretical basis document (ATBD). Collection 4**. Disponível em: "<https://>

[mapbiomas-br-site.s3.amazonaws.com/ATBD\\_Collection\\_4\\_v2\\_Dez2019.pdf](https://mapbiomas-br-site.s3.amazonaws.com/ATBD_Collection_4_v2_Dez2019.pdf)".  
Acesso em: 10 out 2020. 1

MATTILA, T.; HELIN, T.; ANTIKAINEN, R.; SOIMAKALLIO, S.; PINGOUD, K.; WESSMAN, H. **Land use in life cycle assessment**. [S.l.]: Finnish Environment Institute, 2011. (The Finnish Environment 24/2011). ISBN 978-952-11-3926-0. 7

MAYAUX, P.; EVA, H.; GALLEGO, J.; STRAHLER, A. H.; HEROLD, M.; AGRAWAL, S.; NAUMOV, S.; MIRANDA, E. E. D.; BELLA, C. M. D.; ORDOYNE, C.; KOPIN, Y.; ROY, P. S. Validation of the global land cover 2000 map. **IEEE Transactions on Geoscience and Remote Sensing**, v. 44, n. 7, p. 1728–1739, July 2006. ISSN 1558-0644. 8

MENA, C. Trajectories of land-use and land-cover in the northern ecuadorian amazon: temporal composition, spatial configuration, and probability of change. **Photogrammetric Engineering and Remote Sensing**, v. 74, p. 737–751, 06 2008. 4, 12

MERTENS, B.; LAMBIN, E. F. Land-cover-change trajectories in southern Cameroon. **Annals of the Association of American Geographers**, v. 90, p. 467 – 494, 2000. 12

MEYER, W. B.; TURNER, B. L. Human population growth and global land-use/cover change. **Annual Review of Ecology and Systematics**, v. 23, n. 1, p. 39–61, 1992. Disponível em:  
<<https://doi.org/10.1146/annurev.es.23.110192.000351>>. 7

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>>. 25

PICOLI, M. C. A.; CAMARA, G.; SANCHES, I.; SIMÕES, R.; CARVALHO, A.; MACIEL, A.; COUTINHO, A.; ESQUERDO, J.; ANTUNES, J.; BEGOTTI, R. A.; ARVOR, D.; ALMEIDA, C. Big earth observation time series analysis for monitoring brazilian agriculture. **ISPRS Journal of Photogrammetry and Remote Sensing**, v. 145, p. 328 – 339, 2018. ISSN 0924-2716. SI: Latin America Issue. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0924271618302260>>. 10, 11

PICOLI, M. C. A.; SIMOES, R.; CHAVES, M.; SANTOS, L. A.; SANCHEZ, A.; SOARES, A.; SANCHES, I. D.; FERREIRA, K. R.; QUEIROZ, G. R. Cbers data

cube: a powerful technology for mapping and monitoring brazilian biomes. **ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences**, V.3, p. 533–539, 2020. Disponível em:

<<https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/V-3-2020/533/2020/>>. 1, 55

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, 2015. **Anais...** São José dos Campos, INPE, 2015. ISBN 978-85-17-0076-8. Disponível em:

<<http://urlib.net/rep/8JMKD3MGP6W34M/3JM4K7L>>. 21

ROGAN, J.; FRANKLIN, J.; STOW, D.; MILLER, J.; WOODCOCK, C.; ROBERTS, D. Mapping land-cover modifications over large areas: a comparison of machine learning algorithms. **Remote Sensing of Environment**, v. 112, p. 2272–2283, 05 2008. 10

ROGAN, J.; MILLER, J.; STOW, D.; FRANKLIN, J.; LEVIEN, L.; FISCHER, C. Land-cover change monitoring with classification trees using landsat tm and ancillary data. **Photogrammetric Engineering & Remote Sensing**, v. 69, 07 2003. 10

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. Disponível em:

<<https://app.dimensions.ai/details/publication/pub.1092957640>>. 9

SONG, X.-P.; SEXTON, J. O.; HUANG, C.; CHANNAN, S.; TOWNSHEND, J. R. Characterizing the magnitude, timing and duration of urban growth from time series of landsat-based estimates of impervious cover. **Remote Sensing of Environment**, v. 175, p. 1 – 13, 2016. ISSN 0034-4257. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0034425715302480>>. 7

SOUZA, C.; SHIMBO, J. Z.; ROSA, M.; PARENTE, L.; ALENCAR, A.; RUDORFF, B.; HASENACK, H.; MATSUMOTO, M.; FERREIRA, L.; SOUZA-FILHO, P.; OLIVEIRA, S.; ROCHA, W.; FONSECA, A.; BALZANI, C.; DINIZ, C.; COSTA, D.; MONTEIRO, D.; ROSA, E.; VÉLEZ-MARTIN, E.; AZEVEDO, T. Reconstructing three decades of land use and land cover changes in

brazilian biomes with landsat archive and Earth engine. **Remote Sensing**, v. 12, 08 2020. Disponível em:

<[Disponível em: https://www.mdpi.com/2072-4292/12/17/2735](https://www.mdpi.com/2072-4292/12/17/2735)>. Acesso em: Acesso em 4 maio 2019. 18

SPERA, S. A.; COHN, A. S.; VANWEY, L. K.; MUSTARD, J. F.; RUDORFF, B. F.; RISSO, J.; ADAMI, M. Recent cropping frequency, expansion, and abandonment in Mato Grosso, Brazil had selective land characteristics.

**Environmental Research Letters**, v. 9, n. 6, p. 064010, 2014. 1, 11

TUCKER, C. J.; TOWNSHEND, J. R.; GOFF, T. E. African land-cover classification using satellite data. **Science**, v. 227, n. 4685, p. 369–375, 1985. 8

VINHAS, L.; QUEIROZ, G. R.; FERREIRA, K. R.; CÂMARA, G. Web services for big earth observation data. In: . [S.l.: s.n.], 2017. 2, 4, 10, 19, 20, 21

WAGEMANN, J.; CLEMENTS, O.; FIGUERA, R. M.; ROSSI, A. P.; MANTOVANI, S. Geospatial web services pave new ways for server-based on-demand access and processing of big earth data. **International Journal of Digital Earth**, v. 11, n. 1, p. 7–25, 2018. Disponível em:

<<https://doi.org/10.1080/17538947.2017.1351583>>. 20

WANG, D.; GONG, J.; CHEN, L.; ZHANG, L.; SONG, Y.; YUE, Y. Spatio-temporal pattern analysis of land use/cover change trajectories in xihe watershed. **International Journal of Applied Earth Observation and Geoinformation**, v. 14, n. 1, p. 12 – 21, 2012. ISSN 0303-2434. Disponível em:

<<http://www.sciencedirect.com/science/article/pii/S0303243411001152>>. 12, 13

\_\_\_\_\_. Comparative analysis of land use/cover change trajectories and their driving forces in two small watersheds in the western loess plateau of china.

**International Journal of Applied Earth Observation and Geoinformation**, v. 21, p. 241 – 252, 2013. ISSN 0303-2434. Disponível em:

<<http://www.sciencedirect.com/science/article/pii/S0303243412001729>>. 13

WANG, L.; WANG, S.; ZHOU, Y.; LIU, W.; HOU, Y.; ZHU, J.; WANG, F. Mapping population density in china between 1990 and 2010 using remote sensing. **Remote Sensing of Environment**, v. 210, p. 269 – 281, 2018. ISSN 0034-4257.

Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0034425718301007>>. 7

WULDER, M.; MASEK, J.; COHEN, W. B.; LOVELAND, T.; WOODCOCK, C. Opening the archive: how free data has enabled the science and monitoring promise of landsat. **Remote Sensing of Environment**, v. 122, p. 2–10, 07 2012. 9

WULDER, M.; WHITE, J.; LOVELAND, T.; WOODCOCK, C.; BELWARD, A.; COHEN, W.; FOSNIGHT, E.; SHAW, J.; MASEK, J.; ROY, D. The global landsat archive: status, consolidation, and direction. **Remote Sensing of Environment**, v. 185, p. 271–283, 10 2016. 8

WULDER, M. A.; WHITE, J. C.; GOWARD, S. N.; MASEK, J. G.; IRONS, J. R.; HEROLD, M.; COHEN, W. B.; LOVELAND, T. R.; WOODCOCK, C. E. Landsat continuity: issues and opportunities for land cover monitoring. **Remote Sensing of Environment**, v. 112, n. 3, p. 955 – 969, 2008. ISSN 0034-4257.

Disponível em: <[http:](http://www.sciencedirect.com/science/article/pii/S0034425707003331)

[//www.sciencedirect.com/science/article/pii/S0034425707003331](http://www.sciencedirect.com/science/article/pii/S0034425707003331)>. 8

ZHANG, C.; SARGENT, I.; PAN, X.; LI, H.; GARDINER, A.; HARE, J.; ATKINSON, P. M. Joint deep learning for land cover and land use classification. **Remote Sensing of Environment**, v. 221, p. 173–187, February 2019.

Disponível em: <<https://eprints.soton.ac.uk/427873/>>. 7

ZHOU, Q.; LI, B.; KURBAN, A. Trajectory analysis of land cover change in arid environment of China. **International Journal of Remote Sensing**, v. 29, n. 4, p. 1093–1107, 2008. Disponível em:

<<https://doi.org/10.1080/01431160701355256>>. 4

ZOMLOT, Z.; VERBEIREN, B.; HUYSMANS, M.; BATELAAN, O. Trajectory analysis of land use and land cover maps to improve spatial–temporal patterns, and impact assessment on groundwater recharge. **Journal of Hydrology**, v. 554, p. 558 – 569, 2017. ISSN 0022-1694. Disponível em: <[http:](http://www.sciencedirect.com/science/article/pii/S0022169417306364)

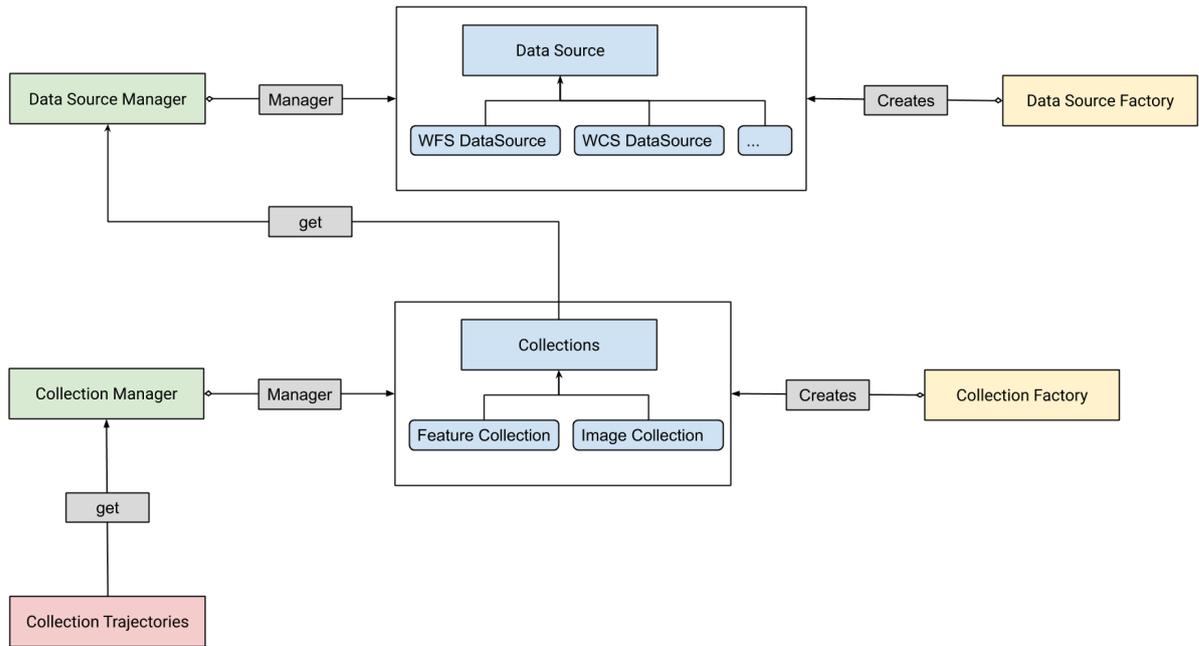
[//www.sciencedirect.com/science/article/pii/S0022169417306364](http://www.sciencedirect.com/science/article/pii/S0022169417306364)>. 13



## ANEXO A - DIAGRAMAS

### A.1 Diagrama camada CDS do WLTS

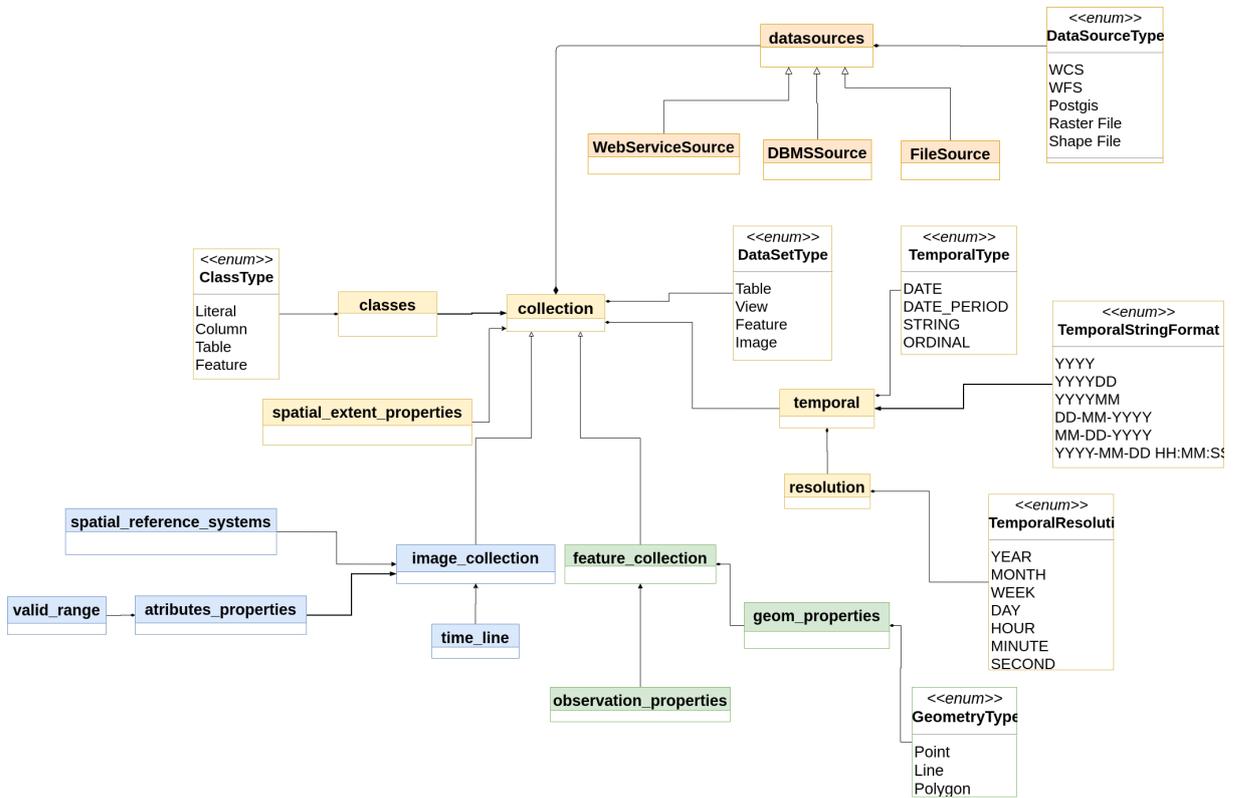
Figura A.1 - Diagrama de Classe simplificado da camada de CDS.



Fonte: Produção da Autora.

## A.2 Diagrama do modelo de dados proposto para o WLTS

Figura A.2 - Diagrama do modelo de dados proposto para o WLTS



Fonte: Produção da Autora.

## ANEXO B - ESPECIFICAÇÃO *OPEN API*

### B.1 WLTS-SPEC

---

```
1 openapi: 3.0.0
2 info:
3
4   title: Web Land Trajectory Service
5
6   description: >
7     Web Land Trajectory Service. You can find out more about WLTS at
8     ↪ [https://github.com/brazil-data-cube/wlts](https://github.com/brazil-data-cube/wlts)
9     There are free and open source implementations based on this
10    ↪ specification:
11
12    - [wlts](https://github.com/brazil-data-cube/wlts): is a Land Use and
13    ↪ Cover Trajectory web service implemented in Python.
14
15    - [wlts.py](https://github.com/brazil-data-cube/wlts.py): is a client
16    ↪ library for Python programmers that makes easy the access to
17    ↪ trajectories from land use and cover maps.
18
19    - [lccs-ws](https://github.com/brazil-data-cube/lccs-ws): is a WLCSS web
20    ↪ server implemented in Python.
21
22    - [lccs.py](https://github.com/brazil-data-cube/lccs.py): is a client
23    ↪ library for Python.
24
25   contact:
26     name: WLTS Specification
27     url: https://gitter.im/brazil-data-cube/community#
28     email: brazildatacube@dpi.inpe.br
29
30   license:
31     name: MIT
32     url: https://opensource.org/licenses/MIT
33
34   version: '0.4.0'
35
36 servers:
37   - url: 'http://brazildatacube.dpi.inpe.br/dev/wlts'
38     description: Development server
```

---

Listagem 11 - Especificação Open API WLTS.

---

```

1 paths:
2   '/list_collections':
3     get:
4       summary: List Collections
5       description: Retrieves the list of available data collections.
6       operationId: listCollections
7       responses:
8         '200':
9           $ref: '#/components/responses/Collections'
10        '500':
11          $ref: '#/components/responses/ServerError'
12  '/describe_collection':
13    get:
14      summary: Describe Collection
15      description: Retrieves the metadata of a given data collection
16      operationId: describeCollection
17      parameters:
18        - $ref: '#/components/parameters/collection_id'
19      responses:
20        '200':
21          $ref: '#/components/responses/CollectionMetadata'
22        '404':
23          $ref: '#/components/responses/NotFound'
24        '500':
25          $ref: '#/components/responses/ServerError'
26  '/trajectory':
27    get:
28      summary: Trajectory
29      description: >
30        Retrieves the land use and cover trajectories from the data
31        ↪ collections given
32        a location in space. The property `result` contains the feature
33        ↪ identifier information, class, time, and the collection
34        ↪ associated to the data item.
35      parameters:
36        - $ref: '#/components/parameters/collections'
37        - $ref: '#/components/parameters/latitude'
38        - $ref: '#/components/parameters/longitude'
39        - $ref: '#/components/parameters/start_date'
40        - $ref: '#/components/parameters/end_date'
41      responses:
42        '200':
43          $ref: '#/components/responses/Trajectory'
44        '404':
45          $ref: '#/components/responses/NotFound'
46        '500':
47          $ref: '#/components/responses/ServerError'

```

---

Listagem 12 - Especificação Open API WLTS Continuação 1.

---

```

1 components:
2   parameters:
3     collection_id:
4       name: collection_id
5       description: >
6         Identifier (name) of a collection.
7       in: query
8       required: true
9       schema:
10        type: string
11     collections:
12       name: collections
13       description: >
14         List of Identifier (name) of collections delimited by comma. i.e
15         ↪ collections=prodes,deter
16       in: query
17       required: false
18       schema:
19        type: string
20     latitude:
21       name: latitude
22       description: >
23         Latitude coordinate .
24       in: query
25       required: true
26       schema:
27        type: number
28     longitude:
29       name: longitude
30       description: >
31         Longitude coordinate.
32       in: query
33       required: true
34       schema:
35        type: number
36     start_date:
37       name: start_date
38       description: >
39         Start date. i.e start_date=2007-01-01.
40       in: query
41       required: false
42       schema:
43        type: string

```

---

Listagem 13 - Especificação Open API WLTS Continuação 2.

---

```

1     end_date:
2         name: end_date
3         description: >
4             End date. i.e end_date=2017-12-31.
5         in: query
6         required: false
7         schema:
8             type: string
9     schemas:
10        CollectionsType:
11            title: Features and Images Collections.
12            type: object
13            properties:
14                collections:
15                    type: array
16                    items:
17                        type: string
18        CollectionMetadataType:
19            title: Collection.
20            description: >
21                Describes a collection.
22            type: object
23            properties:
24                name:
25                    type: string
26                description:
27                    type: string
28                detail:
29                    type: string
30                collection_type:
31                    type: string
32                resolution_unit:
33                    type: object
34                    properties:
35                        unit:
36                            type: string
37                        value:
38                            type: number
39                period:
40                    type: object
41                    properties:
42                        start_date:
43                            type: string
44                        end_date:
45                            type: string
46                spatial_extent:
47                    $ref: '#/components/schemas/SpatialExtentType'

```

---

Listagem 14 - Especificação Open API WLTS Continuação 3.

---

```

1      required:
2          - name
3          - description
4          - collection_type
5          - resolution_unit
6          - period
7
8      TrajectoryType:
9          title: Trajectory.
10         description: >
11             This type represents a trajectory of a giving latitude and
12             ↪ longitude.
13         type: object
14         properties:
15             trajectory:
16                 type: array
17                 items:
18                     type: object
19                     properties:
20                         class:
21                             type: string
22                         collection:
23                             type: string
24                         date:
25                             type: string
26             query:
27                 type: object
28                 properties:
29                     collenctions:
30                         type: array
31                         items:
32                             type: string
33                     longitude:
34                         type: number
35                     latitude:
36                         type: number
37                     start_date:
38                         type: string
39                     end_date:
40                         type: string

```

---

Listagem 15 - Especificação Open API WLTS Continuação 4.

---

```

1      ExceptionType:
2          title: The type of exception raised by the service.
3          description: >
4              Information about the exception An error code plus an optional
5                  ↪ description
6          type: object
7          properties:
8              code:
9                  type: string
10             description:
11                 type: string
12         required:
13             - code
14     SpatialExtentType:
15         type: object
16         properties:
17             xmin:
18                 type: number
19             xmax:
20                 type: number
21             ymin:
22                 type: number
23             ymax:
24                 type: number
25     responses:
26         Collections:
27             description: >
28                 JSON document with the names of collections offered by the
29                     ↪ service.
30             content:
31                 application/json:
32                     schema:
33                         $ref: '#/components/schemas/CollectionsType'
34     CollectionMetadata:
35         description: >
36             JSON document describing the classification collection.
37         content:
38             application/json:
39                 schema:
40                     $ref: '#/components/schemas/CollectionMetadataType'
41     Trajectory:
42         description: >
43             JSON document with trajectory of a givin location.
44         content:
45             application/json:
46                 schema:
47                     $ref: '#/components/schemas/TrajectoryType'

```

---

Listagem 16 - Especificação Open API WLTS Continuação 5.

---

```
1      ServerError:
2          description: >
3              A JSON document informing that an error has occurred in the
4              system.
5          content:
6              application/json:
7                  schema:
8                      $ref: '#/components/schemas/ExceptionType'
9              text/html:
10                 schema:
11                     type: string
12
13     NotFound:
14         description: >
15             The requested URI was not found.
16
17     InvalidParameter:
18         description: >
19             A query parameter has an invalid value.
```

---

Listagem 17 - Especificação Open API WLTS Continuação 6.



## B.2 WLCS-SPEC

---

```
1 openapi: 3.0.0
2
3 info:
4   title: Web Land Classification System Service
5
6   description: >
7     Web Land Classification System Service (WLCSS) is designed to deal with
8     ↪ class system representations in traditional land use and cover maps.
9     ↪ Besides that it can make easy for applications to have access to the
10    ↪ symbology of well know land use and cover maps in applications such
11    ↪ as QGIS or GeoServer.
12
13
14   There are free and open source implementations based on this
15   ↪ specification:
16
17   - [lccs-ws](https://github.com/brazil-data-cube/lccs-ws): is a LCCS web
18   ↪ server implemented in Python.
19
20   - [lccs.py](https://github.com/brazil-data-cube/lccs.py): is a client
21   ↪ library for Python.
22
23   - [wlts](https://github.com/brazil-data-cube/wlts): is a Land Use and
24   ↪ Cover Trajectory web service implemented in Python.
25
26   - [wlts.py](https://github.com/brazil-data-cube/wlts.py): is a client
27   ↪ library for Python programmers that makes easy the access to
28   ↪ trajectories from land use and cover maps.
29
30
31   contact:
32     name: WLCS Specification
33     url: https://gitter.im/brazil-data-cube/community#
34     email: brazildatacube@dpi.inpe.br
35
36   license:
37     name: MIT
38     url: https://opensource.org/licenses/MIT
39
40   version: '0.4.0'
41
42   servers:
43     - url: 'http://www.esensing.dpi.inpe.br/lccs'
44       description: Production server
45
46     - url: 'http://www.dev.esensing.dpi.inpe.br/lccs'
47       description: Development server
```

---

---

```

1 paths:
2   /classification_systems:
3     get:
4       tags:
5         - Classification Systems
6       summary: List of Classification Systems.
7       description: Retrieves the list of names of all available land use
8         ↪ and and cover classification systems in the service.
9       operationId: getClassificationSystems
10      responses:
11        '200':
12          $ref: '#/components/responses/ClassificationSystems'
13        '500':
14          $ref: '#/components/responses/ServerError'
15      post:
16        tags:
17          - Classification Systems
18        summary: Add a new classification system.
19        description: Adds a new classification system and returns a
20          ↪ consolidated document with unique ids and links.
21        operationId: addClassificationSystem
22        requestBody:
23          $ref: '#/components/requestBodies/ClassificationSystem'
24        responses:
25          '201':
26            $ref: '#/components/responses/ClassificationSystem'
27      /classification_systems/{system_id}:
28        parameters:
29          - $ref: '#/components/parameters/system_id'
30        get:
31          tags:
32            - Classification Systems
33          summary: Describes a given classification system.
34          description: Returns a document with full information about the
35            ↪ classification system identified by `system_id`.
36          operationId: describeClassificationSystem
37          responses:
38            '200':
39              $ref: '#/components/responses/ClassificationSystemMetadata'
40            '404':
41              $ref: '#/components/responses/NotFound'
42            '500':
43              $ref: '#/components/responses/ServerError'

```

---

Listagem 19 - Especificação Open API WLCSS Continuação 1.

---

```

1  /classification_systems/{system_id}/classes:
2      parameters:
3          - $ref: '#/components/parameters/system_id'
4      get:
5          tags:
6              - Classes
7          summary: Fetch the classes of a given class system.
8          description: >
9              Fetch classes of the classification system identified by
10             ↪ `system_id`.
11         operationId: getClasses
12         responses:
13             '200':
14                 $ref: '#/components/responses/Classes'
15             '404':
16                 $ref: '#/components/responses/NotFound'
17             '500':
18                 $ref: '#/components/responses/ServerError'
19     post:
20         tags:
21             - Classes
22         summary: >
23             Add a new land use and cover class to classification system.
24         operationId: addClasses
25         requestBody:
26             $ref: '#/components/requestBodies/Class'
27         responses:
28             '201':
29                 $ref: '#/components/responses/Class'
30
31 /classification_systems/{system_id}/classes/{class_id}:
32     get:
33         tags:
34             - Classes
35         summary: fetch a single class.
36         description: >
37             Fetch the classe with id `class_id` in the classification
38             system with id `system_id`.
39         operationId: getClass
40         parameters:
41             - $ref: '#/components/parameters/system_id'
42             - $ref: '#/components/parameters/class_id'
43         responses:
44             '200':
45                 $ref: '#/components/responses/ClassMetadata'
46             '404':
47                 $ref: '#/components/responses/NotFound'
48             '500':
49                 $ref: '#/components/responses/ServerError'

```

---

---

```

1  /mappings/{system_id_source}:
2      get:
3          tags:
4              - Mappings
5          summary: Retrives available classification systems mappings.
6          operationId: getAvailablesMappings
7          parameters:
8              - $ref: '#/components/parameters/system_id_source'
9          responses:
10             '200':
11                 $ref: '#/components/responses/AvailableMappings'
12             '404':
13                 $ref: '#/components/responses/NotFound'
14             '500':
15                 $ref: '#/components/responses/ServerError'
16 /mappings/{system_id_source}/{system_id_target}:
17     get:
18         tags:
19             - Mappings
20         summary: Retrives classification systems mappings.
21         description: Mapping of classification systems.
22         operationId: getMappings
23         parameters:
24             - $ref: '#/components/parameters/system_id_source'
25             - $ref: '#/components/parameters/system_id_target'
26         responses:
27             '200':
28                 $ref: '#/components/responses/Mappings'
29             '404':
30                 $ref: '#/components/responses/NotFound'
31             '500':
32                 $ref: '#/components/responses/ServerError'
33 /classification_systems/{system_id}/styles:
34     get:
35         tags:
36             - Styles
37         summary: fetch styles
38         description: >
39             Retrives styles of the classification with id `system_id`.
40         operationId: getStyles
41         parameters:
42             - $ref: '#/components/parameters/system_id'
43         responses:
44             '200':
45                 $ref: '#/components/responses/Styles'
46             '404':
47                 $ref: '#/components/responses/NotFound'
48             '500':
49                 $ref: '#/components/responses/ServerError'

```

---

---

```

1 /classification_systems/{system_id}/styles/{styles_format_id}:
2   get:
3     tags:
4       - Styles
5     summary: fetch styles
6     description: >
7       Retrives style of the classification with id `system_id`.
8     operationId: getStylesFormat
9     parameters:
10      - $ref: '#/components/parameters/system_id'
11    responses:
12      '200':
13        $ref: '#/components/responses/StylesFormat'
14      '404':
15        $ref: '#/components/responses/NotFound'
16      '500':
17        $ref: '#/components/responses/ServerError'
18  components:
19    schemas:
20      ClassificationSystemsType:
21        title: A document with the list of land use and and cover
22        ↪ classification systems.
23        description: >
24          The name of a classification system is unique and it can be used
25          ↪ in successive queries in the system.
26        type: object
27        properties:
28          links:
29            type: array
30            items:
31              $ref: "#/components/schemas/LinkType"
32        required:
33          - links
34      ClassificationSystemType:
35        title: Land Use and Cover Classification System
36        description: >
37          This type represents a given land use and cover classification
38          ↪ system.
39        type: object
40        properties:
41          name:
42            type: string
43          description:
44            type: string
45          authority_name:
46            type: string
47          version:
48            type: string
49        required:
50          - name
51          - description
52          - authority_name
53          - version

```

---

---

```

1      ClassificationSystemMetadataType:
2          type: object
3          properties:
4              id:
5                  type: number
6              name:
7                  type: string
8              authority_name:
9                  type: string
10             description:
11                 type: string
12             version:
13                 type: string
14             styles:
15                 type: array
16                 items:
17                     $ref: "#/components/schemas/LinkType"
18             example:
19                 - href:
20                     ↪ http://www.esensing.dpi.inpe.br/lccs/system_id/styles/style_format
21                     rel: item
22                     title: Styles
23             links:
24                 type: array
25                 items:
26                     $ref: "#/components/schemas/LinkType"
27             example:
28                 - href:
29                     ↪ http://www.esensing.dpi.inpe.br/lccs/classification_systems/system_id/
30                     rel: self
31                 - href:
32                     ↪ http://www.esensing.dpi.inpe.br/lccs/classification_systems/system_id/
33                     rel: classes
34                 - href:
35                     ↪ http://www.esensing.dpi.inpe.br/lccs/classification_systems
36                     rel: parent
37                 - href: http://www.esensing.dpi.inpe.br/lccs/
38                     rel: root
39
40         required:
41             - id
42             - name
43             - links

```

---

Listagem 23 - Especificação Open API WLCSS Continuação 5.

---

```

1      ClassesType:
2          title: Land Use and Cover Classes.
3          description: >
4              Land Use and Cover Classes.
5          type: object
6          properties:
7              links:
8                  type: array
9                  items:
10                     $ref: "#/components/schemas/LinkType"
11          required:
12              - links
13      ClassType:
14          title: Land Use and Cover Class
15          description: Land Use and Cover Class.
16          type: object
17          properties:
18              name:
19                  type: string
20              code:
21                  type: string
22              description:
23                  type: string
24              parent_name:
25                  type: string
26          required:
27              - name
28              - code
29              - description
30      ClassMetadataType:
31          title: Land Use and Cover Class.
32          description: >
33              Describes a class from a classification system.
34          type: object
35          properties:
36              id:
37                  type: number
38              name:
39                  type: string
40              code:
41                  type: string
42              description:
43                  type: string
44              parent:
45                  $ref: "#/components/schemas/ClassMetadataType"
46              links:
47                  type: array
48                  items:
49                      $ref: "#/components/schemas/LinkType"
50          required:
51              - id
52              - name
53              - code

```

---

---

```
1 StylesType:
2   type: object
3   properties:
4     links:
5       type: array
6       items:
7         $ref: "#/components/schemas/LinkType"
8   required:
9     - links
10 StyleFileType:
11   type: string
12   format: base64
13 MappingType:
14   type: object
15   properties:
16     links:
17       type: array
18       items:
19         $ref: "#/components/schemas/LinkType"
20     description:
21       type: string
22     degree_of_similarity:
23       type: number
24 MappingsType:
25   type: array
26   items:
27     $ref: "#/components/schemas/MappingType"
```

---

Listagem 25 - Especificação Open API WLCSS Continuação 7.

---

```

1  LinkType:
2      title: Link for related information.
3      description: This is a generic way to link information in the
4      ↪ service.
5      type: object
6      properties:
7          href:
8              type: string
9              format: url
10             example:
11                 ↪ 'http://www.esensing.dpi.inpe.br/lccs/classification\_systems/system\_id/c
12             rel:
13                 type: string
14                 example: child
15             title:
16                 type: string
17                 example: Child
18         required:
19             - href
20             - rel
21
22 ExceptionType:
23     title: The type of exception raised by the service.
24     description: >
25         Information about the exception: an error code plus an optional
26         ↪ description.
27     type: object
28     properties:
29         code:
30             type: string
31         description:
32             type: string
33     required:
34         - code
35
36 responses:
37     ClassificationSystems:
38         description: >
39             JSON document with the name list of land use and and cover
40             ↪ classification systems offered by the service.
41         content:
42             application/json:
43                 schema:
44                     $ref: '#/components/schemas/ClassificationSystemsType'

```

---

Listagem 26 - Especificação Open API WLCSS Continuação 8.

---

```

1      ClassificationSystem:
2          description: >
3              A JSON document with the classification system information.
4          content:
5              application/json:
6                  schema:
7                      $ref: '#/components/schemas/ClassificationSystemType'
8      ClassificationSystemMetadata:
9          description: >
10             JSON document describing the classification system.
11         content:
12             application/json:
13                 schema:
14                     $ref:
15                         ↪ '#/components/schemas/ClassificationSystemMetadataType'
16     Classes:
17         description: >
18             A JSON document consisting of a list of land use and cover
19             ↪ classes.
20         content:
21             application/json:
22                 schema:
23                     $ref: '#/components/schemas/ClassesType'
24     Class:
25         description: >
26             A JSON document consisting of a land use and cover class.
27         content:
28             application/json:
29                 schema:
30                     $ref: '#/components/schemas/ClassType'
31     ClassMetadata:
32         description: >
33             fetch the classe with id `class_id` in the classification system
34             with id `system_id`
35         content:
36             application/json:
37                 schema:
38                     $ref: '#/components/schemas/ClassMetadataType'
39     Styles:
40         description: >
41             The response is a document consisting of all styles in the
42             ↪ classification system.
43         content:
44             application/json:
45                 schema:
46                     $ref: '#/components/schemas/StylesType'

```

---

Listagem 27 - Especificação Open API WLCSS Continuação 9.

---

```

1 StylesFormat:
2   description: >
3     The response the style file of the classification system.
4   content:
5     application/json:
6       schema:
7         $ref: '#/components/schemas/StyleFileType'
8 AvailableMappings:
9   description: >
10    fetch all mappings availables with `system_id_source`.
11  content:
12    application/json:
13      schema:
14        $ref: "#/components/schemas/ClassificationSystemsType"
15 Mappings:
16   description: >
17     fetch the mapping with `system_id_source` and `system_id_target`.
18  content:
19    application/json:
20      schema:
21        $ref: '#/components/schemas/MappingsType'
22 ServerError:
23   description: >
24     A JSON document informing that an error has occurred in the
25     ↪ system.
26  content:
27    application/json:
28      schema:
29        $ref: '#/components/schemas/ExceptionType'
30    text/html:
31      schema:
32        type: string
33 NotFound:
34   description: >
35     The requested URI was not found.
36 InvalidParameter:
37   description: >
38     A query parameter has an invalid value.

```

---

Listagem 28 - Especificação Open API WLCSS Continuação 10.

---

```

1 parameters:
2     system_id:
3         name: system_id
4         in: path
5         description: Identifier (name) of a classification system.
6         required: true
7         schema:
8             type: string
9     class_id:
10        name: class_id
11        in: path
12        description: Identifier (name) of a class.
13        required: true
14        schema:
15            type: string
16    system_id_source:
17        name: system_id_source
18        in: path
19        description: Identifier (name) of a source classification system.
20        required: true
21        schema:
22            type: string
23    system_id_target:
24        name: system_id_target
25        in: path
26        description: Identifier (name) of a target classification system.
27        required: true
28        schema:
29            type: string
30    requestBodies:
31        ClassificationSystem:
32            description: >
33                A document describing the new classification system.
34            required: true
35            content:
36                application/json:
37                    schema:
38                        $ref: '#/components/schemas/ClassificationSystemType'
39        Class:
40            description: >
41                A land use and cover class to be added to a class system.
42            required: true
43            content:
44                application/json:
45                    schema:
46                        $ref: '#/components/schemas/ClassType'

```

---

Listagem 29 - Especificação Open API WLCSS Continuação 11.

# ANEXO C - REPOSITÓRIOS DE DADOS

## C.1 LCS-DB

Figura C.1 - Página do repositório do LCS-DB no Github.

The screenshot shows the GitHub repository page for 'lcss-db' under the organization 'brazil-data-cube'. The repository has 3 watchers, 0 stars, and 2 forks. It is currently on the 'master' branch with 2 other branches and 1 tag. The repository contains 137 commits and has a recent merge pull request #64 from 'fabianaziotti/release04'. The file list includes folders like 'docs', 'examples', and 'lcss\_db', and files like '.gitignore', '.readthedocs.yml', '.travis.yml', 'CHANGES.rst', 'LICENSE', 'MANIFEST.in', 'README.rst', 'pytest.ini', 'run-tests.sh', 'setup.cfg', and 'setup.py'. The 'About' section describes it as a 'Land Cover Classification System Database Model' with tags for 'land-cover', 'land-use', 'database', 'postgresql', 'gis', and 'geospatial'. It includes a 'Readme' and 'MIT License'. The 'Releases' section shows 'Version 0.2.0' as the latest release on April 11. The 'Packages' section indicates no packages are published. The 'Contributors' section lists 'fabianaziotti' and 'gqueiroz'. The 'Languages' section shows Python at 95.3%, Shell at 2.6%, and Mako at 2.1%. The 'README.rst' section features a title 'Land Cover Classification System Database Model', a badge for 'license MIT', and an 'About' section starting with 'Currently, there are several data sets on regional, national and global scales with information on land use and land cover that aim to support a large number of applications, including the management of natural resources, climate change and the forest and biodiversity conservation. These data sets are organized in a...'.

Fonte: Autora

## C.2 WLCSS

Figura C.2 - Página do repositório do WLCSS no Github.

The screenshot shows the GitHub repository page for 'brazil-data-cube/lccs-ws'. The repository is private and has 4 unwatched items, 0 stars, and 1 fork. The main navigation includes Code, Issues (14), Pull requests, Actions, Projects, Security, and Insights. The repository is currently on the 'master' branch with 2 other branches and 1 tag. A recent pull request #32 by 'gqueiroz' is highlighted, merging from 'fabianaziotti/refactoring'. The file list shows various directories and files, including 'doc', 'docker', 'lccs\_ws', 'tests', and configuration files like '.dockerignore', '.gitignore', and 'requirements.txt'. The right sidebar contains 'About' (Land Cover Classification System Web Service), 'Releases' (Version 0.2.0-0), 'Packages' (No packages published), 'Contributors' (fabianaziotti, gqueiroz), and 'Languages' (Python 95.8%, Shell 3.3%, Dockerfile 0.9%). The README.rst file is partially visible at the bottom, showing the title 'Land Cover Classification System Web Service' and an 'About' section.

File	Commit Message	Time Ago
doc	Preparing release of version 0.2.0-0 (ref #2)	6 months ago
docker	adding docker-compose close #17	6 months ago
lccs_ws	fixing rel names	17 days ago
tests	adding test	8 months ago
.dockerignore	Adding Dockerfile	6 months ago
.gitignore	adding lccs skeleton	10 months ago
.readthedocs.yml	Improve documentation in data.py and changing 2019 to 2020 in files	17 days ago
.travis.yml	Preparing release of version 0.2.0-0 (ref #2)	6 months ago
CHANGES.rst	Improve documentation in data.py and changing 2019 to 2020 in files	17 days ago
DEPLOY.rst	reviewing deploy and install files	17 days ago
INSTALL.rst	reviewing deploy and install files	17 days ago
LICENSE	Improve documentation in data.py and changing 2019 to 2020 in files	17 days ago
MANIFEST.in	Improve documentation in data.py and changing 2019 to 2020 in files	17 days ago
README.rst	Improve documentation in data.py and changing 2019 to 2020 in files	17 days ago
build.sh	Improve documentation in data.py and changing 2019 to 2020 in files	17 days ago
docker-compose.yml	adding docker-compose close #17	6 months ago
pytest.ini	Improve documentation in data.py and changing 2019 to 2020 in files	17 days ago
requirements.txt	Adding Dockerfile	6 months ago
run-tests.sh	Improve documentation in data.py and changing 2019 to 2020 in files	17 days ago
setup.cfg	add sphinx files ref #7	6 months ago
setup.py	Improve documentation in data.py and changing 2019 to 2020 in files	17 days ago

**README.rst**

### Land Cover Classification System Web Service

license MIT build passing coverage unknown docs unknown lifecycle experimental tag repo not found chat on gitter

#### About

Currently, there are several data sets on regional, national and global scales with information on land use and land cover that aim to support a large number of applications, including the management of natural resources

Fonte: Autora

Figura C.3 - Página do repositório do Cliente em Python para WLCSS no Github.

Repository: [brazil-data-cube / lccs.py](#) | Unwatch 4 | Star 1 | Fork 2

Code | Issues 10 | Pull requests 1 | Actions | Projects | Wiki | Security | Insights

master | 1 branch | 0 tags | Go to file | Add file | Code

**gqueiroz** Merge pull request #23 from fabianaziotti/refactory | ebd7be3 4 days ago | 34 commits

File/Folder	Commit Message	Time
doc/sphinx	Adding sphinx documentation close #12 #13	4 days ago
examples	Adding sphinx documentation close #12 #13	4 days ago
lccs	fixing tests	4 days ago
tests	fixing tests	4 days ago
.gitignore	Adding .gitignore	10 months ago
.readthedocs.yml	adding discord icon README and review headers	4 days ago
.travis.yml	review comments and minor changes	5 months ago
CHANGES.rst	adding discord icon README and review headers	4 days ago
CLI.rst	adding discord icon README and review headers	4 days ago
INSTALL.rst	adding discord icon README and review headers	4 days ago
LICENSE	adding discord icon README and review headers	4 days ago
MANIFEST.in	adding discord icon README and review headers	4 days ago
README.rst	adding discord icon README and review headers	4 days ago
RUNNING.rst	adding discord icon README and review headers	4 days ago
pytest.ini	adding discord icon README and review headers	4 days ago
run-tests.sh	adding discord icon README and review headers	4 days ago
setup.cfg	adding discord icon README and review headers	4 days ago
setup.py	adding discord icon README and review headers	4 days ago

**About**  
Python Client Library for Land Cover Classification System Web Service  
Readme  
MIT License

**Releases**  
No releases published

**Packages**  
No packages published

**Contributors** 2  
fabianaziotti Fabiana Ziotti  
gqueiroz Gilberto Ribeiro de Quei...

**Languages**  
Python 98.1% | Shell 1.9%

**README.rst**

## Python Client Library for Land Cover Classification System Web Service

license MIT | build unknown | coverage 0% | docs passing | lifecycle experimental | chat 21 online

### Installation

See [INSTALL.rst](#).

### Using LCCS in the Command Line

See [CLI.rst](#).

Fonte: Autora

## C.3 WLTS

Figura C.4 - Página do repositório do WLTS no Github.

The screenshot displays the GitHub repository page for `brazil-data-cube/wlts`. The repository is private and has 0 stars and 0 forks. The main content area shows a list of files and folders with their commit history:

File/Folder	Commit Message	Time Ago
<code>doc/sphinx</code>	Review for release 0.2.0	6 months ago
<code>docker</code>	removing comment line	5 months ago
<code>tests</code>	preparing for test	6 months ago
<code>wlts</code>	remove duplicate code and minor changes	last month
<code>.dockerignore</code>	adding docker support close #12	5 months ago
<code>.gitignore</code>	adding GIT .gitignore file	11 months ago
<code>.readthedocs.yml</code>	changing to flask-cli and adding doc	6 months ago
<code>.travis.yml</code>	Update .travis.yml	5 months ago
<code>CHANGES.rst</code>	adding template for CHANGES.rst	11 months ago
<code>DEPLOY.rst</code>	adding docker support close #12	5 months ago
<code>INSTALL.rst</code>	Adding a real example	3 months ago
<code>LICENSE</code>	Review for release 0.2.0	6 months ago
<code>MANIFEST.in</code>	adding docker support close #12	5 months ago
<code>README.rst</code>	Review for release 0.2.0	6 months ago
<code>docker-compose.yml</code>	adding docker support close #12	5 months ago
<code>pytest.ini</code>	improving test suite	11 months ago
<code>requirements.txt</code>	adding Travis CI	11 months ago
<code>run-tests.sh</code>	Review for release 0.2.0	6 months ago
<code>setup.cfg</code>	Review for release 0.2.0	6 months ago
<code>setup.py</code>	split json config file	3 months ago

The README section is titled "Web Land Trajectory Service" and includes a status bar with the following information: license MIT, build passing, coverage 51%, docs unknown, lifecycle experimental, Release, chat on gitter. The "About" section states: "Land Use and Cover information is essential to support governments in decision making on the impact of human activities on the environment, for planning the use of natural resources, conservation of biodiversity, and monitoring climate change."

The sidebar on the right contains the following sections:

- About:** Web Land Trajectory Service, Readme, MIT License.
- Releases:** No releases published.
- Packages:** No packages published.
- Contributors:** 2 contributors: fabianaziotti (Fabiana Ziotti) and gqueiroz (Gilberto Ribeiro de Que).
- Languages:** Python 98.1%, Other 1.9%.

Fonte: Autora

Figura C.5 - Página do repositório do Cliente em Python para WLTS no Github.

The screenshot displays the GitHub repository page for 'brazil-data-cube/wlts.py'. At the top, there are navigation tabs for Code, Issues (9), Pull requests (1), Actions, Projects, Wiki, Security, and Insights. Below the repository name, there are buttons for 'Go to file', 'Add file', and 'Code'. A recent commit by 'gqueiroz' is highlighted, showing a merge pull request #26 from 'fabianazioti/review\_wlts' with 65 commits. A list of files and folders follows, including 'doc/sphinx', 'examples', 'tests', 'wlts', and various configuration files like '.gitignore', '.readthedocs.yml', and 'requirements.txt'. The sidebar on the right contains sections for 'About' (Python Client Library for Web Land Trajectory Service), 'Releases' (No releases published), 'Packages' (No packages published), 'Contributors' (gqueiroz and fabianazioti), and 'Languages' (Python 97.5%, Shell 2.5%). The main content area shows the 'README.rst' file, which includes the title 'Python Client Library for Web Land Trajectory Service', a progress bar for repository status (license MIT, build passing, coverage 82%, docs passing, lifecycle experimental, chat 21 online), and an 'About' section describing the library's purpose.

Fonte: Autora

