



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

 etsinf

Escola Tècnica
Superior d'Enginyeria
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

Intelligent Radio Spectrum Monitoring

Trabajo Fin de Máster

Máster Universitario en Gestión de la Información

Autor: Fábio Santos Lobão

Tutor: Cèsar Ferri Ramírez

2018-2019

Acknowledges

Many helped in this endeavour, but some deserve an honourable mention. I tried to remember some.

I shall thank many colleagues at [Anatel](#), including Agostinho Linhares, Alexandre Lobão, Andrea Costa, Arthur Pisaruk, Carlos Lourenzato, David de Oliveira, David Sidney, Gianluca Fiorentini, Marcelo Loschi, Renato Sales, Teógenes Nobrega, Walkiria Pereira and Yroá Robledo.

All professors and colleagues at UPV, especially from the MUGI class of 2018/2019 and from DSIC the fellow researchers Carlos Aranda, Cèsar Ferri, David Nieves, Fernando Martínez, José Hernández-Orallo, Lidia Contreras, M^a José Ramírez and Pablo Arnalte.

Representatives from different manufactures including Narda, GEW, Rohde&Schwarz, TCI, ATDI and LS Telecom and especially from [CRFS](#), most thanks to Alistair Massarella, Filip Zganec and Jon Bradley.

Colleagues at ITU and regulatory agencies BNetzA, Ofcom, TRA and FCC

Family and friends at València and Brasília

Abstract

Spectrum monitoring is an important part of the radio spectrum management process, providing feedback on the workflow that allows for our current wirelessly interconnected lifestyle. The constantly increasing number of users and uses of wireless technologies is pushing the limits and capabilities of the existing infrastructure, demanding new alternatives to manage and analyse the extremely large volume of data produced by existing spectrum monitoring networks. This study addresses this problem by proposing an information management system architecture able to increase the analytical level of a spectrum monitoring measurement network. This proposal includes an alternative to manage the data produced by such network, methods to analyse the spectrum data and to automate the data gathering process. The study was conducted employing system requirements from the Brazilian National Telecommunications Agency and related functional concepts were aggregated from the reviewed scientific literature and publications from the International Telecommunication Union. The proposed solution employs microservice architecture to manage the data, including tasks such as format conversion, analysis, optimization and automation. To enable efficient data exchange between services, we proposed the use of a hierarchical structure created using the HDF5 format. The suggested architecture was partially implemented as a pilot project, which allowed to demonstrate the viability of presented ideas and perform an initial refinement of the proposed data format and analytical algorithms. The results pointed to the potential of the solution to solve some of the limitations of the existing spectrum monitoring workflow. The proposed system may play a crucial role in the integration of the spectrum monitoring activities into open data initiatives, promoting transparency and data reusability for this important public service.

Keywords: radio spectrum; spectrum monitoring; spectrum management; hdf5; ontology; bayesian detection; clustering

Resumen

El control y análisis de uso del espectro electromagnético, un servicio conocido como comprobación técnica del espectro, es una parte importante del proceso de gestión del espectro de radiofrecuencias, ya que proporciona la información necesaria al flujo de trabajo que permite nuestro estilo de vida actual, interconectado e inalámbrico. El número cada vez más grande de usuarios y el creciente uso de las tecnologías inalámbricas amplían las demandas sobre la infraestructura existente, exigiendo nuevas alternativas para administrar y analizar el gran volumen de datos producidos por las estaciones de medición del espectro. Este estudio aborda este problema al proponer una arquitectura de sistema para la gestión de información capaz de aumentar la capacidad de análisis de una red de equipos de medición dedicados a la comprobación técnica del espectro. Esta propuesta incluye una alternativa para administrar los datos producidos por dicha red, métodos para analizar los datos recolectados, así como una propuesta para automatizar el proceso de recopilación. El estudio se realizó teniendo como referencia los requisitos de la Agencia Nacional de Telecomunicaciones de Brasil, siendo considerados adicionalmente requisitos funcionales relacionados descritos en la literatura científica y en las publicaciones de la Unión Internacional de Telecomunicaciones. La solución propuesta emplea una arquitectura de microservicios para la administración de datos, incluyendo tareas como la conversión de formatos, análisis, optimización y automatización. Para permitir el intercambio eficiente de datos entre servicios, sugerimos el uso de una estructura jerárquica creada usando el formato HDF5. Esta arquitectura se implementó parcialmente dentro de un proyecto piloto, que permitió demostrar la viabilidad de las ideas presentadas, realizar mejoras en el formato de datos propuesto y en los algoritmos analíticos. Los resultados señalaron el potencial de la solución para resolver algunas de las limitaciones del tradicional flujo de trabajo de comprobación técnica del espectro. La utilización del sistema propuesto puede mejorar la integración de las actividades e impulsar iniciativas de datos abiertos, promoviendo la transparencia y la reutilización de datos generados por este importante servicio público.

Palabras clave: espectro de radiofrecuencias; comprobación técnica del espectro; gestión del espectro; hdf5; ontología; detección Bayesiana; agrupamiento de datos.

Resum

El control i anàlisi d'ús de l'espectre electromagnètic, un servei conegut com a comprovació tècnica de l'espectre, és una part important del procés de gestió de l'espectre de radiofreqüències, ja que proporciona la informació necessària al flux de treball que permet el nostre estil de vida actual, interconnectat i sense fils. El número cada vegada més gran d'usuaris i el creixent ús de les tecnologies sense fils amplien la demanda sobre la infraestructura existent, exigint noves alternatives per a administrar i analitzar el gran volum de dades produïdes per les xarxes d'estacions de mesurament. Aquest estudi aborda aquest problema en proposar una arquitectura de sistema per a la gestió d'informació capaç d'augmentar la capacitat d'anàlisi d'una xarxa d'equips de mesurament dedicats a la comprovació tècnica de l'espectre. Aquesta proposta inclou una alternativa per a administrar les dades produïdes per aquesta xarxa, mètodes per a analitzar les dades recollides, així com una proposta per a automatitzar el procés de recopilació. L'estudi es va realitzar tenint com a referència els requisits de l'Agència Nacional de Telecomunicacions del Brasil, sent considerats addicionalment requisits funcionals relacionats descrits en la literatura científica i en les publicacions de la Unió Internacional de Telecomunicacions. La solució proposada empra una arquitectura de microserveis per a l'administració de dades, incloent tasques com la conversió de formats, anàlisi, optimització i automatització. Per a permetre l'intercanvi eficient de dades entre serveis, suggerim l'ús d'una estructura jeràrquica creada usant el format HDF5. Aquesta arquitectura es va implementar parcialment dins d'un projecte pilot, que va permetre demostrar la viabilitat de les idees presentades, realitzar millors en el format de dades proposat i en els algorismes analítics. Els resultats van assenyalar el potencial de la solució per a resoldre algunes de les limitacions del tradicional flux de treball de comprovació tècnica de l'espectre. La utilització del sistema proposat pot millorar la integració de les activitats i impulsar iniciatives de dades obertes, promovent la transparència i la reutilització de dades generades per aquest important servei públic.

Paraules clau: espectre de radiofreqüències; comprovació tècnica de l'espectre; gestió de l'espectre; hdf5; ontologies; detecció Bayesiana; agrupament de dades.

Table of Contents

Preface	15
1 Introduction	17
1.1 Motivation.....	17
1.2 Objective	18
1.3 Expected impact	19
1.4 Project organization methodology.....	19
2 Introduction to spectrum monitoring.....	21
2.1 The radio spectrum.....	21
2.2 Radio spectrum regulation	22
2.3 Spectrum monitoring.....	23
2.4 Spectrum monitoring evolution.....	26
3 Problem Analysis.....	29
3.1 User stories	29
3.1.1 From zero to hero	29
3.1.2 Do more	30
3.1.3 Danger! Danger!	31
3.1.4 I want it all! I want it now!	32
3.1.5 Enjoy the show	32
3.1.6 Is everything ok?.....	33
3.1.7 Everybody.....	33
3.1.8 Big boss.....	34
3.1.9 My team, my data	34
3.2 Data Sources	34
3.2.1 Measurement Data	34
3.2.2 Other data sources.....	39
3.3 System Requirements	39
3.3.1 Transparency and open data	40
3.3.2 Confidentiality and data protection	40
3.3.3 License issues and intellectual property	41
3.3.4 Security and network management.....	42
3.3.5 Interoperability.....	43
3.3.6 Usability and availability.....	44
3.3.7 Platform and IT environment	45
3.3.8 Efficiency and Scalability	45



4	State of the Art	47
4.1	Technological context review.....	47
4.1.1	Market survey	47
4.1.2	Information management system	49
4.1.3	Data acquisition.....	50
4.1.4	Data conversion.....	51
4.1.5	Metadata aggregation	54
4.1.6	Manage	58
4.1.7	Publish.....	59
4.2	Critics to the state of the art.....	59
5	Proposed Solution	63
5.1	Solution identification	63
5.1.1	Do nothing.....	63
5.1.2	Buy something now	64
5.1.3	Wait to buy something	64
5.1.4	Assemble from open software	64
5.2	Risk analysis	65
5.3	Solution design	67
5.4	Detailed design	71
5.4.1	Reference data and measurement index	71
5.4.2	HDF5 Measurement data files	72
5.4.3	System modules.....	76
5.5	Technologies used.....	84
5.5.1	Development Languages	85
5.5.2	Application framework.....	86
5.5.3	Automation framework	87
5.6	Pilot implementation	87
6	Implementation	91
6.1	Development of the proposed solution.....	91
6.1.1	Ancillary components.....	92
6.1.2	Data format conversion module.....	92
6.1.3	Data processing automation module	93
6.1.4	Indexing module.....	94
6.1.5	Data aggregation modules.....	94
6.1.6	Analysis modules.....	95
6.1.7	Reference database and WebGUI.....	97

6.2	Trials	98
6.3	Results.....	99
7	Discussion	109
8	Conclusions	113
8.1	Theoretical implications	114
8.2	Practical implications	114
8.3	Limitations and further research.....	115
9	References	117
10	Annexes	123
10.1	RFEye Node 20-6 Brochure.....	125
10.2	Message template used on the market survey	127
10.3	Detailed relational model	129
10.4	Detailed HDF5 file structure description	131
10.5	Code of the pilot implementation	139
10.6	Description of analysis algorithms	141
10.6.1	Emission detection	141
10.6.2	Emission clustering	143
10.7	WebGUI demonstration with Joomla!	145
10.8	The relation between the project and the MUGI subjects	151
11	Glossary.....	153



Table Index

Table 1.	Dissertation structure	15
Table 2.	Grouping of monitoring tasks	24
Table 3.	Risk, Impact and Countermeasures associated with the project.....	65
Table 4.	Optimization module trigger events, input and output.	78
Table 5.	Configuration module trigger events, input and output.....	80
Table 6.	Interface daemon trigger events, input and output.	81
Table 7.	Decode and conversion module trigger event, input, and output.	81
Table 8.	Indexing module trigger events, input, and output.	82
Table 9.	Data aggregation module trigger events, input, and output.	83
Table 10.	Analysis module trigger events, input and output.	83
Table 11.	Ancillary system components	92

Figure Index

Figure 1.	Maslow's hierarchy meme with Wi-Fi addition. Source [2]	17
Figure 2.	Traditional spectrum monitoring workflow	25
Figure 3.	Levels of analytic capability. Source: [15], [16], with adaptation.	26
Figure 4.	Example of radio spectrum visualization produced by CRFS application...	36
Figure 5.	Anatel standard workflow for the use of the spectrum monitoring network	37
Figure 6.	Spectrum monitoring data management GUI mock-up.....	48
Figure 7.	A simplified model for data workflow. From [32] with edition.	50
Figure 8.	Diagram representing HDF5 file format structure	53
Figure 9.	DTT channels 48 to 52 using ISDB-T transmission on adjacent channels. .	61
Figure 10.	The same band displayed in Figure 9, scanned with frequency bins of 590.62kHz width	62
Figure 11.	General view of the system architecture.....	68
Figure 12.	Basic relations associated with the spectrum measurement data.....	72
Figure 13.	Tree view of the proposed HDF5 file structure including only the objects on the first two levels.....	73
Figure 14.	Tree view of the proposed HDF5 file with highlight to the frequency sweep group.....	76
Figure 15.	Navigation flow for the Web GUI	84

Figure 16.	Automation of the file conversion process.	94
Figure 17.	HDF View interface presenting a converted data file with the noise level profile dataset	99
Figure 18.	Detection algorithm debugging output.	100
Figure 19.	The power level of emissions detected (A) 12,5kHz emission and (B) 25kHz emission.	101
Figure 20.	Channel with three distinct emissions overlapped	102
Figure 21.	Matrix profile distance (right) analysis of a spectrogram (left) for a model case.	102
Figure 22.	Matrix profile distance (right) analysis of a spectrogram (left) for a complex scenario.	103
Figure 23.	Normalized channel representation for comparison of two traces.	104
Figure 24.	Normalized channel representation for comparison of two narrowband traces.	105
Figure 25.	Normalized channel representation for comparison of a narrowband and a wideband trace.	106
Figure 26.	Distance matrix graphical visualization for 151 channels.	107
Figure 27.	Complete dendrogram representing clusters for the 151 channels.	107
Figure 28.	Dendrogram top part with most representative groups.	108
Figure 29.	The relational model for the measurement index	129
Figure 30.	Tree view of the HDF5 file structure including root elements and raw data groups.	131
Figure 31.	Tree view of the HDF5 file structure including root elements and analytical data groups.	132
Figure 32.	A detailed description of root objects on the proposed HDF5 format.	133
Figure 33.	A detailed description of objects within the time capture group on the proposed HDF5 format.	134
Figure 34.	A detailed description of objects within the frequency sweep group on HDF5 format.	135
Figure 35.	A detailed description of objects within the noise group on the proposed HDF5 format.	136
Figure 36.	A detailed description of objects within the channel group on the proposed HDF5 format.	137
Figure 37.	Home page of the system WebGUI provides access to open data repositories and user identification screen.	145
Figure 38.	The lower part of the WebGUI homepage also provides an area for information.	146
Figure 39.	References page give access to the information on reference tables of the system	146



Figure 40. Repository page gives access to raw data files, information on detected emissions and noise.....	147
Figure 41. User identification employs standard interfaces.....	147
Figure 42. Using a profile with administrative privileges allow the user to edit any of the system tables.....	148
Figure 43 Each table on the database is also accessible through an advanced editing page.....	148
Figure 44. Individual data files may be retrieved using a map and list interface.....	149
Figure 45. Data may be accessed by any information through advanced queries.....	149

Preface

The structure for this document was inspired by the recommendation from *Escola Tècnica Superior d'Enginyeria Informàtica*, ETSINF [1]. Changes were made on the sequence and on the hierarchy of some of the chapters and sections.

The adjustments from the referenced standard were made to better fit the project objectives and scope within the field of information management, directly associated with the master course, MUGI.

Some of the changes had also the intent to provide better linkage to more traditional structures used for academic dissertations, a correlation that is highlighted in the table below. This table also indicates a correlation of the content with the six stages of the software development process and is intended as an aid to those more familiar with different organization templates.

Table 1. Dissertation structure

ETSINF Guide [1]	Current Structure	Classical Structure	Software Development Stages
<i>Prefacio o prólogo</i>	Preface		
<i>Introducción</i>	Introduction Introduction to spectrum monitoring Problem analysis	Introduction	Planning & Analysis
<i>Estado del arte</i>	State of the art	Literature review	
<i>Análisis del problema</i> <i>Solución propuesta</i>	Proposed Solution	Methodology	Design
<i>Implantación</i>	Implementation	Findings	Implementation & Test
<i>Conclusiones</i>	Discussion Conclusions	Discussion Conclusions	Maintenance

The main body of the dissertation is contained from chapter 1, with the introduction, until chapter 8, with the conclusion. The beginning of each chapter presents a summary of its sections and subsections.

The chapter numbering continues beyond chapter 8 to enable easier referencing of the complementary material. Chapter 9 presents the bibliography, chapter 10 the annexes and chapter 11 the glossary.

Mathematical expressions were not included since there are no deductions or derivations that require extensive explanations. The specific mathematical background employed on various implementations can be found on the presented references, or at least easily derived from those available.

Throughout the text, some key information aspects that may represent a security risk to any of the involved organizations were omitted. It is expected that such intentional omissions on the descriptions may not represent a relevant issue to the understanding of the proposition, the replication or extension of the work.

The following conventions were applied to this document:

- Bibliography employ the IEEE citation style;
- Literal quotation uses double quote marks and italic font style;
- Non-English expressions are written using italic font style;
- Hyperlinks to the glossary are presented with underline and different font colour;
- Hyperlink to internet pages that are not bibliographical references, such as a data repository, a company or a product page, are included as footnotes.

The text was written by an engineer trying to approach a reader who may not have a great understanding of the telecommunications world, although have some grasp about its importance and about computer sciences.

1 Introduction

This chapter presents some essential information to understand the conducted work. The motivation is summarily presented in section 1.1 and the objective in section 1.2. The expected impact is presented in section 1.3 and a brief view of the methodological approach used in this project in section 1.4.

1.1 Motivation

When considering memes such as the one presented in Figure 1, it seems pointless to add any argument about the importance of wireless technologies on modern life, but what's often forgotten is the essential underlying effort continually employed into maintaining the infrastructure required to enable such lifestyle.

This work is about how data science and information management can help in such an effort.

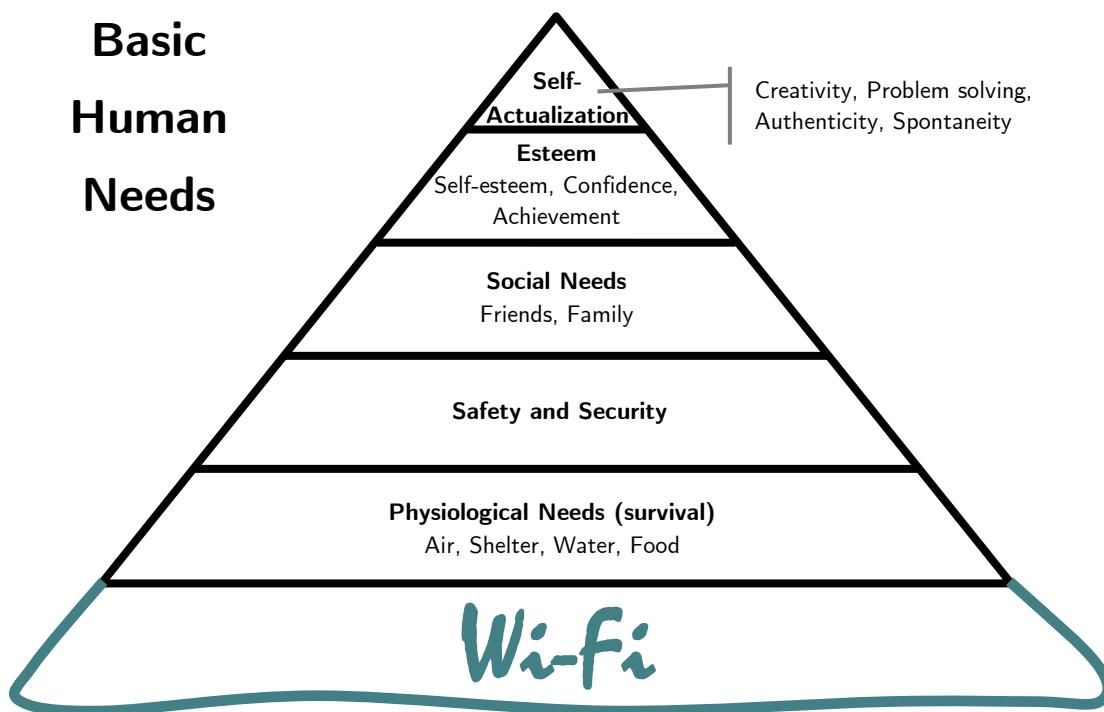


Figure 1. *Maslow's hierarchy meme with Wi-Fi addition. Source [2]*

To understand the mentioned effort to provide telecommunications, one must consider a key element to all wireless technologies, the shared propagation medium through which all radio signals must transverse to move information from one device to another.

This shared and open transmission medium is more usually referred to as “radio spectrum”, or, for short, simply “spectrum”.

The radio spectrum is considered a limited resource and as such is subject to an administrative process that has the objective of guaranteeing its availability by regulating its use.

The national authorities entrusted to administer the spectrum maintain specialized services for the systematic monitoring of radio emissions at an attempt to create knowledge about how the spectrum is actually being used and provide feedback to the regulatory process and eventually needed enforcement procedures.

Maintaining a spectrum monitoring service demands the use of a myriad of resources, including personnel and measurement instruments that are able to qualify and quantify the use of the radio spectrum.

In the past decades, these measurement resources evolved from huge man handled stations into small and autonomous data acquisition devices that may be deployed to create a spectrum monitoring network composed of hundreds of small sensor stations.

Such modern spectrum monitoring networks are able to collect huge amounts of data of which the analysis challenge the traditional operational procedures employed by national authorities, demanding innovative information management approaches that may, with autonomous intelligence, create the required knowledge about the spectrum.

The motivation for the present work comes from the recognition of such possibilities during the author activities on several projects associated with spectrum monitoring at the Brazilian National Telecommunications Agency, [Anatel](#).

[Anatel](#) supported this project on the terms of the ordinance 953 of July 12nd, 2017 (SEI/ANATEL 1647408).

1.2 Objective

The objective of this project is to propose an information management system architecture able to increase the analytic level of a spectrum monitoring measurement network.

This increase shall be translated into proposals to allow the implementation of the following features:

- to manage the data produced by such network;
- to provide the user with analytical tools to explore the spectrum data;
- to automate and optimize the data gathering process.

The proposed architecture is to be tested by a pilot software implementation of some of the functionalities over a specific spectrum monitoring network in use by [Anatel](#).

1.3 Expected impact

The proposed architecture is expected to provide the basis for the development of a new paradigm to manage spectrum monitoring data and control a spectrum monitoring network.

This architecture may be explored by the pilot implementation and on data retrieved by the Brazilian National Telecommunications Agency, [Anatel](#), providing an alternative that may solve the problem associated with the spectrum monitoring data management that has been juggled by different systems within that agency for the past 20 years.

Based on the user experience at [Anatel](#) and real operational data acquired, one may expect further development of the pilot implementation into a full realization of the architecture herein proposed, expanding it from a demonstration system to a capable framework for digital asset management and automation of the spectrum monitoring network.

The realization of the concepts here explored, with the introduction of automation and advanced analytics into the spectrum monitoring process, are expected to enable such intelligent measurement network to provide valuable insights and knowledge about the spectrum usage, reducing the costs and resources needed to perform this important spectrum management task, not only to [Anatel](#) but also to other administrations, that may join to share the benefits of the developed framework.

1.4 Project organization methodology

On this project, the development was conducted applying an incremental process model using agile methods [3], [4].

The software development for the pilot implementation is styled around a microservice architecture [5], [6]. The design of each module is handled as a set of iterations on the development process of the fully operational system.

A more detailed discussion about the criteria used for the selection of the architectural style and process model is presented in chapter 5.

The complete process is aligned with the presentation structure of this document considering the following topics:

- An expanded introduction containing telecommunications concepts that will be used throughout the text is presented in the chapter “Introduction to spectrum monitoring”.
- An initial review of the system requirements is presented in the “Problem Analysis” chapter.
- Literature, possible existing solutions and bottlenecks to be tackled are reviewed on the “State of the Art” chapter.
- Further refinement of the requirements and design of a solution is presented on the “Proposed Solution” chapter.



- The detailed description of the modules created is presented in the “Implementation” chapter. Individual iterations on the development process of each module are not discussed unless there are relevant results or lessons learned.
- A final discussion on the proposed architecture and obtained results is presented in the “Discussion” chapter.
- The project closure is presented on the “Conclusion” chapter.

For the problem analysis, it was decided to use the agile approach of “User Stories” [7] to enable a better understanding of some of the intended uses of the proposed solution. This was combined with a minimum description of essential system requirements in a more traditional approach.

Inquiries were conducted with users from various administrations to develop the mentioned “User Stories”. These interviews were not conducted under a rigid and structured script, being subject only to the initial suggestion of the objective stated for the present work and the expectations about the results. On two occasions, group meetings were conducted for the presentation of the project drafts followed by an open discussion.

Similar inquiries were also conducted with product manufacturers to gather information about the existing products.

No extensive qualitative study on the inquiries was conducted. These were simply summarised at the corresponding sections of the present document.

Such an approach was considered enough to provide the required information to achieve the project objective.

2 Introduction to spectrum monitoring

Considering the specificity of the object for the current project and the possible diverse public for the present material, it becomes necessary to provide some additional background information about the telecommunications world, including definitions that will be used throughout this study.

Subsections 2.1, 2.2 and 2.3 provide some basic concepts about spectrum management and spectrum monitoring. Later is presented a vision about the future of spectrum monitoring (2.4) and how the present work relates to such vision.

2.1 The radio spectrum

Firstly let us take a step back to the definition from the object of the present study and try to understand what really is the radio spectrum.

The word radio is a reference to what is called “radio frequencies”, that are defined from 9,000 Hz (9kHz) up to 3,000,000,000,000Hz (3THz). When electromagnetic fields oscillate at these frequencies they may self sustainably propagate on the free space in the form of a wave that is usable for telecommunications.

To enable different users to share the same communication medium without interfering on each other, a set of access policies to this shared medium must be put in place [8]. One of the most basic recognized sharing policies is simply the allocation to each user of a small frequency range, usually referred to as “channel”.

Although one may be overwhelmed by these frequency numbers and the insurmountable number of channels that should be available, in fact, the radio spectrum, like most of the natural resources, is restricted in its availability. At one hand by technological limitations and on another by natural phenomena that introduce effects such as attenuation, reflection and diffraction of the radio waves on their travel from one device to the other.

One obvious way to increase the spectrum availability is through the technological advancement that enables the use of a larger segment of the radio spectrum. As reported in [9], the usable portion of the spectrum has been systematically increased from about 1.5MHz in 1918 by a factor of 10 every decade in the first half of the XX century up to 40GHz by the end of the Second World War. This limit continued to be pushed afterwards but with much slower pace since the propagation effects render higher frequencies more limited in terms of possible usable applications on today's scenarios.

More than technological limitations, considering possible harmful effects to humans and the environment, there are power and directivity limitations on the transmission and reception that result on restraints to the distance and operation conditions for the transmitter and the emitter of any radio communication.



Such restraint in the distance, although imposing a hard limit on the possible applications, also enables the reuse of frequencies at separate locations, one essential aspect of all modern telecommunications systems, especially mobile phones and short-range devices, such as Wi-Fi. The geographical separation of users is another standard form of access policy to share the spectrum.

One may also mention other more sophisticated sharing policies, including the time-sharing and code sharing. Today, all these methods are applied in different forms and combinations to maximize spectrum availability.

For these reasons, it is accepted that the radio spectrum is a limited resource of public interest in its applications. With such understanding, soon in the history of telecommunications, it became clear that there was a need for some form of regulation to control and mediate conflicts associated with the use of the radio spectrum.

2.2 Radio spectrum regulation

The need for regulation on the telecommunications market came in fact before the creation of radio communications. It was born from the need of international cooperation and standardization for the interconnection of wired telegraph networks, with the creation in 1865 of the International Telegraph Union. Later, in 1947, it became the International Telecommunications Union, ITU, a specialized agency of the United Nations devoted to the coordination, standardization and development of telecommunications. [10]

The international activities of ITU are supported by regional and national regulatory authorities. Currently, ITU directory lists 191 telecommunications regulatory authorities in 181 countries [11]. At the regional level, there are 11 organizations [12].

These organizations are responsible for more than the management of the radio spectrum resource, they regulate the use of all needed resources for telecommunications, including numbering and orbital positions. Other important tasks include the promotion of cooperation between private companies, the enforcement of rules for network interoperability and interconnection [13].

In relation to the radio spectrum, the national authorities are responsible for the assignment of channels or groups of channels, also known as bands, to a specific user or group of users. Some frequency bands are also left open, for free and unlicensed use.

The different arrangements of assignment and licensing are usually under a larger organization framework on the national and international levels. Such arrangements are described in a higher abstraction level by the Allocation of different frequency bands to specific uses and services, and in a more specific and formal way, by the allotment of channels within these bands. Such arrangements are agreed by all countries participating in the ITU conference and are accepted as an international treaty.

All these plans and arrangements are engineered to minimize the possible cases of interference between different users and services, i.e. situations where the use of a frequency band by someone, including activities not related to telecommunications such as industrial and medical applications, may impair or even prevent the use of the radio

spectrum to some other activity, again including uses not related to telecommunications such as radio astronomy and passive remote sensing of the earth.

One important aspect of the spectrum engineering task is that it usually assumes a conservative approach, it means that to improve reliability it includes some level of inefficiency, such as guard bands and channels that are assigned as “taboo” and left vacant in order to avoid undesirable [intermodulation](#) effects.

Even though all these preventive work and planning is conducted, one still is routinely faced with technological advancements and unexpected result of human activities.

On the technological side, the frequency plans need to be periodically revised, something that on the ITU level is performed every few years. On the national level, the revision of frequency plans has the added complexity associated with the revision of existing [assignments](#) and the direct consequences of such revisions on the activities that depend on the use of such frequencies.

This engineering task of reassigning frequency bands to different uses is known as spectrum refarming. A notable example of such activity on the recent years has been the change on broadcast terrestrial television services from analogue to digital technology. This change resulted in a reduction on the required spectrum band for this service by allowing two channels to be assigned closer together. The frequency surplus was reallocated to other services, such as for mobile telecommunications.

On the human activity side, one is faced with the mysterious combination of creativity and laziness resulting in the uncontrollable creation of new devices that uses, intentionally or not, the radio spectrum. Sometimes the lack of maintenance and knowledge may turn a house appliance or an advertisement panel into a source or nuisance to many spectrum applications. One may also find situations when a user or service has been assigned to a specific channel or band but does not make use of it due to any reason whatsoever, resulting in further inefficiency on the spectrum use.

In all these cases, the proper management of the radio spectrum demands a closed-loop process, which means that one must not only rely on the theoretical assignment and planning, but also observe the actual use by the numerous services. To close this loop the national authorities are advised, on the form of several recommendations issued by ITU, to maintain a spectrum monitoring service.

2.3 Spectrum monitoring

Considering the obligation derived from the ITU Radio Regulations, that, as mentioned, are under an agreement that stands as an international treaty, the spectrum monitoring services are entrusted with the following activities according to the Spectrum Monitoring Handbook [14]. On this citation was added the parenthesis with letters from A to H for easier later referencing.



“...

- *monitoring emissions for compliance with frequency assignment conditions; (A)*
- *frequency band observations and frequency channel occupancy measurements; (B)*
- *investigating cases of interference; (C)*
- *identifying and stopping unauthorized emissions; (D)*

...”

Additionally, other activities may also employ spectrum monitoring services, including: [14]

“...

- *assistance on special occasions such as major sporting events and state visits; (E)*
- *radio coverage measurements; (F)*
- *radio compatibility and EMC studies; (G)*
- *technical and scientific studies. (H)*

...”

These spectrum monitoring activities may be grouped into three very distinct types according to the regularity in which they are performed, the type of equipment and automation that is most suitable. Such grouping is presented in the following table.

Table 2. Grouping of monitoring tasks

Focused monitoring tasks	Monitor a specific radio spectrum band for a specific period to perform activities such as above assigned with letters C, D, E and F. This type of task would be employed to detect anomalous spectrum usage on a specific band, and time, considering a reference based on another period or location or to evaluate a specific use condition or event Since it is a form of monitoring strongly delimited in time and scope, it is usually conducted through manual operation or using dedicated configuration scripts. It is common for activities of this type to employ portable equipment, mobile or transportable monitoring stations.
---------------------------------	--

Systematic monitoring tasks

When a wide part of the spectrum is scanned, including several services, during a large period to perform activities such as above assigned with letters A, B, G and H.

Usually employed to attain broader objectives, such as those derived from the Radio Regulations.

To such task, the spectrum sweep should be optimized to achieve the highest possible reliability in the process of characterization of specific spectrum use patterns, channels and bands.

The ideal scenario for this type of monitoring would be as a background task, to be continuously executed, i.e. 24x7 regime, by fixed or transportable monitoring stations.

Test and configuration monitoring tasks

Including several measurement procedures designed to evaluate the characteristics of the measurement equipment and record the operating conditions such as to enable predictive and preventive maintenance activities and provide an operational baseline, e.g. evaluate system noise to aid other measurement procedures.

This is a regular procedure that should be performed on all equipment, as an intercalary evaluation between more complex and demanding calibration procedures.

Such intercalary evaluation is important to identify equipment malfunctions and avoid costly correction over decisions based on erroneous results.

Traditionally and much as described in the ITU documents, especially the Spectrum Monitoring Handbook [14], the spectrum monitoring activities provide the information to the spectrum management services, or SM for short, using mostly systematic monitoring tasks that are easier to automate. These tasks are usually performed as represented in the following diagram.

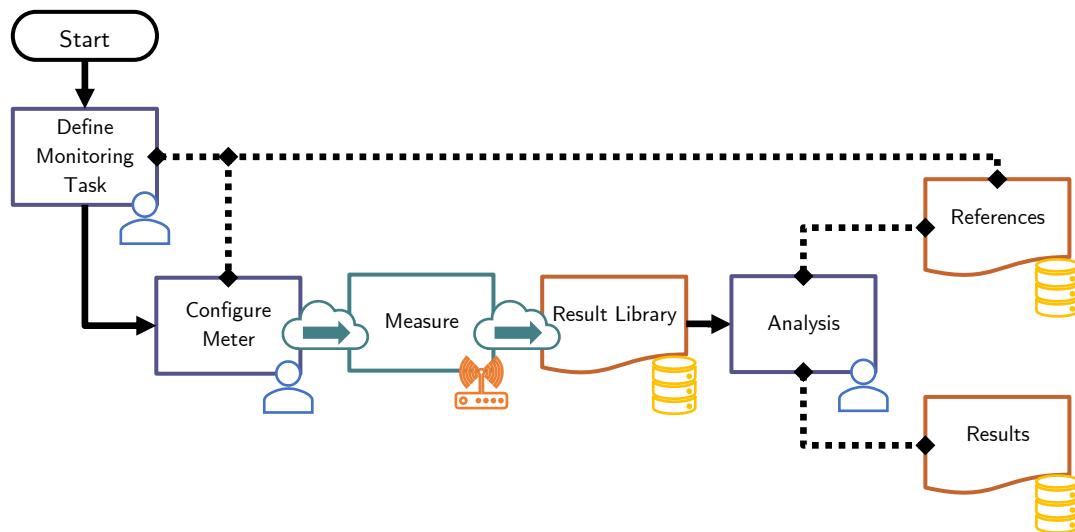


Figure 2. Traditional spectrum monitoring workflow

On the above diagram, tasks such as configuring the measurement equipment and performing the analysis are performed by engineers and technicians using specialized analysis software.

Data storage usually is centralized in order to provide some level of backup security, but often there is no standard about the organization of such storage, this aspect is generally being left open to be explored by different measurement equipment and system manufacturers, as part of their solution.

The analysis tasks are also performed within the solution framework of each manufacturer and the results, in terms of reports, may be used by the authority responsible for spectrum management in different ways, e.g.: on conflict resolution; on the enforcement of license conditions by the suppression of interferences and unauthorized use of spectrum; to support spectrum planning activities such as refarming; etc.

2.4 Spectrum monitoring evolution

As described, the traditional spectrum monitoring process is driven by the manual and explicit creation of demands and dependant on manual tasks for analysis and configuration of the equipment to answer such demands.

To evaluate possible evolution scenarios to the spectrum monitoring services, we may take as reference the analytic capability model described in [15], [16], such as reproduced in Figure 3 with some adaptation.

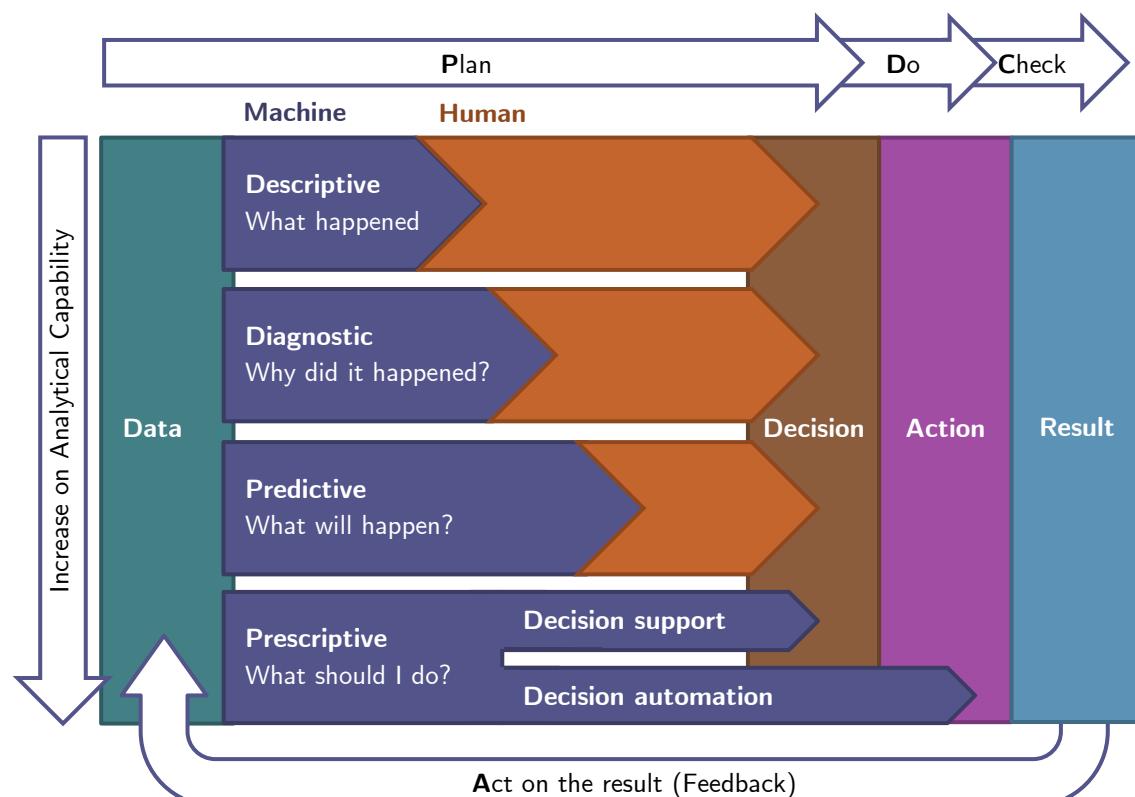


Figure 3. Levels of analytic capability. Source: [15], [16], with adaptation.

Using this model, one may realize that the spectrum monitoring process previously described applies only the first two analytical capability levels. For example: when specialized personnel use software applications to derive reports that **describe** the spectrum usage conditions; or when measurement equipment is used to **diagnose** and solve radio interference cases.

It may be argued that the third level, of predictive analytical capabilities, may be used on special conditions, such as when finding the best site or route to perform measurements by using simulation tools or when using measurement data to optimize a simulation model for radio communications.

Considering the author's experience on the spectrum monitoring services, including discussions with national authorities from several countries, such spectrum monitoring services workflow usually implies on reduced use of the available equipment resources in terms of hours per week since it demands specialized personnel to analyse the data and produce reports and diagnostic.

One can hypothesize that such limitation may reduce the benefits that such resources could provide in terms of knowledge about the use and availability of the radio spectrum simply because of the limited availability of personnel and the consequent prohibitive cost of manually translating the large amount of collected data into actual insight about the radio spectrum usage.

To achieve the maximum possible use of the measurement equipment, e.g. on a 24x7 basis, one envisions, as an alternative, the use of a fully automated workflow, applying the highest levels of analytical capability. In other words, the monitoring service must autonomously adapt to variations on the spectrum use and characteristics to improve the chances of collecting relevant information concerning multiple objectives.

In addition to the analytical capabilities to perform autonomous data gathering, the spectrum monitoring service must also be equipped with the tools to derive knowledge about the spectrum usage from this data. Such knowledge is the key to the larger feedback provided by the spectrum monitoring to the spectrum management services and radio spectrum regulation, in general.



3 Problem Analysis

This chapter describes in more detail the problem of attaining the objectives proposed in the introduction. It first employs the concept of user stories (3.1) to provide a general idea about the user needs that may be addressed by the project. It follows with a detailed description of the data sources (3.2 and 3.2.2) and finishes with an analysis of other references that create boundary conditions that must be observed by the project (3.3).

3.1 User stories

To enable a better understanding of the requirements for the implementation of the proposed information management system, this session presents a small collection of user stories, on the spirit of agile project management methodologies.

These stories are supplemented by comments such as to be self-sufficient since the agile scrum meeting interaction is not possible on a dissertation format.

The user stories were collected by interviews and discussions with employees working on different spectrum management roles, mostly employed at [Anatel](#) but also on other authorities that have a regular presence on ITU conferences on topics related to spectrum monitoring. Meetings were also conducted with representatives from different manufacturers.

This loose approach to the prospective initial study is not considered a strong limitation of the present study since it is out of the scope of the intended objective to conduct a full market survey and due to time limitations. Such a study may be considered in the future development of the concept to provide a better understanding of users' needs, particularly relevant to the development of the user interface.

The subsection titles try to create an easy reference to the user story context. The story itself is presented as a highlight inside a rectangular box with rounded corners.

3.1.1 From zero to hero

Users want to see the existing emissions, their patterns and behaviour, but cannot provide background information to start.

This story may be divided into two parts, the first is really the core of the business and associates to the very definition of spectrum monitoring. The visualization methods have ample demonstration on reports and on the referenced literature, they will be discussed in detail at this point and the importance of this story is centred around the second part.

The second part is a bit more tricky and to better understand this story, one must realize that the traditional approaches to spectrum monitoring rely on reference information about the channel [allocation](#) to specify essential measurement configuration parameters such as filter bandwidth, frequency step and revisit time.

The channel frequency information may also be used to consolidate data into metrics such as [occupancy ratio](#), i.e. the ratio between the time a certain frequency is used in relation to a reference time period. Reference information about the [assignment](#) may also be used to evaluate if a detected emission is authorized or not.

Regardless of the importance of [allocation](#) and [assignment](#) information, these might not be available in many instances, either because the frequency band of interest is not subject to strict licensing, such as the case for the band used for Wi-Fi, or simply because the license information is not available to the user in a usable format due to limitations on other systems, such observed when one is faced with a strong vendor lock or some data exchange incompatibility.

Apart from the information from spectrum management databases, there are other pieces of information that may limit the effortless operation of the traditional monitoring network. The most notable is the frequency band-specific noise floor, that is required by most of the carrier detection algorithms. An automated system should be able to determine the noise floor with as little input information as possible.

The main lesson on the reference story is that, on such situations where limited or no information about the spectrum is available, the monitoring system should be able to perceive its environment and generate meaningful results by inference of channels and radiofrequency environment characteristics.

3.1.2 Do more

Users want to get the maximum usage from the existing measurement equipment.

How to get the most from existing measurement stations is a matter of long discussion.

The desirable approach would be to maximize efficiency taking as measure the rate of resulting benefits in relation to the acquisition and operational costs of the measurement equipment. Qualifying and quantifying the benefits is not an easy task. In fact, it may not be feasible since it is hardly possible to predict the outcomes from a new data product before it is made available to all possible users and for a reasonable period of time.

One important note here is that users for the spectrum monitoring data may include not only the spectrum management authority but also operators and the scientific community. Such broad availability may enable unexpected benefits, such as technical developments, better quality on projects submitted for new licenses, etc.

As an initial approach, one may then consider that the spectrum monitoring network should aim at collecting as much data as possible, making it as widely and easily accessible as feasible, and leave the matter of benefit to be explored by a user community as wide as one can muster.

On taking this approach, some additional considerations must be made in relation to community creation and engagement and the interfaces that shall be created with this objective.

3.1.3 Danger! Danger!

Users want to be warned when something unusual happens on the spectrum.

An important task of spectrum monitoring is the detection of anomalous behaviour that may disrupt the operation of telecommunication services. Searching for odd emissions may demand the evaluation of a huge amount of data, something that may not be feasible by manual inspection.

More sophisticated systems may perform this task by spatial separation (direction of arrival) and geolocation of the emission sources, but to enable the use of simpler measurement equipment, the alternative is to emulate the human analysis, that is, evaluate each emission, how it uses the spectrum in terms of power and time distributions.

The use of alarms is not uncommon on spectrum monitoring systems and they are usually based on power level mask. i.e. an alarm is raised whenever the power level rises above a previously defined threshold. Most systems allow for the definition of a threshold with different frequency resolutions, down to each individual frequency bin.

Unfortunately, it is not uncommon that the problematic emission may be brief in time, rare in occurrence and/or be on the same band of other intermittent authorized emissions. In such cases, regular users push the alarm threshold above the required minimum to detect the interference, hindering ineffective the use of a simple threshold-based alarm.

A complementary approach to improve the chances of detection would be to use more of the information available on the spectrum trace to separate regular users from interferences, such as by characterizing modulation characteristics and/or the time behaviour patterns. For example, an unmodulated signal presents a power distribution over the frequency spectrum that is quite different from an FM modulated signal; a phone call has a longer duration than the communication observed on push-to-talk communication, such as that observed between an aeroplane and a control tower.



3.1.4 I want it all! I want it now!

Users want to see the evolution and impact of new technologies on the radio spectrum environment and usage conditions.

Users want to know the usage conditions at any frequency band without the need to wait for the data to be collected.

To have data available on-demand about any frequency band, at any time, the only alternative is to systematically collect as much measurement data as possible and, timely, index it into a management system that facilitates the following retrieval.

One important implication of the continuous operation of the monitoring network is the organization of the data storage, how to reduce the data with minimum loss of information and how to discard it.

Other implications are the requirements in terms of data indexing, retrieval and visualization.

3.1.5 Enjoy the show

Users want to know if the spectrum allocated to an event is really being used.

In contrast to the previous stories, where the user displays interested in the long term and continuous operation of the system, there are some unique use scenarios where the interest is focused in time, locations and frequency bands. This was earlier defined as focused monitoring.

Examples of such scenarios include major sports events, e.g. the Olympic Games, and the deployment of new technologies, e.g. analogue TV switch-off.

The need for focused monitoring implies that apart from the autonomous and continuous operation, the monitoring network must allow the users to direct the operation. This dual mode of operation needs to share the same measurement equipment resources, which means that some prioritizing method must be put in place.

3.1.6 Is everything ok?

Users want to know if the equipment is working fine.

Another issue that often rises when managing a network of spectrum monitoring equipment is the maintenance of the constituent measurement devices.

Applying [SNMP](#) or remote control automation over the monitoring stations may provide basic information about the monitoring equipment, such as if it is active or not, or even more details such as temperature and control voltages on equipment that provide some form of self-test or onboard monitoring. These alternatives usually will not provide information about the measurement capability, for example, if there is some level of deterioration of antenna systems and cables, something critical to the reliability of the obtained results.

By continuous monitoring, it is possible to consider the use of multiple emitters to establish an operational baseline and the use of a threshold around this baseline to set-off alarms in the event that a sensible variation on the overall results are observed.

This demands the creation of specific reports and analysis tools dedicated to the maintenance of the spectrum monitoring network.

3.1.7 Everybody

Users want the data shared through the organization in a flexible way. Analysts from various locations may be working on the same data. Lessons and patterns learned in a project should be available and reusable by all.

[Anatel](#) is an organization with authority over a large territory. There are offices in 27 cities, some of them located more than 6.000km apart. The number of personnel in each office also varies a lot. The head office houses more than 500 people while the smallest offices may have as few as 5 public employees permanently assigned.

This scenario poses a challenge to the creation of an organizational identity and the development of human resources. It is not uncommon at [Anatel](#) that some department or local office focus their expertise on some technique or method, taking advantage of the background knowledge from the existing personnel.

Information technologies are essential to solve this problem, that is in the core of the present user story. i.e. provide tools to allow a geographically sparse organization to maximize the use of its human resources by sharing content and knowledge between all departments.

Aside from this operational mode, an underlying feature of this user story concerns data reuse and it implies in the need to make the gathered data available with as little restriction as possible.



3.1.8 Big boss

The administrator configures the system.

One may foresee that the system may require administrative actions, such as the edition of reference parameters that may affect all users and thus might be better managed by a smaller group of super users.

Different user profiles must be available to address security issues.

3.1.9 My team, my data

Users want quick access to data and may want to restrict access to data within a smaller group of users.

Users may organize their work within groups or task forces, with greater common interest within those participating in such restricted party. This shared interest needs to be recognized by the system to promote easier communication between team members when sharing data or collaboratively working on the analysis of a dataset.

Other aspects that may arise is the need for restriction on the access of a certain part of the data due to privacy or security issues. Such restriction may affect only a certain group of users, again generating the need for the creation of users, user roles and groups.

3.2 Data Sources

The user stories are centred on the information management problem associated with the spectrum monitoring, thus, to support the system development is essential to understand the data sources that provide such information.

3.2.1 Measurement Data

As previously presented, this work is centred around the data management problem associated with [Anatel](#)'s spectrum monitoring network.

The specific example chosen within [Anatel](#)'s spectrum monitoring structure is representative of most commonly available system, such as described by ITU references [14] and is similar to those described in several references found in the literature and used throughout the present work [17]–[19].

The complete spectrum monitoring network used by [Anatel](#) is composed by several fixed, mobile and transportable stations from 4 different vendors and several portable measurement equipment, adding at least 4 other vendors to the list of measurement equipment suppliers. This myriad of platforms and manufacturers places difficulty in

creating a framework able to consolidate all data into a single view since each vendor has its own protocols and interfaces.

To create a functional pilot system and proof of concept, we are going to restrict our analysis to the elements that may take more benefit from the new system due to the number of stations, nature of monitoring performed, existing workflow and documentation. For these reasons, it was selected to use as reference the network composed by the equipment model RFEye Node 20-6, manufactured by [CRFS](#).

This equipment is used by [Anatel](#) on a fixed or a transportable configuration. A total of 184 units are available.

Each measurement equipment is composed by a Debian Linux [SBC](#) that controls various communication interfaces (Ethernet, USB, 3G Modem), local storage (500GB SSD) and sensors (radio spectrum meter, GPS, UPS, temperature, voltage). The main sensor is the radio spectrum meter, composed by a radio receiver with digitizer and real-time FFT engine. This sensor can perform measurement on frequency bands from 20MHz to 6GHz with bin width down to 10Hz and instantaneous bandwidth of 20MHz. A brochure with a basic description of this equipment is presented at annexe 10.1.

This equipment, as deployed by [Anatel](#), is suitable to perform long term analysis about the radio spectrum such as described as systematic monitoring. This is another important reason for the selection of this specific equipment as the primary data source.

To better understand the data produced by such equipment, one needs some basic understanding of how the measurement is performed and the meaning of the generated results.

In short, the equipment works by filtering out a segment of the radio spectrum and digitizing it for a brief period. The digitized waveform is submitted to a Fourier transform, resulting in its representation on the frequency domain. This process is sequentially repeated over several segments of the radio spectrum to create a view of a wider frequency range and also may be repeatedly conducted on the same frequency range in order to provide a view about the time behaviour of the existing emissions.

The result is a sequence of data points, where each point corresponds to the power level measured at a specific frequency, at a specific time and location. Figure 4 presents an example of a visualization of such results as provided by the [CRFS](#) software used by [Anatel](#) to remotely operate the monitoring stations in real-time.

To provide a concrete measure about the volume of data generated by such a device, the visualization presented in Figure 4 corresponds to the content of 37 measurement data files with a total volume of about 1.4GB of data stored using a lightly compressed 8-bit format for the power level representation.

In this example, a single 20MHz band is monitored by a fixed measurement station for a period of about 7 hours and 44 minutes. This band is divided into 51 200 frequency bins, each with about 390Hz. The revisit time for each bin is of about 1 second, which means that the data representation presented on the visualization corresponds to a square matrix with dimensions of 51 200 columns (frequency) and 27 830 rows (time).

It is not unrealistic to consider that each station could produce about 4.35GB of data each day and a network composed by 184 stations, operating on various levels and roles, could easily produce more than 500GB of data per day.

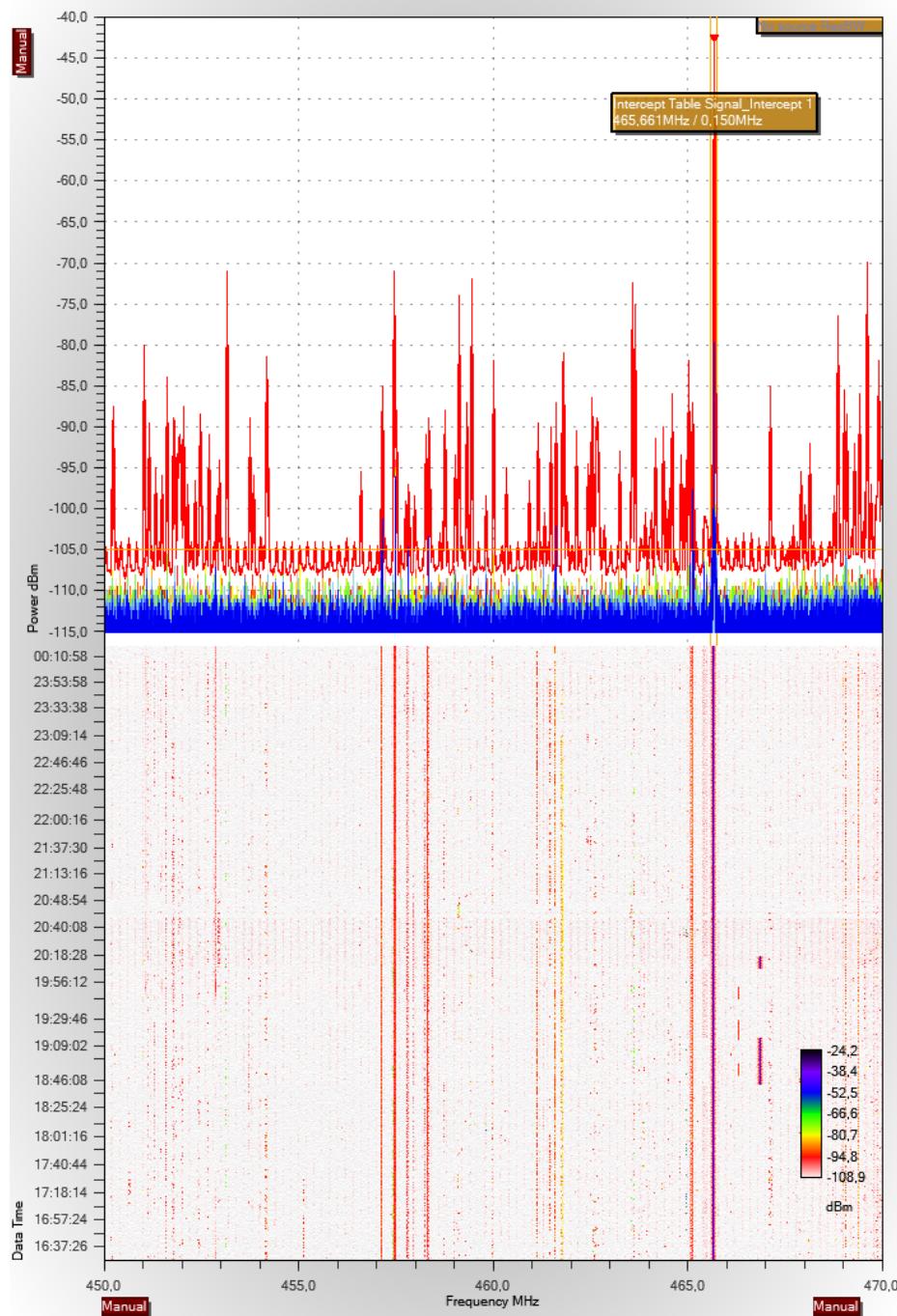


Figure 4. Example of radio spectrum visualization produced by [CRFS](#) application

Going a step back to review the above picture, on the upper part is presented a set of spectrum traces, that is, a set of scatter plots presenting the variation of the power level indicated in dBm, measured at the equipment entry port as a function of the frequency indicated in MHz.

A peak on the power level marks the presence of emissions. The baseline defined by the minimum level mark the noise floor.

On the lower part of the illustration, the power level is presented as the colour associated with each point on a bi-dimensional plot, where the same frequency range is represented using the horizontal axis and the time is represented using the vertical axis.

From the perspective of generated data, the equipment provides several modes of operation and two main interfaces/protocols. Again, for the present work we are going to limit ourselves to the main topology used by [Anatel](#) for systematic monitoring at fixed locations, a use case closer to the intended application

The diagram presented in Figure 5 illustrates the integration between the workflow and the data exchange on the existing network for a pair of workstation and measurement equipment (RF Sensor), integrated through a central server.

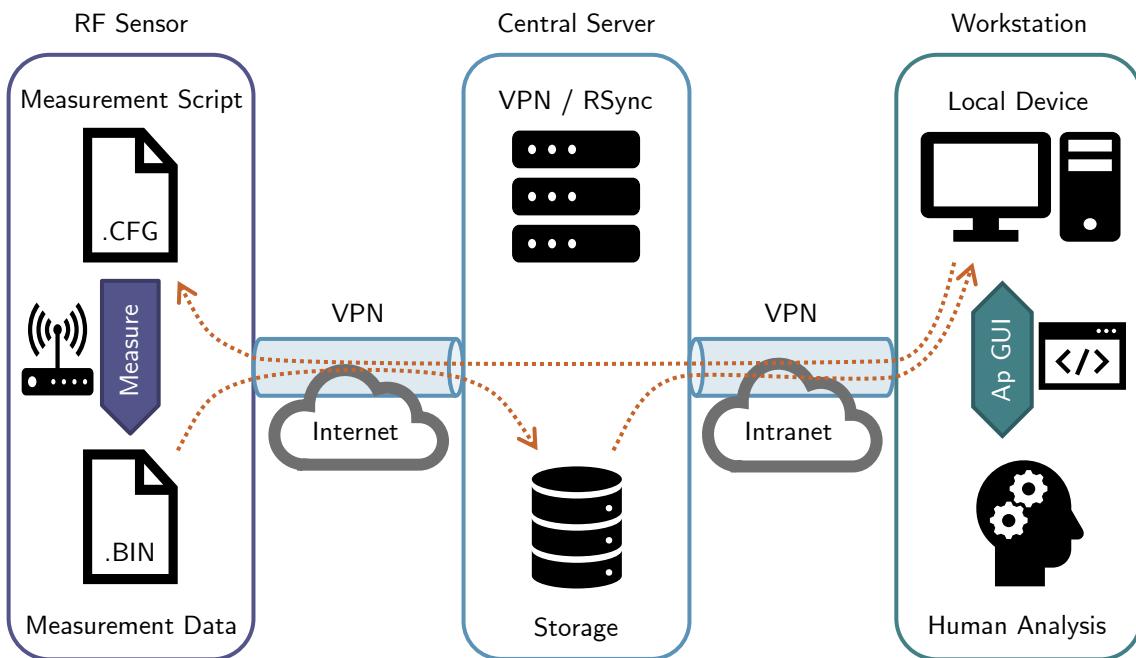


Figure 5. [Anatel](#) standard workflow for the use of the spectrum monitoring network

The operation workflow for the equipment here considered is the one provided by [CRFS](#) in the form of the RFEye Logger application. This application is used by [Anatel](#) much as depicted in Figure 2, that is also aligned with the previously described traditional spectrum monitoring workflow.

The process starts with the user at the workstation, by the definition of a measurement task as a set of commands and parameters listed into a plain text file using a specific syntax.

The measurement script file referred above by the extension “.CFG” must be transferred to the measurement station and loaded into the logger application using the web interface provided by the equipment.

On the standard operation, security is provided by limiting the access to the measurement stations through Virtual Private Network tunnels created through a central server. Password access control is also provided to critical functions of the measurement station. This is essential since many of the stations might be operating directly connected to the internet, even using mobile networks.

The “.CFG” script file is executed by the RFEye Logger application that runs at the measurement station. The results are stored at the measurement station local storage.

A complete description of the “.CFG” script file standard is presented on the [CRFS](#) RFEye Logger documentation [20].

To facilitate the data transfer and reduce the risk of data loss, the data generated by continuous monitoring is segmented into several files, usually by size. e.g. 40MB file. These are binary files using a [CRFS](#) proprietary format and are identified by their extension “.BIN”. A complete description of this format is also provided by [CRFS](#) [21].

The Linux utility rsync¹ is used to transfer the files from the measurement station to the central server. In the case of [Anatel](#), there is no additional indexing of the data, it is organized in folders and may be sorted by basic file metadata such as name and creation date. This implies that names and folder structure are usually an important configuration parameter.

The users access the server using tools like FTP or through a standard web sharing service, therefore they need to download the data to their local workstations, where they may be processed using specialized tools, such as “RFEye Data Tool” and “RFEye Site” to produce analysis and visualizations such as presented in Figure 4. After analysis figures such as the one presented will illustrate reports and be used to support spectrum management decisions and future spectrum monitoring activities.

Apart from the radio spectrum trace data already discussed, the RFEye Node 20-6 monitoring stations and associated software framework may be used to acquire other data types about the spectrum.

Another type of data that may be acquired by this equipment is the raw digitized waveform at a fixed centre frequency. This data is often referred to in the literature as I/Q data. This format is not used on the current pilot project since it is not extensively used by [Anatel](#), it is more demanding on storage and transfer resources and finally because it requires more signal processing functions to be of use, thus extrapolating the scope of this initial work. Nevertheless, some discussion about this format and its uses are presented throughout this document.

A third data type that may be produced by the RFeye Node stations is the processed frequency occupancy based on a simple energy detection algorithm. This processed trace information is not further considered in this work since it limits the available information in such a way that would compromise the results and ability to further extend the system capabilities and answer to the requirements underlined by the presented user stories.

As a final note on the data collected, it is important to highlight that each RFEye Node is equipped with GPS and time reference modules that enable them to register the precise location and time of each measurement. This information along with onboard audit data such as temperature, power supply status, etc, are also recorded within the “.BIN” data file and are independent of the selected type of spectrum data to be stored.

¹ <https://rsync.samba.org/>

3.2.2 Other data sources

Apart from the measurement data, a valuable information source to enable comprehensive analysis and visualization of the spectrum monitoring data is the spectrum management information associated with the [allocation](#), [allotment](#) and [assignment](#) of frequency channels and bands.

In the case of [Anatel](#), only the [allocation](#) tables are currently available as a SOAP web service. The full description of this service is available under a WSDL definition published within [Anatel](#) internal knowledge base and the service itself is restricted to the intranet.

From this service is possible to get most of the information associated with the [allocation](#), such as the telecommunication service names, established channels, channel designations and associated regulatory base.

The [allotment](#) tables are available only as HTML and PDF reports and one may conceive the use of text scrappers to retrieve the required information about the spectrum usage for a more immediate and reference use. It's important to highlight that the [allotment](#) is not constantly changed and, when changed, not extensively. Even a manual extraction of the data is a viable alternative, although the use of a properly configured web service would be preferable.

The [assignment](#) database represents a more complex issue since the available data sources are sparse and in various formats, according to the desired service. This considered the information available at the Brazilian national open data portal² and specific reports provided by [Anatel](#) systems to each service or group of services. Currently, the most efficient way to access the complete dataset would be through direct queries to the license DBMS or to a mirror of this DBMS created just for this application.

Even though there is a lot of dynamicity on the license database, for spectrum monitoring purposes, it would be acceptable to have some delay on the sync from the transaction DB to a mirror database. One must realize that most spectrum monitoring tasks that can be automated, such as discussed in the current project, does not require immediate human intervention, allowing for such lag on the data without much risk for misjudgement.

3.3 System Requirements

Having a general understanding concerning the data that should be handled and user-desired functionalities, it is important to consider non-functional requirements that may provide additional boundaries to the system architecture development.

At this, one must first consider that the proposed system is expected to be deployed within [Anatel](#) organizational infrastructure and IT environment, thus in the context of the Brazilian regulations and existing contracts and agreements, although parallel conditions may apply to other cases and administrations.

² <http://dados.gov.br/>



A complete list of functional and non-functional requirements is almost unattainable at the current development stage. Again, going back to the idea of an agile development process, we are going to concentrate our attention on the key aspects that must be observed, taking special attention to legal and ethical considerations.

3.3.1 Transparency and open data

The Brazilian National Open Data Infrastructure was created in 2012 [22] and later formalized as a policy for the Brazilian federal government in 2016 [23].

[Anatel](#) is the signatory of the open data initiative and currently holds 60 datasets published on the Brazilian open data portal³.

Even so, there is very little information about spectrum monitoring, mostly because reports published by [Anatel](#) are descriptive in nature and in PDF format, not as structured tables compatible with the open data portal requirements.

Another limiting aspect to the full disclosure is that many of the issued reports are produced within the context of regulatory enforcement and thus subjected to confidentiality until the due administrative process is concluded and often even afterwards, due to privacy and data protection issues.

Apart from an active engagement into the open data framework to include spectrum monitoring datasets, it's important to highlight the obligations related to the information access law [24], that allows any citizen to request access to public generated data. In the case of spectrum monitoring, most of the time the Agency is unable to affirmatively answer such demand in a systematic and organized format, at reasonable operational costs.

The proposed new system could solve such limitations by structuring the spectrum monitoring data into an open format and decoupling the measurement information from the actual enforcement activities and analysis, i.e. having individual reports and analysis stored into the separate repository used by [Anatel](#) as a document management system.

As a short conclusion to this subsection, the use of an open standard to store the spectrum monitoring data will enable [Anatel](#) to improve its policy towards open data and active transparency. As such, this alternative should be preferable and considered into the system requirements.

3.3.2 Confidentiality and data protection

Data protection in Brazil is defined by a recent legal framework [25] and follow the same basic principles of the General Data Protection Regulation in Europe [26] and similar regulations on other countries.

When considering spectrum monitoring data such as described in the previous sections, one must realize that is extraordinarily little confidential information contained on the raw measurement data.

³ <http://dados.gov.br/>

It becomes a matter of concern when the raw data is manually associated with metadata that identifies a specific spectrum private user or when it may be used to identify such user through additional signal processing.

One important aspect concerning spectrum data is that most of the licenses issued by [Anatel](#) are already public in nature. Such publicity means that is already publicly available the association between licenses and several spectrum usage parameters, such as channel frequency and power.

With this one may conclude that there is no need for concern about the public availability of spectrum measurement data. In fact, one may say that such publicity is required to allow engineers to estimate the operational conditions for new systems and solve probable conflicts before any transmission device is deployed.

Even so, still being of interest to create a system feature that enables the user to flag specific frequency bands as of restricted access. Such a feature may be useful to allow greater privacy on the usage information of specific bands, such as the ones used by the military and security forces.

3.3.3 License issues and intellectual property

There are two aspects to consider in relation to intellectual property. Firstly, the use of components on the system that are restricted by patents or copyright. Secondly, under which terms the created system will be made available.

As a rule, that should be considered on the requirements, the proposed system should not make use of components that are commercially restricted by intellectual property rights. Mostly due to cost restrictions, since the final implementation is not supported by any funds or intended for commercial use, which restricts the project ability to any form of payment for rights.

Nevertheless, the proposed system will need to interface with existing measurement equipment, that may employ proprietary formats and protocols of which copyright and patents may play a restrictive factor.

For the current pilot project within [Anatel](#) application context, one needs to consider that most of the acquisitions from that agency on the last 7 years included specific clauses that enabled the use of the available interfaces with the measurement units within its business, and thus encompassing the current project.

In any case, one may consider that disputes may arise in some cases and disrupt the proposed system functionality. To minimize such risk, the system architecture must be flexible enough to keep its operation regardless of the interruption of single interfaces. Since [Anatel](#) monitoring network topology is composed of equipment produced by several manufacturers, it may be considered resilient to some level of disruption by one or two manufacturers.

Such a concept of a resilient and modular system may be realized using a microservice architecture template, something that may be noted as another system requirement.



Considering under which terms the system will be made available, one must ponder that future maintenance and development may be too costly for a single entity like [Anatel](#) to keep. Also, one must consider the benefits of sharing governance in terms of new applications and modules that may be developed by a diverse community of users and partners.

The alternative is to distribute the software as an open-source initiative and promote the engagement of other entities, e.g. in the academia and other spectrum management authorities. To enable such an approach, the system must be made available as unrestrictive as possible, for example, using MIT license [27].

3.3.4 Security and network management

On this section, the security analysis includes only a general description of key elements. Due to the public nature of the present work, discussion about specific components, versions will be avoided.

The spectrum monitoring network used to create the pilot application, as described in Figure 5 and the accompanying text, employs a security scheme that facilitates the implementation of the current system with minimal additional effort.

Since monitoring stations and the server communicate through the public network, all are equipped with firewalls that block all traffic except on specific ports and services.

All measurement data is transferred through encoded [VPN](#) tunnels using keys that are generated by the server and shared during the initial station setup. Tunnels may also be used to remote access the monitoring stations.

Access to the configuration web page on the stations and file transfer is performed through [SSH](#) tunnel with additional password authentication to the specific service or page.

Since the Linux operating system is used either on the central server and on the [CRFS](#) spectrum monitoring stations, the use of applications such as Nagios⁴ is a viable alternative to monitor the network operation using [SNMP](#) interfaces.

But the [SNMP](#) interface is not suitable for comprehensive network automation since write operation are not allowed due to device-specific implementations and security constraints that restricted [MIB](#) to read operations.

On the current configuration, any automation will depend on the existing [SSH](#) to directly access the system shell, [SFTP](#) or rsync⁵ to transfer configuration files, and provided web services to interact with the spectrum monitoring station daemons.

Even though we are here considering the specific case of the [CRFS](#) equipment deployed at [Anatel](#), a high-level evaluation of other systems available at [Anatel](#) indicates

⁴ <https://www.nagios.org/>

⁵ <https://rsync.samba.org/>

that most of the monitoring stations operate under similar conditions and could be controlled by remote access using similar topology directly or by a controller interface.

When confronting the presented features with the user requirements related to a more autonomous operation and maintenance of the monitoring stations, one may consider the use of an [SSH](#)-based automation system such as Ansible⁶ to provide this service. The use of this kind of automation system may allow the implementation of better security protocols and thus improve the overall network reliability.

Considering that the complete system is working under a [VPN](#), a first release could be made available with little control over users since only previously authenticated machines will have access to the server.

An authentication method is required to perform [CRUD](#) operations, an action that will be required to include specific measurement requests or to annotate data prior to the generation of reports. Considering the multi-user environment of teams working on a report using the same dataset, it also would be important to identify user actions to facilitate collaboration.

One must take note that the chosen framework to build the web GUI to the system should include user authentication features, if possible, with [SSO](#) and [LDAP](#) features, compatible with [Anatel](#) intranet and most of the modern systems.

3.3.5 Interoperability

As a reference for the interoperability criteria, in order to be aligned with [Anatel](#)'s policy, it's going to be adopted the reference of the Brazilian Interoperability Standard defined as [ePING](#) [28].

This interoperability standard defines 5 major policy drivers that must be considered on the system definition:

1. Preferential adoption of open standards, proprietary standards should be the exception;
2. Use of public and open software should be preferential;
3. Transparency is the norm. Systems should allow the public distribution of data. Confidentiality should be the exception;
4. Security should be compatible with the application;
5. Use of standards accepted by the market to reduce costs.

These policies were already discussed in this chapter and had their relative importance to the current project highlighted in relation to other requirements. The definition by the mentioned interoperability standard comes to support these previous discussions.

⁶ <https://www.ansible.com/>



The referred interoperability standard also considers three dimensions for interoperability, the technical, the semantic and the organizational.

Within the technical dimension, it is highlighted the importance of increasing access to public datasets and scalability issues associated with the system requirements.

Within the semantic dimension, is considered the importance of developing ontologies to the datasets, the adoption of open standards for data modelling and publication, allowing for the mentioned transparency and public access to the data.

Finally, within the organizational dimension, it is highlighted the importance of simplifying the interactions between government and society, the promotion of collaboration and the guarantee of privacy.

The [ePING](#) standard also defines several formats and protocols as recommended or as adopted by the government. This list of standard interfaces should be considered when describing the new system.

3.3.6 Usability and availability

[Anatel](#)'s spectrum monitoring activities are mostly decentralized, something that makes sense to most spectrum management organizations since several regular activities demand action from local field teams, e.g. those related with the solution of radio interference cases.

Even on more centralized organizations, is desirable that the spectrum monitoring data should be made available to distinct groups within the organization, including not only those assigned with tasks related to the spectrum monitoring itself but also departments assigned to other spectrum management tasks, including licensing, spectrum engineering and regulatory enforcement.

Apart from the spectrum management authority, one may realize that the spectrum monitoring data is a public resource that must be made available to all potentially interested parties, including for example universities and research institutes that may reuse the data to conduct investigations regarding topics of general public interest.

All these applications highlight the importance of creating a system that may be easily accessible, if possible, on a device-independent platform, for example as a web application.

It also implies that the system should allow for easy integration of different applications for data analysis, in line with the interoperability requirements already discussed, i.e., should employ an open standard compatible with multiple platforms.

Concerning usability, one may consider that the management of spectrum monitoring data is not different from many other data-intensive applications. It is expected that the system interface should allow for [CRUD](#) operations over the dataset and all classical information retrieval alternatives. Examples that may serve as references for usability will be discussed in the section about the “State of the Art”. More advanced query methods, such as using classifiers and automatic tagging of the data should be explored when available.

One important aspect is that data should be integrated seamlessly, with a focus on the measurement content, regardless of any data segmentation created by the measurement and data transmission system.

3.3.7 Platform and IT environment

[Anatel](#)'s web servers operate with Linux distribution to which there is an enterprise-level support contract in place. Considering the requirement previously discussed, that points in the direction of a web platform for the solution interface, the new system should also be constructed around an enterprise-level Linux distribution.

The use of Linux is also in line with the open architecture requirement discussed on the “Interoperability” section.

3.3.8 Efficiency and Scalability

As discussed in section 3.2.1, considering just the core of transportable monitoring stations at [Anatel](#), each station could produce about 4.35GB of data each day and a network composed by 184 stations, operating on various levels and roles, could easily produce more than 500GB of data per day.

The proposed system should be able to process all incoming data and thus, aspects such as efficiency and scalability become paramount and should be considered on the technologies to be used.

4 State of the Art

This chapter provides a brief review of the market alternatives (4.1.1) and of the literature relevant to the project in different stages of the data processing workflow (4.1.2, 4.1.3, 4.1.4 and 4.1.5). The chapter ends with a discussion on key aspects of the reviewed literature and products (4.2).

4.1 Technological context review

In order to translate the system requirements presented on the “Problem Analysis” section into a viable proposition, one must review existing market alternatives and available technologies that might already provide the desired service or may serve as components into the integration of the desired information management system.

4.1.1 Market survey

The present work has no intention of benchmarking the existing solutions, but a review of the market alternatives was essential not only to better understand the common traits on the available solutions but also to evaluate how much the available solutions were able to relate to the expectations for the current project.

Companies were inquired by email and video conferencing. An example of a message template used to conduct the initial approach and inquiry is presented in annex 10.2. No extensive qualitative study was conducted on these inquiries, something considered out of the scope of the current project.

Contact was made with 7 different solution providers. These were selected due to their regular participation in the ITU spectrum monitoring and spectrum management working party and/or due to their participation in spectrum monitoring and spectrum management related projects carried out by the Brazilian administration. The inquired companies were:

- ATDI⁷
- CRFS⁸
- GEW⁹
- LSTelecom¹⁰
- Narda - L3Harris¹¹

⁷ <http://www.atdi.com/spectrum-engineering/?lang=en#>

⁸ <https://www.crfs.com/>

⁹ <http://www.gew.co.za/spectrum-monitoring/>

¹⁰ <https://www.lstelcom.com/en/home/>

¹¹ <https://www.narda-sts.com/en/>



- Rohde&Schwarz¹²
- TCI¹³

From these companies, two are more dedicated to software solutions, integrating their application with third party measurement equipment and the remaining are more focused on the measurement equipment manufacture and associated software.

In different degrees, all these manufacturers provide some form of data framework where spectrum monitoring measurement data may be indexed by its essential acquisition parameters, e.g. location, time and frequency. Most of the solutions also include the possibility of indexing the data by parameters such as user-defined tags, equipment used, the technician responsible, etc.

The reviewed solutions often include some interface to spectrum management data, which allows for the information retrieval based on the allocation or licensing information. The license information might need to be manually associated with the emission, and the automation level available on each solution may vary.

Figure 6 presents a mock-up of such a system interface for illustration purpose. In this illustration, the areas related to the query parameters are highlighted with distinct colours and the query result is presented on the lower left side.

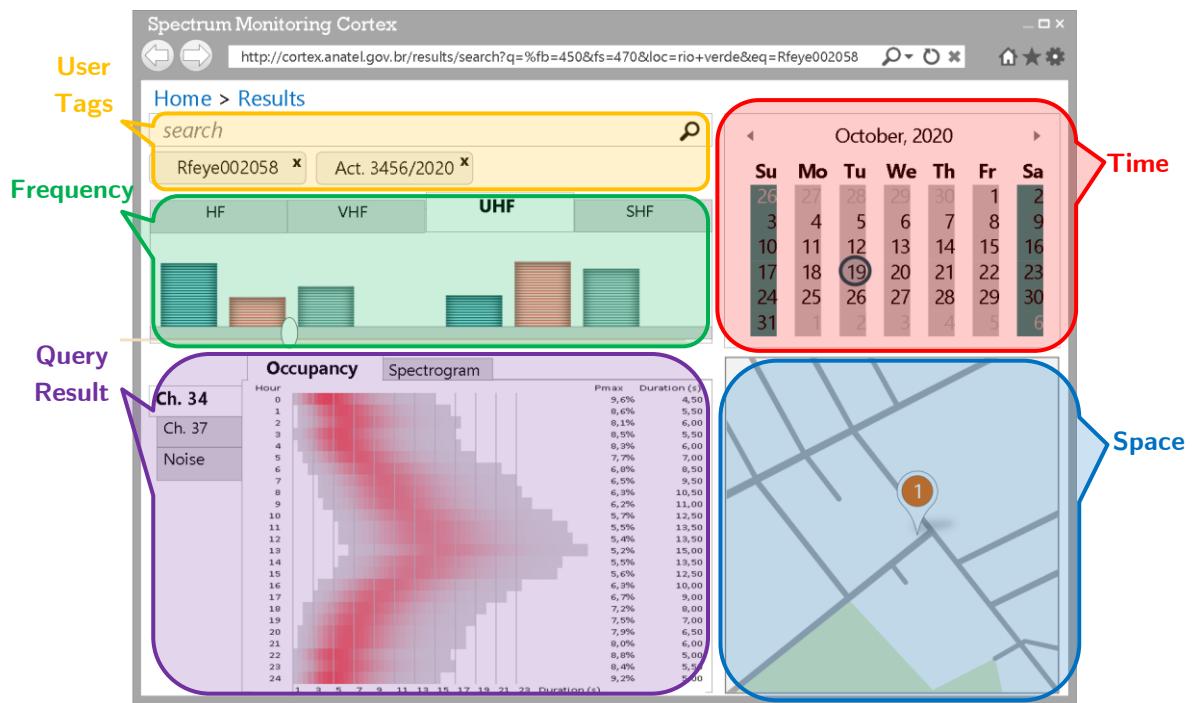


Figure 6. Spectrum monitoring data management GUI mock-up.

An important reference on this subject also comes from ITU, in the form of reports and recommendations specifically dealing with the subject of spectrum monitoring evolution [29], [30]. Such documents, as the consensus product of the spectrum

¹² https://www.rohde-schwarz.com/products/aerospace-defense-security/spectrum-monitoring/pg_overview_63720.html

¹³ <https://www.tcibr.com/>

monitoring community participating at the ITU forums, including manufacturers and regulatory authorities, provide a good overview of the understanding and expectations of this community.

Some companies also provide white papers that may hint on future development and products. On this line, one must highlight the work on the application of machine learning to spectrum monitoring as reported by Knott from [CRFS](#) [31]. This paper provides an interesting application of neural network to provide signal detection and classification based on the IQ waveform data.

Considering in more detail the different solutions on how the measurement data is managed, a recurrent aspect is that the use of DBMS is restricted to indexing, while the data itself is kept as files or, at most, as byte streams encompassing entire frequency bands into a single object. One of the companies were even working on a specialized file system architecture dedicated spectrum monitoring data, in an attempt to improve the access time and storage efficiency.

4.1.2 Information management system

Contemplating the idea of building an information management system, we organize the present section following the content management model defined by Bobko [32]. It's important to highlight that when we apply the concept of a content management system to organize the spectrum monitoring data, we are using it in a more encompassing form than a web publishing tool.

In fact, the concept of digital asset management system, such as described by Keathley [33] might be considered more appropriate for the current case, but still, bearing in mind the distributed nature of the proposed system as discussed in the user stories, the intended system is most likely to become a web-enabled platform much similar to a content management system.

For the present application, the standard model from Bobko [32] may be applied and represent a simple and effective alternative, avoiding the mentioned more sophisticated models used for asset management.

The model defined by Bobko [32] divides the data workflow into three stages: Collect, Manage and Publish, each with several tasks that may be employed in different scenarios. The following subsections review the state of the art using these stages and tasks as a reference, providing further developments on the problem analysis and advancing in the direction of an objective proposition.

A reduced version of this model from Bobko [32] is presented in Figure 7.

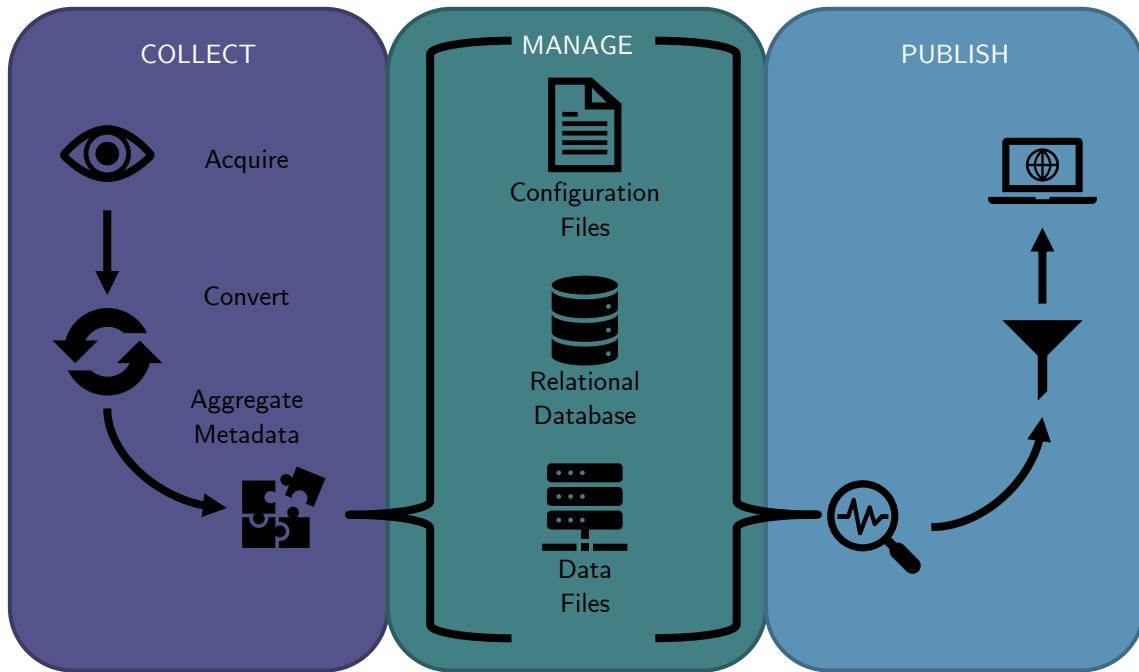


Figure 7. A simplified model for data workflow. From [32] with edition.

4.1.3 Data acquisition

Data acquisition is the first task on the “Collect” stage of Bobko [32] model. On this project, we consider it a fully automated process that generates the “.BIN” files containing the measurement results produced by the spectrum monitoring stations and later transferred to a central server for storage and processing.

For a more detailed view of this stage, we refer to the discussion presented in the chapter: “Problem Analysis”, in the subsection: “Data Source”. To a more encompassing understanding of the data acquisition purposes, it is recommended to review the introductory notes presented on this document about the role of spectrum monitoring and strategies used to acquire information about the spectrum usage. As a seminal reference on the subject, one should refer to ITU publications, especially the Spectrum Monitoring Handbook [14], freely available in electronic format at the ITU website.

One important aspect of the data acquisition on the present case is that it concerns measurements of physical units. It is beyond the scope of the present work to perform a metrological analysis of the spectrum monitoring procedures but is paramount that the information system designed as a repository to measurement data is aligned with the best standards in order to ensure the integrity of the information.

Speaking about measurements standards one must speak about the *Bureau International des Poids et Mesures*¹⁴, BIPM, which is the recognized international organization with a mission to promote the measurement science, the creation and adoption of measurement standards.

¹⁴ <https://www.bipm.org/en/about-us/>

The primal references on measurement vocabulary are provided by the BIPM [34], from where one can find a series of definitions essential to understand the concepts of measurement, traceability and uncertainty. For a better understanding of this last concept, it is worth mentioning other publication from the BIPM, the “Guide to the expression of uncertainty in measurement” [35].

From these concepts, one may summarize that a system designed to store measurement results must not only store the quantities but also the essential information to its interpretation, which means the measurement units and the information needed to provide the traceability to internationally recognized references, in short, identifying who is responsible for the measurement, using which equipment and procedure, and on which conditions.

4.1.4 Data conversion

Data conversion is the second task on the “Collect” stage on Bobko [32] model. It plays an essential role in the current project, since, as discussed in the “Problem Analysis” chapter, [Anatel](#)’s monitoring network is composed of equipment from several different manufacturers and each manufacturer employs a different data storage format. This situation also applies to most of the spectrum management authorities.

To enable the seamless integration of data from several different data sources into a single analytical platform, one must consider the creation of a single standard for data exchange and storage of the spectrum monitoring data, to which, individual conversion modules would communicate and thus simplify the use of the analytical functions.

On this project, we are not going to discuss specific implementations of each manufacturer and suffice to say that most of them are documented and accessible to [Anatel](#) such as that the decoding of the data is not an issue. More specifically the files from [CRFS](#) are thoroughly described in the system documentation [20], [21].

Considering the amount of data generated by the spectrum monitoring network, that can easily achieve hundreds of gigabytes per day, it is essential that the choice of the format allows for such a large scale. When faced with data that may be represented on tables with hundreds of thousands of columns and millions of rows of 6 digit, floating-point numbers, it is not conceivable to store single power measurement for each frequency bin, using a relational database or text-based formats such as CSV, XML or JSON.

One may consider that it is possible to serialize the data into a binary stream and even store it as an object within the XML or JSON file or within a relational database. But such alternatives are not efficient when considering the data manipulations required to perform any data retrieval since it would demand that the data complete dataset to be deserialized and decoded before processing.

Looking for solutions that are able to handle large datasets one may find numerous open digital formats that have appeared on different fields to [36]–[40], encompassing areas from physics to medicine but in 2018 came about one important reference that tip the choice towards a specific format. This was a recommendation by ITU concerning a digital format to be used for the exchange of I/Q data [41].



As discussed, I/Q data is the raw waveform representation of a segment of the radio spectrum and is the most complete and data-intensive alternative sometimes used on spectrum monitoring activities. The mentioned ITU recommendation ITU-R SM.2117 [41] employs the HDF5 format to store the I/Q data.

HDF5 is supported by “The HDF Group”, a non-profit organization that provides support to the API that enables the use of HDF5 on C, C++, Fortran and Java. The available APIs are fully documented and there are various source code examples to facilitate the use of the format.

Other open projects and commercial products have used this API to create fully-functional HDF5 interfaces on Python, R, Matlab, Labview, IDL and even geospatial applications such as QGIS and ArcGIS [42].

Apart from the APIs, “The HDF Group” also provide support to several tools that enable the visualization and data manipulation of HDF5 files within a single system and over a network environment using cloud platforms.

Concerning the file organization, data is stored within the HDF5 file into objects that may be datasets or attributes. Datasets are multidimensional arrays. Each element of the array must be of the same type, e.g. int, float, string or even a compound datatype made of these atomic elements. Attributes may be arrays or single-valued. Again, the attribute must be of a specific type, much in the same way as the datasets.

There are a few differences between datasets and attributes, for example, attributes may be associated as metadata to a dataset, but not the other way around. Most importantly, datasets have functions that allow for efficient use of storage and access, such as chunking, data compression, partial I/O and sub-setting, which means that datasets are better equipped to handle large data tables while attributes do not and may be used only for associated metadata.

One important characteristic is that both objects, datasets and attributes, must be uniquely identified by a path and a name within an HDF5 file. To understand the concept of “path” one must realize that the HDF format is hierarchically structured much in the same way as folders within a file system.

Here comes in place another crucial element of the HDF5 structure, “Groups”. Groups are like folders and may contain datasets and attributes. All HDF5 files have at least the root folder, from where multiple groups, datasets and attributes may be created.

The HDF5 format has another interesting feature that may aid into the data optimization: One may create logical links, either internal, within a single file and external, between multiple files. These links operate much in the same way as symbolic links on Linux file systems and may be used to avoid data replication, e.g. have a single frequency axis array to index multiple datasets on various groups.

Figure 8 represents a simple example of how HDF5 can be structured using different objects to represent a complex set of data, including diverse types and structures.

Object Name	Object Type	Data Type	Object Value Example
/	Root		
File format	Attribute	Var Len String	"IKT.2515"
Temperature	Dataset	Float array	[18.5, 19.3, 19.8....]
Measurement Unit	Attribute	String	Celsius
Median	Attribute	Float	
Timestamp	Dataset	64bit Int array	[1882643285, ...]
Site	Attribute	Compound {Lat; Long; Alt}	
Audio report	Group		
Audio	Dataset	Byte array	[f&n@h6Hwg(\$%\$....]
Audio Format	Attribute	Var Len String	"MP3"

Figure 8. Diagram representing HDF5 file format structure

As the most important references to the current project, we may mention the C++ examples and libraries available on the [HDF Group website](#)¹⁵ and the Python API reference from Andrew Collette [43].

Defining the use of the HDF5 format is a crucial step into the definition of how the digital assets will be managed, but it opens a door to another issue. Which ontology should be used?

Although one could consider that the definition of an ontology is an overkill approach to the problem, it is important to remember that the spectrum monitoring data is used within a broader context of spectrum management, that may be described as a domain with classes, attributes, relations and events within which the spectrum monitoring information is just a subset.

To understand the relation between the HDF5 format and ontologies is important to remember that the HDF5 format only specifies how the data shall be organized in terms of basic types, objects and relations. It does not define the names and which attributes must be present for any application. In fact, the only restriction is that the same name must not be used twice for objects within the same group or for attributes associated with the same dataset or group.

Going back to the ITU references, there is a set of attributes and naming conventions defined by the standard for I/Q data at ITU-R SM.2117 [41], but these dataset names and attributes are not sufficient to cover all the semantic scope associated with other spectrum monitoring data, more specifically in our case, concerning trace data and occupancy data.

Fortunately, to some extent, ITU also produced other recommendations concerning these other types of spectrum monitoring data [44]–[46]. Although these formats are

¹⁵ <https://portal.hdfgroup.org/display/support/Documentation>

designed for text-based data exchange, the attributes and naming conventions described may be used to create an equivalent representation using HDF5 format.

It is important to highlight that the text-based exchange formats described on the mentioned recommendations were created much before the development of modern acquisition systems, as early as 1990 in the case of the first edition of the ITU-R SM.668 [45]. The reduced amount of generated data allowed for a compromise of the machine processing efficiency in favour of the human readability, a situation that no longer holds true.

Concerning the use of ontologies within ITU, one can find a few references dealing with M2M and cybersecurity topics [47], [48]. These may be used as a reference for an eventual discussion on ITU forums about the creation of a spectrum management ontology, otherwise, they are not helpful references to the current project.

Broadening our scope of research into academic references, one may find several articles and projects that tackle different subsets of the problem, from the broadly accepted definition of ontologies for measurements [49] to more specific and less consensual ontologies applicable to wireless systems [50]–[54]. Supporting references may also be found that employ some of the proposed ontologies or variations around them to create several specific applications [55]–[57].

4.1.5 Metadata aggregation

The metadata aggregation is the third and last task on the “Collect” stage on Bobko [32] model.

From the previous steps, metadata information related to the data acquisition and conversion have already been aggregated to the data, e.g. where it was collected, by which equipment, who was the technician responsible, etc.

Still, to allow the new system to answer the requirements depicted by the user stories, providing an evolution of the spectrum monitoring services as described in section 2.4, it must include improved analytics to enrich the metadata information.

Considering this perspective of needed evolution to the spectrum monitoring services, one of the initial inspiration to the present project came from the work by Shi, Bahl and Katabi [58], where the authors present a solution that is intended to optimize the balance between the need of a detailed evaluation of each band (long acquisition time at a specific frequency) and the need of scanning a wide range in the spectrum (short acquisition time at each frequency).

A key aspect of the solution proposed in that article is that it applies a learning algorithm to store patterns about how the spectrum is used. These patterns are in turn used to optimize the acquisition process to maximize the probability of intercepting all emissions on the operational frequency range of the measurement equipment, allowing for larger acquisition time when it is needed and shorter acquisition time when it is not.

The idea of creating a pattern databank and optimizing the acquisition in tune with the learned patterns is crucial to the current project and answer to two of the presented requirements.

Firstly, the idea of a fully automated system that can perform the optimum data acquisition timing setup with minimum human input.

Secondly, the association of emissions to a finite number of patterns, such as described in the mentioned article, allows for new alternatives of indexing and retrieving the measurement data based on such patterns.

4.1.5.1 System Automation

Further developing the idea of system automation, one must consider the spectrum monitoring workflow as described in section 2.3.

It is possible to imagine that the tasks associated with the analysis may be aided by the signal clustering/classification and the associated metadata enrichment, but the measurement setup may be the activity to take the most advantage of automation by some of the ideas previously discussed.

Shi, Bahl and Katabi [58] point to the automation of the timing parameters but in some conditions the technician responsible for the measurement must also configure the spectrum sweep parameters, i.e. the FFT bin size, or equivalent RBW filter on traditional spectrum monitoring equipment and the step between adjacent bins (on some specific measurement systems).

Most of this setup was not used by Shi, Bahl and Katabi [58] due to reasons discussed later on the section about the “Critics to the state of the art”, but one important issue is that the information about the frequency resolution is critical to allow for the automatic channel segmentation, an important feature if one considers the task of analysing an unknown radio frequency environment.

One may say in fact that the evaluation of an unknown environment is a more common situation than the orderly spectrum scenario. Often the radio spectrum is bugged by unknown sources of interference or one may find a band shared by multiple services that operate with different channel [allocation](#) schemes, resulting in a myriad of unpredictable spectrum occupation scenarios.

To overcome the problem of detecting an emission, i.e. differentiate a signal of interest from the background noise, several methods have been proposed and have ample review on the literature [59], [60]. More recently, approaches using neural networks and deep learning have also been used with increasing success [61]–[63].

Many of the described alternatives for detection, or combinations of these, may be used to perform the task of segmenting the spectrum into channels, a critical step if one does not want to depend or simply to rely on knowledge about the channel [allocation](#).



For the current project, the selection of the detector is driven by the following arguments:

1. The absence of a known signal database restricts the ability to use neural network and deep learning approaches, although this may be considered the most promising alternative in terms of algorithmic efficiency and efficacy.
2. The use of the raw waveform data (I/Q) is limited because it is too demanding on data storage and transmission resources. This alternative would also be more demanding on signal processing algorithms, extrapolating the scope of this initial system development, that needs to deliver functionalities at the lowest possible cost.

When discarded the use of I/Q data, it becomes impractical to use methods based on covariance, autocorrelation or cyclostationary properties of the signals. The use of wavelet-based detection and matched filters may also be discarded for similar reasons.

3. The use of simple trace data is desirable due to its applicability to simpler systems, such as spectrum analysers, that may be used as an acquisition tool into the proposed information system at a lower cost than by acquiring new wideband digital receivers and digitizers.

When considered all reviewed detectors and the exposed arguments, it was decided to concentrate the efforts on the energy-based detectors.

The use of energy-based detectors usually requires the definition of a noise reference level, which returns the issue back to the original question about the need to characterize the noise and differentiate it from signals.

Fortunately, there are some approaches that propose to solve this problem by automatically characterizing the noise. On [60] this alternative is presented as “information theoretic criteria-based detection”, and in short computes the statistical properties of each sub-band and compare it with the expected distribution of the gaussian noise. A similar approach is described in more detail as a Bayesian Detector by Adams and MacKay [64].

Using such a detector, a system should be able to detect emissions based solely on the trace information, without previous information about the spectrum noise level or channel [allocation](#). Once channels are created by a detector, it is possible to employ time optimization algorithms to configure the remaining spectrum sweep parameters and thus perform the autonomous configuration as intended on the current project.

Furthermore, once channel detection has been performed, minimum requirements in terms of bin width may be enforced onto the optimization algorithm to allow for clear channel separation.

One interesting aspect about the use of Bayesian detectors is that it does not require training and the results obtained can be easily understood by the system users, avoiding some of the limitations of black-box alternatives, such as seem on the use of neural network algorithms.

4.1.5.2 Clustering and Analysis

Going back to the ideas of Shi, Bahl and Katabi [58] concerning the pattern clustering, it is easy to envision that such patterns could be used as additional metadata to allow richer indexing and analysis of the spectrum information.

The approach used in the mentioned article is based on the I/Q waveform data, something already discarded for the current project as previously discussed and although some effort could be made to adapt the implemented algorithm to work with trace data, let's take a step back and explore alternatives for clustering of this type of data.

The channel power variation over time is a problem that can be approached with traditional time series analysis tools. The book from Aileen Nielsen [65], in early release when this text was written, provides an encompassing view of the time series analysis problem. Further research on the topic leads to the additional references [66]–[69] on how to characterize distances and perform clustering of time series segments.

Keogh and Mueen from the University of California proposed an interesting approach that was branded Matrix Profile. A series of papers describes this time series data mining strategy. For clustering purposes, one may refer to [68], [69] and the Euclidean distance metric called MPdist. This measure and the algorithms proposed are available on several platforms, including Matlab, Python, R and Go.

Using such measure over the trace segment corresponding to a single channel, it is expected that one should be able to pinpoint changes in the channel usage conditions, such as a variation on the occupied bandwidth or the modulation parameters. These changes may be the result of changes in the emitter or the occurrence of interferences.

Apart from the distance within each channel, one could also use similar approach to compute the distance between different channels, enabling for application of clustering algorithms that may allow for the identification of different emission patterns, such as due to variations on the modulation and bandwidth, e.g. differentiate an FM stereo transmission from a broadcast station and an FM mono transmission from a wireless microphone.

The idea of clustering based on the distance computed between traces has the interesting potential to be more meaningful than a distance computed between descriptive parameters extracted from the trace, such as occupied bandwidth and power at specific relative levels. The main difference is that in the case of the distance between traces the entire information contained on the spectrum trace is considered. Within this additional information that describes aspects such as how is the slope as the power rises and falls within the channel band, and how flat is the peak.

There are some aspects that must be highlighted on the comparison of two arrays of spectrum data points:

- Firstly, the relative index of each point within each channel is relevant since it is associated with the actual frequency characteristic of the emission. Thus, it is not desirable to use elastic measures such as DTW, Edit distance or Longest Common Subsequence to measure the similarity of the vectors;

- Secondly, the absolute power level of each bin is not relevant, since propagation effects and reception characteristics will affect it. Thus, two similar emissions, or even the same emission, may present significantly different absolute power levels on the same bins when detected from different sites or with different equipment setup, but still, the power level difference between bins will be maintained;
- Thirdly, the noise level may change according to the equipment setup or operational environment. Such change may hide some of the emission characteristics and thus must also not be considered.

Another important analytical use of the clustering results is that such patterns may be extrapolated into a statistical model of the transmission events, such as the probability density function that describes the activity on each channel [70].

Such statistical representation of occupancy may be considered a relevant improvement over the traditional form of representation, based on the percentage of active time over a fixed window interval. This more detailed representation may allow the use of real data on analytical tools, such as SEAMCAT [71], an open-source solution created by CEPT to assess interference levels between telecommunication systems using “Monte Carlo” method.

Another seminal article from Marko et al. [72] presents a similar approach, combining the occupancy measurements with simulation tools to produce interference maps that may be employed to provide meaningful data about the spectrum usage over large areas.

4.1.6 Manage

After the data is processed, comes into place the second stage on Bobko [32] model, that is managing the data repository.

Considering the use of the HDF5 format, all metadata is embedded in the data files that could be handled using only the HDF structure and indexing mechanism. This approach has some drawbacks and such extensive use of HDF5 functionalities have been criticized in the academic literature and by users on various forums [38], [73]–[75].

Taking such references, a more stable approach would be to use the HDF5 metadata, internal and external links as the means to store the measurement information only for transmission and backup. For information retrieval purposes, the use of a standard database system would be more effective.

With this, the choice of a database system to index the HDF5 binary files is more influenced by the publishing mechanism than by the data format.

The database structure should then be created around the indexed HDF5 files, using corresponding attributes and also add metadata from reference information associated with the assignment, allocation and allotment.

4.1.7 Publish

The third and last stage on Bobko [32] model is to publish the data.

The basic publication interface may follow in line with existing applications, such as presented in the section about the “Market survey” but the availability of signal clustering and classification, more detailed channel usage information and alarms, may enable a more engaging and meaningful integration of the data into the GUI.

This encompassing approach means that the interface should include more than the query fields needed for information retrieval, it must include controls that allow for the user interaction through data annotation and the report generation.

Data annotation may play a vital role, since it may be used to ratify/rectify automatic clustering results and create a classification database, allowing for later learning algorithms to improve and automatically classify detected emissions into meaningful groups.

Numerous frameworks are available to create an interface, including desktop and web alternatives. Such alternatives will be discussed in the chapter about the “Proposed Solution”.

4.2 Critics to the state of the art

When reviewing the ITU references about spectrum monitoring evolution [29], [30] and even some of the products available, it becomes clear that little attention is paid to the information management problem. In fact, such a problem is only hinted on the 2013 recommendation on the subject [29]. Most of the concern displayed on these documents are related to measurement and signal processing strategies and the data management problem left open.

In part, this limitation on the referents may be due to their release date and the enormous evolution on the recent years concerning big data, analytics and machine learning.

On the prospect of creating an open information management system for spectrum monitoring, this lack of interest create an opportunity but also implies that many of the required foundations are missing, more importantly, a standard digital format for data exchange and ontologies associated with the spectrum monitoring and spectrum management domains.

Concerning the standard digital format, as previously discussed, the best alternative currently available is the use of the HDF5 format, although for its use one must be aware of the critics and highlighted restrictions [38], [73]–[75].

One main concern is the complexity of the HDF5 format, which makes it difficult to be used in real-time by embedded systems with low processing power. Such restriction may make it hard to adopt an HDF5 standard directly on the acquisition platform, but this may not be viewed as a relevant restriction if the format used by the measurement device is properly documented and suitable export/conversion tools are made available by the equipment manufacturer.



One of the reported main limitations of HDF5 is associated with the ability to perform parallel I/O [73]–[75]. The latest release, 1.10 as per early 2019, supports only SWMR (single-writer/multiple-reader) operation [76], a severe limitation considering the natural parallelism of the intended system, that will operate with multiple sensors collecting and uploading data in asynchronously and in parallel.

To avoid problems one interesting alternative is the use of HSDS (commercially available as Kita¹⁶), a successor of the HDF Server, that provide a more reliable data management solution for large numerical datasets.

Other limitation concerns stability and the possible data corruption when all information is stored into a single multi-gigabyte file, something that can easily be avoided by splitting the data into smaller files. This strategy also allows for easier maintenance and support of the system, although it might result in a reduction of storage use efficiency.

The proper use of the HDF5 format also requires the collective agreement on a domain ontology a problem more complex since its acceptance depends on the consensus of the relevant community. In the present case, such a community is represented on the spectrum management study group, SG1, at ITU.

Moving to the topic of metadata aggregation, a first highlight is that most of the reviewed algorithms employ the I/Q waveform data as input. This is the most complete representation of the signal and, as discussed, overtaxes transmissions and storage resources when compared with other representations such as trace data.

Another limitation is that I/Q waveform data is not available on most of the legacy spectrum monitoring equipment, at least not with the bandwidth and time durations required to generate meaningful results, especially regarding spectrum occupancy measurements.

Trying to provide an analytical platform based solely on the trace data address the mentioned restrictions, and although it most likely will not have the same efficacy as the I/Q waveform based methods, it still may represent an interesting and efficient alternative to enable the use of legacy systems and optimize storage and transmission usage.

As a reference of the implementation based on I/Q data, one may again refer the work by Shi, Bahl and Katabi [58]. Surely enough, when using an FFT spectrum monitoring equipment, the bin width in Hertz is numerically tied to the sample duration. e.g. to achieve a bin width of 1Hz, one must sample the signal for a period of at least 1 second.

This relation of time and frequency would suggest that the optimization proposed by Shi, Bahl and Katabi [58] is sufficient to set all essential measurement configuration parameters, but this is not true due to several reasons. One limitation is that the solution proposed in the mentioned article still employs the standard frequency allocation tables to perform the splicing of the spectrum into the channels that shall be analysed.

¹⁶ Trademark. <https://www.hdfgroup.org/solutions/hdf-kita/>

Most importantly, the time optimization proposed in the article is focused on the temporal behaviour pattern described by the signal, e.g. when it turns on and when it turns off, and may not be sensitive to features such as the channel separation or how the power level changes within the used bandwidth.

One extreme example that serves to illustrate this last aspect can be found on the [DTT](#) emissions. The channel [allocation](#) varies from country to country but is defined with 6MHz, 7MHz or 8MHz [14]. Systems like [DVB-T](#) and [ISDB-T](#) transmits the information employing a scheme known as [OFDM](#), where hundreds of individual narrowband emissions are arranged to transmit the information of a single channel.

Figure 9 illustrates this type of spectrum trace representing a scatter plot of the emission power as a function of the frequency, starting at the middle of channel 46, that is empty except for the presence of a secondary user, maybe a wireless microphone or similar device. After channel 46, five adjacent ISDB-T channels are present, primary users of this band, until half of the channel 52 that is empty. Colours represent the hit count at each specific power level, from dark blue for a small number of occurrences to red for a large number of occurrences.

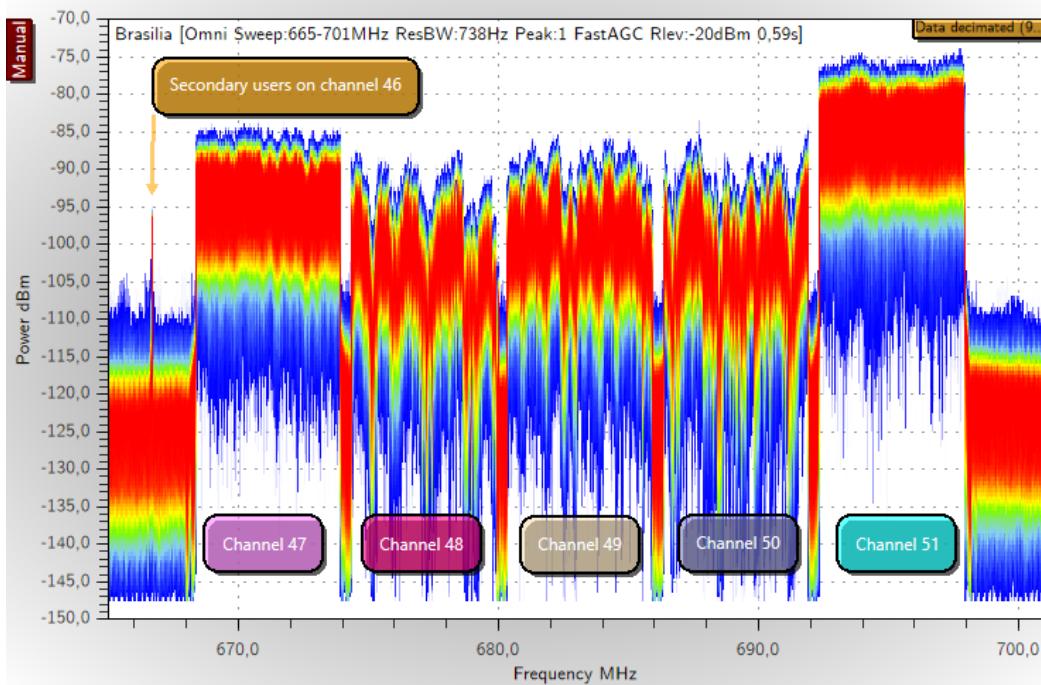


Figure 9. [DTT](#) channels 48 to 52 using [ISDB-T](#) transmission on adjacent channels.

This type of digital modulation, OFDM, gives to the spectrum trace a rectangular shape where the small individual emissions are impossible to discern using a standard, non-synchronous, measurement device such as the ones used on spectrum monitoring. More important, adjacent channels may appear close together because of this rectangular shape, with fast decay on the power level outside the used band.

On the [DTT](#) band, channels have 6 to 8MHz of bandwidth and are constantly present on-air. An optimization algorithm based solely on the time behaviour would indicate that an extremely fast acquisition rate, in the range of nanoseconds every few minutes, would suffice since it would be enough to detect the presence of a channel. Such acquisition

would result in a bin width in the range of hundreds of kHz and thus would not allow for the detection of the edge between channels, i.e. two adjacent channels would appear to be a single large channel, and most likely would miss opportunistic users of empty channels that might not use wideband modulation schemes such as employed by [DTT](#).

This effect is illustrated in Figure 10, where the same band presented in Figure 9 is now scanned with a bin width of 590.62kHz instead of the previously used 738Hz.

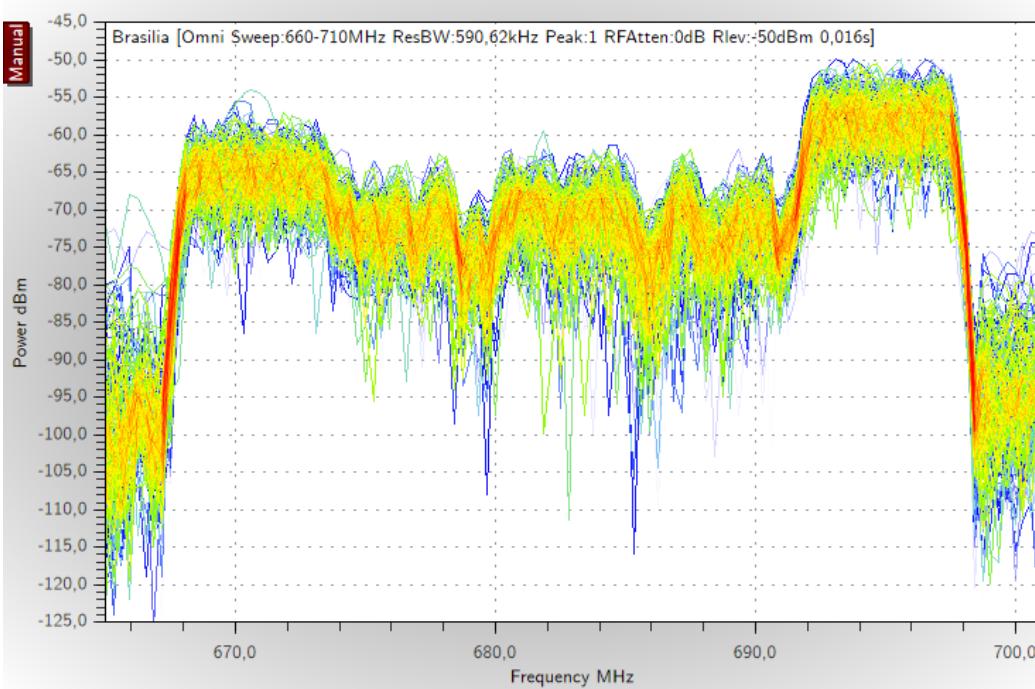


Figure 10. The same band displayed in Figure 9, scanned with frequency bins of 590.62kHz width

One important development that may be a game-changer to the above dilemma, between using I/Q waveform or the trace data, is the use of neural networks to perform detection and classification of the signals on the devices, such as presented in [31], [61], [77], [78].

Having a trained network may allow for an efficient implementation of a detector/classifier that will enable the use of I/Q waveform data on legacy devices to perform the analysis in real-time and store the annotated spectrum trace data, resulting in a more efficient use of the processing and storage capabilities of the entire spectrum monitoring network.

But the training of neural networks requires a lot of ground truth datasets that are not available right now for spectrum monitoring. Usually, the mentioned articles are based on datasets restricted on their content and/or created by simulations.

The idea of establishing a standard format and ontology to share spectrum monitoring data enabling the creation of open dataset repositories may be instrumental in solving this problem, even more if there is a supporting system that enables the systematic collection of real data and its annotation to create ground truth datasets for this domain.

5 Proposed Solution

After an overview of the problem requirements and the state of the art on the subject, one may build an objective proposition to the desired information management system architecture able to increase the analytic level of a spectrum monitoring measurement network.

This chapter presents the details of such proposition, firstly, in section 5.1, considering different approaches that may be used to solve the problem and performing an analysis of the risks associated with the selected approach on 5.2.

The solution design is described with increasing detail in the following sections, from a general view in section 5.3 to more detailed design on 5.4 and a discussion about the technologies to be used on 5.5.

At the end of the chapter, the architecture is reviewed to describe a smaller set of modules and more concrete technological alternatives to enable the creation of a pilot implementation as described in chapter 6.

5.1 Solution identification

We propose at this point a reflection about the possible means to achieve the described objectives.

5.1.1 Do nothing

This can be considered the default case, an existing alternative most of the time.

On the current project, doing nothing would be a bet on the future and the evolution of telecommunications and other services that may solve the problem by simply removing the need of spectrum monitoring or even spectrum management.

The scenario of using the spectrum without spectrum management may be considered not realistic because even alternatives that promote some level of self-organization of the spectrum, with a reduced regulatory framework, still recognize the importance of having a common framework in the form of protocols and reference databases to allow a universal access to the open space transmission medium. This sort of automated management system for the radio spectrum is more commonly referred to as “cognitive radio” and the literature is rich with proposals on this line [79]–[81].

In the same line, one may envision a future where no specialized spectrum monitoring network is necessary, for example, because all users would report to a common database for coordination, including measurement information about the usage conditions of the spectrum at their location [82]–[84]. Even so, the development of a management system using today’s infrastructure may enable the creation of standards and protocols that will lead to the creation of such a future. As such, the argument is more of a support to the execution of the current project than a deterrent.



5.1.2 Buy something now

A good alternative to solve a demand for a software system usually is to use one that already exists.

There are a few commercial products that are intended to solve the proposed problem. This topic is discussed in more detail on the session about the “State of the Art” but one can say that the evaluated commercial systems are in a process of evolution towards a goal similar to the one presented for the current project, at least in terms of incorporating the latest developments in the area of AI and machine learning, although there were not found actual commercial offers that are truly encompassing on this topic.

Unfortunately, most of the new systems are also designed to work with new instruments, rendering useless most of the investments done in the past few years on the acquisition of measurement equipment. Currently, the best alternatives also have a strong vendor lock, resulting in further difficulties for the spectrum management authorities.

5.1.3 Wait to buy something

We may also consider at this point the alternative of waiting for somebody else to do the work of developing a more encompassing solution and just buy it afterwards. This alternative is going to be explored in more detail in the chapter about the “State of the Art”. Peeking in advance on this topic, a critical issue is that there is no formal ontology or open format defined to enable the sharing of spectrum monitoring data in harmony with the open data principles.

Considering [Anatel](#) and other radio spectrum authorities as the consumers of future solutions for spectrum monitoring data management, it is essential to express the need for the creation of this open standards and, as such, a pilot system that implements a functional example of such a solution may be considered an important step forward, even if afterwards it is superseded by commercial applications.

5.1.4 Assemble from open software

Building a full system from scratch is a costly alternative that is simply not viable on any scenario unless one considers an application that may be sold to many or at a high price tag, none of which scenarios apply on the current case. One interesting alternative is to ensemble a workable solution from available systems that perform similar tasks.

[Anatel](#) had a previous experience adapting an open project management web application to control tasks and workflow for the regulatory enforcement teams. Although there is no public report released about that project, the authors' personal experience by following the initial planning stages and implementation steps demonstrated that such alternative can result in faster delivery of stable products at lower costs when compared with a complete system development cycle.

The idea behind this proposal is to maximize the reuse of existing open source applications, sometimes combining more than one application as a set of complementary service and building only the most essential parts as additional services or modules.

At this type of integration, different system parts may be viewed as independent services orchestrated into a microservice architecture. Development is concentrated only on essential services and additional functionalities can be easily expanded by adding new components.

This approach may demand adjustments on how the user requirements will be met, sometimes even challenging pre-defined concepts and expectations to optimize the delivery time and to promote the stability of the final application by maximizing the reuse of existing and proven solutions.

Closing this section, we understand that the alternative of assembling the desired solution using open source applications into a microservice template presents relevant advantages over the alternatives previously analysed. This approach will be followed as a guideline to conduct the present project.

5.2 Risk analysis

At this section, we conduct a brief risk analysis for the adoption of the proposed solution. To this end, we consider what we believe to be the most probable deterrents to the adoption of the proposed solution. To each risk, the impact and countermeasures are evaluated and may be used to guide the project and reduce the probability of occurrence, or the impact of such risk.

The evaluated risks are discussed in the table below. No formal assessment of the risk probability was performed due to the lack of user interaction at this stage that could allow for such an assessment. A more detailed evaluation may be conducted prior to the actual deployment of the first version to direct further development and the adoption of preventive actions.

Table 3. Risk, Impact and Countermeasures associated with the project

Risk	Impact and Description	Countermeasures
▪ Incompatibility with existing infrastructure	Conflict with existing support infrastructure, such as network security, server management, etc, may block the acceptance of a new system by Anatel IT department.	Ensure that the proposed solution employs frameworks that are compatible with the existing systems.
▪ Lack of support from the IT department	Anatel IT department may be restrictive to support applications that are not compatible with the existing environment. The lack of support may impede the deployment of the new solution.	Present a working solution that can supersede the restrictions to the initial adoption and is compatible with the existing environment. Work on promoting the new solution on the organizational level by promoting open forums and training.

Risk	Impact and Description	Countermeasures
<ul style="list-style-type: none"> ▪ Lack of developer availability 	<p>In the event that the group involved in the development is not able to continue the support until a stable deployment is achieved and considering the lack of specific knowledge on signal processing and spectrum monitoring by other developers, the proposed system may never achieve maturity.</p>	<p>The importance of continued development until an acceptable maturity level is achieved must be strongly promoted through organizational levels right from the first deployment.</p>
<ul style="list-style-type: none"> ▪ Lack of system maintenance 	<p>For the long-term survival, the system must undergo regular maintenance cycles, which may be difficult considering the use within a single organization.</p>	<p>The promotion of the system as an open-source solution to spectrum monitoring data management may allow for the adoption by multiple organizations, allowing for the sharing of maintenance costs and ensuring resources for long term system sustainability.</p>
<ul style="list-style-type: none"> ▪ Lack of acceptance by users 	<p>The solution may not be accepted by users as part of their routine tasks. The lack of adoption will result in failure of the project.</p>	<p>Include the users as soon as possible on the development and prioritize the generation of value to the user.</p>
<ul style="list-style-type: none"> ▪ Lack of value to the business 	<p>Even if the user is happy with the interface and results, the solution may not generate value in terms of more efficiency and productivity resulting in later neglect of the system support.</p>	<p>The solution should include features that are invaluable to the business and that may be hard to obtain by other alternatives.</p>
<ul style="list-style-type: none"> ▪ Excessive complexity 	<p>A bold proposition may not be realizable in a reasonable time due to complexity, resulting in poor initial deliveries and failure due to the lack of adoption and support to this initial solution.</p>	<p>Apply agile methodology and prioritize deliveries to maximize value to the user.</p>

Risk	Impact and Description	Countermeasures
<ul style="list-style-type: none"> ▪ Incompatibility with the development policies and commercial contracts 	<p>IT within Anatel is strongly supported by service contracts with external developers and the use of off-the-shelf applications. This scenario is not favourable to the use of inhouse human resources for software development.</p> <p>Although developers available through existing contracts, usually they are not expected to handle complex signal processing algorithms and as such, not suitable to handle a project with the complexity described.</p>	<p>Present a functional result as soon as possible to promote the feasibility of the current project and work later to enable the project development until maturity with the existing team.</p> <p>Promote the community use of the system as an open-source application to reduce the load on internal resources.</p> <p>Work with IT personnel to develop other human resources that may be aggregated to the project.</p>
<ul style="list-style-type: none"> ▪ Licensing and patent conflicts with manufacturers 	<p>Manufactures may question the legality of the use of available interfaces by the developed system.</p>	<p>Keep the system architecture modular such as to reduce the impact of individual conflicts.</p> <p>Promote the development of a standard open format to be used to store spectrum monitoring data. Such development would enable the easier integration of any measurement equipment later acquired into the spectrum monitoring data management system.</p>

5.3 Solution design

The following view presents the general description of the proposed system architecture. Icons and numbers are used to identify different elements described in more detail later in this section.

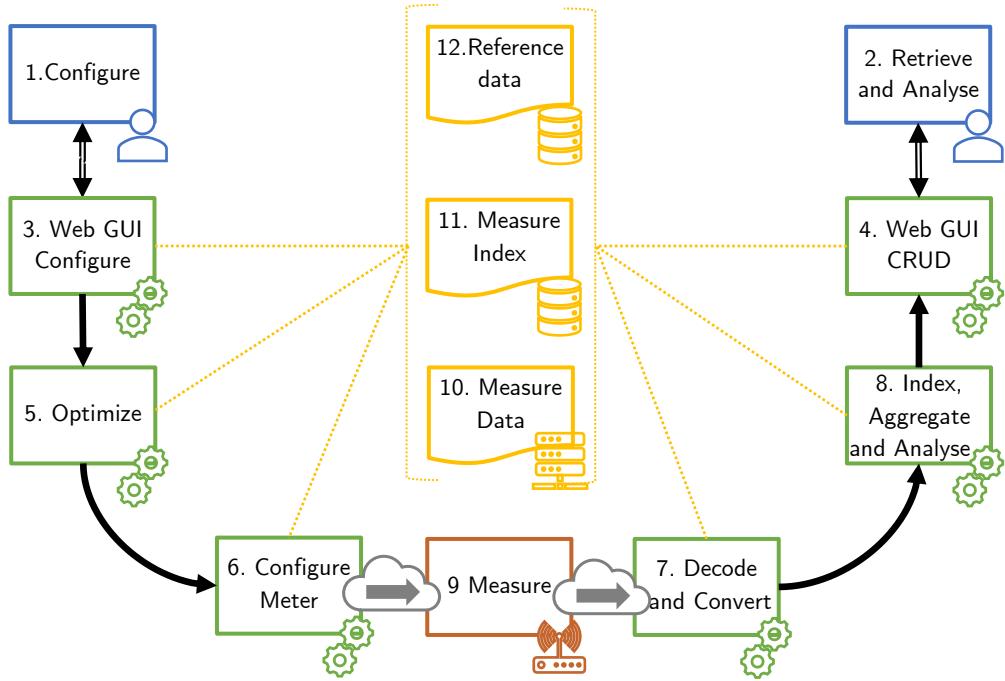


Figure 11. General view of the system architecture

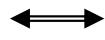


General operation flow

Simple arrows are used to indicate the process flow on this view.

It is important to highlight that all data should be shared through the corresponding data repositories presented at the diagram centre. The arrows do not correspond to any direct interface between modules but only to the regular execution sequence.

Ideally, the modules should be as independent as possible, e.g. operating as microservices over the dataset with activation triggered by events, applying a reactive architecture.



User interaction

User interaction is represented by the double line arrow and is bidirectional in nature, made through the [Web GUI](#) modules



Remote transfer

The arrow within a cloud represents the data and operational connection from the system to the external measurement equipment.

On the specific case of the measurement equipment used by [Anatel](#), the spectrum monitoring stations are scattered over the internet, using diverse connection infrastructure including dedicated data links, ADSL, 3G modems, etc.

The transfer of different files for configuration of the measurement equipment and for backup of measurement results are performed through VPN (virtual private network) established by each node to a central server. Other secure connection types, such as [SSH](#) might be configured and used.



Data connection

The dotted light lines are used to represent the data interface between modules and the repositories, which will store not only the results but also all the configuration parameters needed for the correct system operation.



User

The user may interact with the system using the [Web GUI](#).

1. Initial tasks include the **system configuration** that will bound the measurement parameters such as to allow the achievement of different spectrum monitoring objectives.
2. When measurement data become available, the user may access the [Web GUI](#) to **retrieve** the measurement data indexed on the system, performing **analysis** and [CRUD](#) operations that, in turn, may affect the automated process, e.g. associating tags to specific emission types.



System module

3. The [Web GUI](#) provides the main front end to the **system configuration**, although files may be used on initial releases or to specific purposes and improve interoperability, e.g. using an OWL file to describe the ontology used in the data repository.
4. The [Web GUI](#) also is the front end to all data repositories and provide the [CRUD](#) operations for information retrieval and manipulation functionalities to the system.
5. The system should provide an **optimization module** that shall be responsible to define specific measurement configuration parameters based on the available information from previous measurements, user-defined conditions and spectrum management (SM) information.

Such module is referred by its optimization function since it is expected to perform similar activities as discussed in the chapter about the “State of the Art” when considering the work by Shi, Bahl and Katabi [58], adjusting spectrum sweep parameters such as dwell time, frequency bin size, start and stop frequencies, start and stop times, etc.

The optimization module is necessary because current technology does not allow to monitor all usable frequencies at all times, which implies that the system should accomplish the tasks as a best-effort.

6. After the measurement, results stored in specific formats defined by each manufacturer should be **decoded and converted** to an open standard format that will be used on the server repository. As previously presented on the discussion about the “State of the Art”, we propose the use of the HDF5 format.

Due to efficiency considerations, some basic analysis may be performed during this initial processing of the measurements, e.g. noise measurements and emission detection.



7. A specialized **configuration module** to each equipment is required to translate the optimized spectrum sweep parameters into the configuration script and activation procedures specific to each measurement equipment model.

Although each equipment might impose different restrictions on the optimization module, e.g. different scanning speeds, it is desirable that the configuration and optimization are kept on separate modules to allow easier maintenance and development of the system, especially in order to encompass equipment from different manufacturers.

8. When loaded into the repository, the **measurement data** already converted into HDF5 format should be indexed and analysed using various modules.

The indexing should include more than the simple extraction of metadata for later information retrieval. It must also include the aggregation of data to allow more efficient use of the storage.

Basic metadata such as location, timestamps and frequency range can easily be extracted from the measurement data, but more sophisticated analysis of the measurement data should be performed to enable the desirable increase on the analytical level of the information management system.

Most importantly to accomplish the spectrum monitoring tasks, the system should be able to detect emissions, from which evaluations such as occupancy rate, signal clustering and classification may be produced and used for information retrieval.



9. On the perspective of the information management system, the **measurement** equipment can be viewed as an autonomous external data source. Specific details on the equipment to be used for the pilot implementation were reviewed in section 3.2.1.



10. Once the data is in HDF5 format, it may be stored using any standard cloud storage resources.

Individual files may be merged into larger files in order to make better use of HDF5 optimization features, such as data compression, but sizes should be kept within the boundary of hundreds of megabytes or few gigabytes to avoid some of the limitations pointed out in the reviewed literature.

Keeping measurement data in the HDF5 format allows for easier interoperability of the system since files could simply be downloaded and uploaded by users for analysis using other compatible platforms that include most of the more successful engineering tools.



Reference
data
repositories

As previously discussed, configuration files may be used for specific purposes and improve interoperability, e.g. using an OWL file to describe the ontology used in the data repository. These files should not use the measurement repository, but a separate repository dedicated to a specific purpose.

11. **Measurement data index**, as the name describes, is the index for information retrieval on the data storage.

It may include as much information as needed to enable such operation and may include images, bitstreams or other simplified representations of the data used in the GUI, allowing for a better response time of the user interface.

12. The **reference data** on the diagram represents all information associated with the system configuration and also any other additional information that may be needed by the users to interpret the data but is not directly derived from the measurements, e.g. spectrum management information on [assignment](#), [allocation](#) and [allotment](#).

5.4 Detailed design

Having presented an overview of the proposed solution one may concentrate on key aspects that will create the foundations to the information management system.

In the following sections 5.4.1 and 5.4.2, we start by considering the data organization, concentrating on how the measurement data shall be organized within the proposed system including the reference data imported from related systems within the spectrum management process and the metadata associated with the measurement information.

Later in section 5.4.3, we concentrate our attention on the core system modules, trying to functionally describe them with enough detail to allow the initial development. This description relies on several references already presented in chapter 4. This detailed description of the system modules expands the simplified view presented in section 5.3 including a description of services that are needed to operationalize the system.

5.4.1 Reference data and measurement index

When reviewing the user stories and system description, one concludes that to better index and retrieve the measurement information, the user requires more than the measurement data. It is also necessary to associate these measurements information with known references to the user, such as the spectrum [assignment](#), [allocation](#) and [allotment](#).

Additionally, as briefly discussed in section 4.1.3, the measurement data must be associated with some essential references to maintain its integrity as an accepted metrological information record. This includes data about the measurement units, the identification of who is responsible for the measurement, using which equipment, which procedure and under which conditions.

The preferable solution is that all metrological metadata associated with the measurement should be stored along with the data itself, within the HDF5 file as discussed later in this chapter, but some essential metadata must also be replicated into the measurement index since it will be required by the GUI to facilitate the information retrieval.

These concepts are graphically depicted in Figure 12, in which one can start from the core measurement data and move to the characterization of the measurement source and its metrological references, or, on the other side, move to a more analytical perspective of the measurement data, associating it with the characterization of the noise environment or of specific emissions, which by themselves might be associated with bands, channels or even specific emitters.

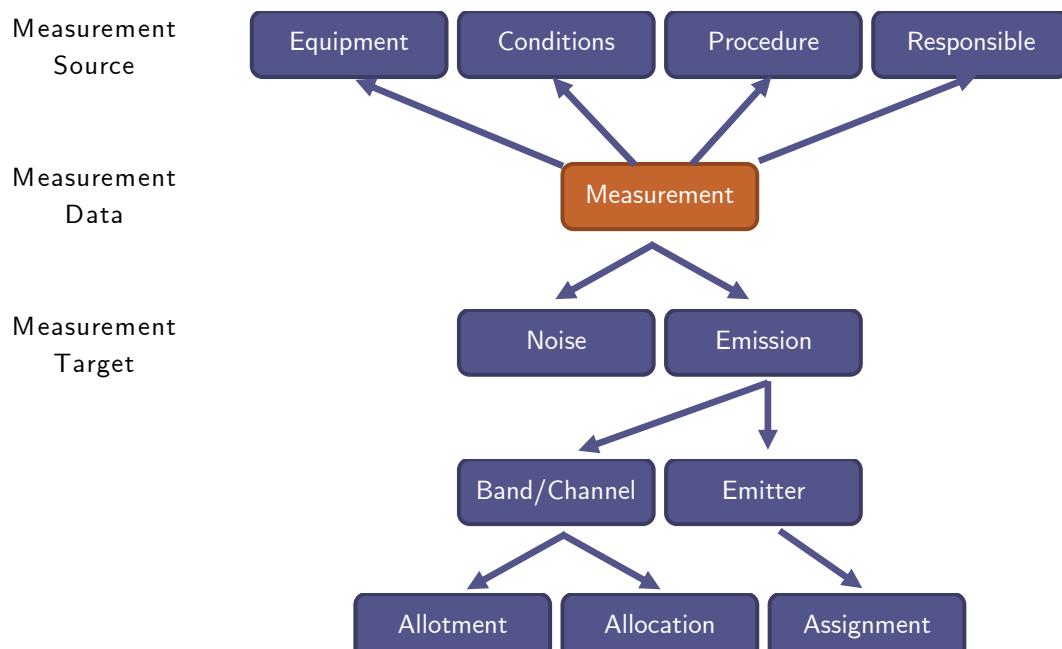


Figure 12. Basic relations associated with the spectrum measurement data.

This basic idea presented in the above diagram can be developed into a relational model, presented in greater detail in annex 10.3.

5.4.2 HDF5 Measurement data files

As previously discussed, the measurement data shall be stored using HDF5 format and, as presented in section 4.1.4, the best use of this format demands the design of data objects and association structure, i.e. data shall be stored as datasets within groups and metadata shall be stored as attributes associated with the datasets and groups.

Furthermore, one may say that the naming conventions associated with datasets, groups and attributes should follow a clearly defined ontology to enable easier understanding and adoption of the defined structure.

Due to limitations on the scope of the present work, a complete description of the reference ontology in OWL or another standard format neither will be proposed nor developed, but some effort was taken to reduce inconsistencies within the proposed HDF5 format and also to harmonically integrate the naming conventions used with

existing ontologies and ITU standards (see section 4.1.4) advancing the future work on the development of the mentioned ontology.

A detailed description of the proposed HDF5 structure is presented at annex 10.4 but we may highlight at this section a few key elements.

Throughout the proposed structure, attributes and even datasets are assigned as mandatory or optional. For example, if a measurement dataset is included, the corresponding measurement unit attribute must also be included but features such as tags may be left out.

Another commonly observed feature is the use of a suffix “.XYZ” to individualize multiple objects that adopt the same root name within a single group. This is necessary due to a functional restriction of the HDF5 format.

Figure 13 presents the basic root structure of the proposed format and from here one can see a few differences compared to the ITU-R SM.2117-0 [41], most importantly, the proposed format creates a structure using groups that are mandatory while the mentioned recommendation leave this alternative open.

Such change is required on the present case due to the proposed aggregation into a single format of a more diverse set of data objects. Such added complexity also implies that the idea of creating a self-explanatory structure and object naming convention falls short of the needs, requiring a more formal definition of the used ontology and reference standard.

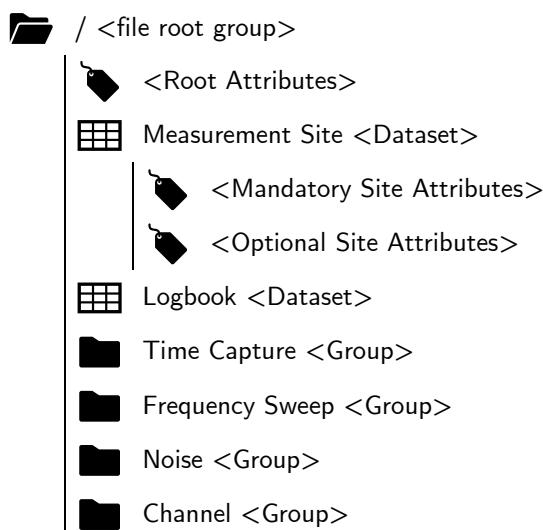


Figure 13. Tree view of the proposed HDF5 file structure including only the objects on the first two levels

Considering the first level of the proposed structure one may highlight a few key aspects discussed in the following points. At these, the figures indicated provides a direct navigation point to the corresponding detailed structure description in annex 10.4, where backlinks are also provided.

- The root attributes (Figure 32) include key metadata associated with the file creation and used format, including the link to the defining standard and reference ontology.

- The “Logbook” dataset (Figure 32) was conceived as a repository for historical information about the data contained within the file.

It stores events using a compound data type composed by a timestamp; an event type string; and a description string.

Since it employs such a simple structure, the logbook may be used to store any arbitrary text object or byte-stream, such as system messages recorded during the data acquisition, identification of the used measurement equipment, software versions, etc.

The event types should be defined within the created ontology to enable easier interpretation.

- Raw measurement data is to be stored within the “Measurement Site” dataset (Figure 32), the “Time Capture” group (Figure 33) and the “Frequency Sweep” group (Figure 34).

the “Measurement Site” dataset (Figure 32) stores the coordinates of the location where the measurements were performed.

The site representation uses a compound object including timestamp and a set of coordinates. This allows for the storage of fixed or mobile measurements, in which case the timestamp information should be used for correlation with other measurements within the file.

Within the “Time Capture” group (Figure 33) may be stored any number of independent I/Q data records, essentially using the format defined by the recommendation ITU-R SM.2117-0 [41], with the added grouping structure.

The grouping structure, not enforced or prohibited in any way by the mentioned ITU standard is useful to avoid the creation of too many objects with unrelated information within a group object or the root path, i.e. mixing together I/Q data, spectrum trace data and channel or noise analysis data.

The last group containing raw measurement data is the “Frequency Sweep” group (Figure 34).

As indicated by its name, it stores the power level of entire frequency bands in the form of spectrogram datasets.

Much like observed with “I/Q” groups within the “Time Capture” group, multiple “EM Spectrum” objects, containing data from independent bands and spectrum sweep tasks, may be grouped within the “Frequency Sweep” group.

An important aspect is that each “EM Spectrum” group contains several datasets. One as a bidimensional array containing the spectrogram itself, others containing the associated frequency and time axis. The time axis itself was divided into two datasets, one containing the coarse timestamp using POSIX time in seconds with no leap second and other containing the fine timestamp, in nanoseconds within each second.

This structure was considered more adequate to allow direct use of the stored data into standard time objects used in Python and C++, without the need for any conversion or formatting.

Although trace data and spectrograms can be derived from the I/Q data, on many situations only the trace data will be available, since I/Q data is more demanding on the processing and storage resources of the measurement equipment and may not even be recorded by many legacy devices.

- Analytical results may be stored within the “Noise” group (Figure 35) and the “Channel” group (Figure 36).

As suggested by their names, it is assumed that there are only these two types of objects to be observed within any measurement data. This requires a broader understanding of noise and channels and may be better understood by comparing the data structures in both cases.

In short, a channel is a subset of the radio spectrum where a time-dependent behaviour of the observed power level is of interest. Such concept may be associated with the presence of an emission, which may be transmitting or not at various intervals and thus display a relevant change in the form that the spectrum is available at that specific frequency range defined by the channel boundaries.

One must include in this concept extreme cases such as broadcast stations, that may be continuously transmitting, or specially reserved channels, such as those used only on emergency, that must be periodically inspected to guarantee that they are free of emissions. Even in such extreme cases, the time behaviour, always on or never on, still of interest.

To include the time activity information of a channel, an “Activity” group is created containing record of the activation and deactivation events, stored individually as a timestamp to the moment when the event was detected and the duration of the event. It also includes statistics about how often the channel was inspected and other pieces of information that are needed to splice the event duration when merging different files.

As a counterpart to the “Activity” group, another object called “Level” was created to aggregate information unrelated to the time behaviour. This object presents the histogram of the power level considering all samples available and all frequency bins of the described within the channel.

On the other hand, the “Noise” group stores everything that does not fit the described definition of channels, thus, the “Noise” group is expected to store only time-independent information about wider frequency ranges, such as the “Level” object.

The power level histogram stored in the “Level” object is an important tool to estimate the probability distribution function of the power on the measured frequency range, a piece of useful information to estimate the availability of the spectrum for communication.

Either the “Noise” group and the “Channel” group may include numerous groups identified as “Profile.XYZ”, where the “XYZ” suffix corresponds to an identifier, e.g. start or central frequency in kHz. This allows for the same file to organize several channels or noise measurement bands in a format that is easily browsable.

Within each “Profile” group, either within “Noise” or “Channel” groups, raw measurement data such as “I/Q” and “EM Spectrum” may be stored, allowing for easier retrieval of such information.

As a final example of the structure proposed, in Figure 14 we expand the groups and datasets under the frequency sweep group.

On the example presented below, considering the storage of several “EM Spectrum” objects within the same “Frequency Sweep” group, it was used the suffix “.XYZ”.

The power level at each frequency bin and at each instant is stored within the “Spectrogram” dataset, the centre frequency of each bin is stored in the “Frequency” dataset and the time on the “Timestamp Coarse” and “Timestamp Fine” datasets.

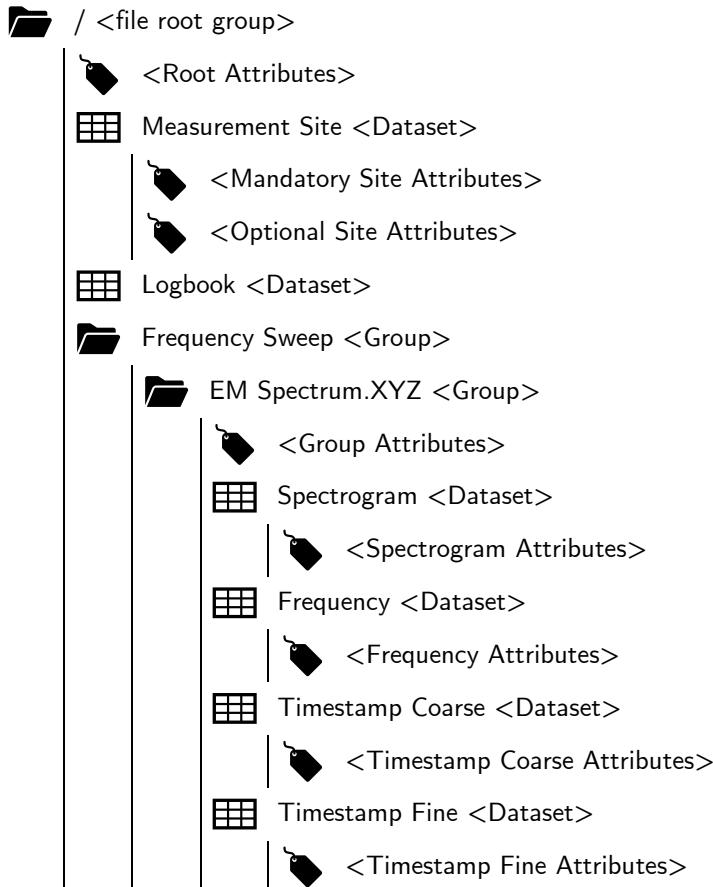


Figure 14. Tree view of the proposed HDF5 file with highlight to the frequency sweep group

5.4.3 System modules

As previously presented, we propose a system organization based on event-driven microservices in a reactive architecture. This alternative provides greater flexibility for the development and integration of different tools into a coherent system and can

provide more scaling flexibility to address variations on the processing and storage demands.

Using such architecture is essential that each module and daemon include systematic reporting of all operations to allow the functional monitoring of the complete system, promoting the desired operational reliability. Specific functionalities that monitor and control the activation triggers are not described but should also be implemented.

5.4.3.1 Optimization module

The optimization module implementation is based on the references discussed in the chapter about the “State of the Art”, most notably the work by Shi, Bahl and Katabi [58]. Due to differences in the type of data acquired, the algorithm proposed by Shi, Bahl and Katabi [58] can not be directly applied. Among the most significative changes associated with the use of trace data, one need a strict definition of the resolution bandwidth and revisit time, parameters required by the measurement equipment.

The optimization aims to reduce the probability of missing a switch on a channel status, e.g. from free to occupied, something that might happen if the channel is not properly detected or if the time between frequency sweeps is too large, resulting in a miss of the transmission event.

The minimum probability of missing a change of the emission status occurs when all changes are detected, or, in other words, when the time and frequency behaviour is fully described. This can be achieved by approximating the time and frequency behaviour to a square function, e.g. an emission might be present or not; a frequency bin might be inside the occupied channel or outside.

It is known that such approximation is severely limited due to the theoretically infinite number of harmonics of a square wave. As a consequence of Nyquist-Shannon sampling theorem, one can say that to obtain acceptable reliability that all events were captured, no less than 3 samples should be taken within the smallest event duration (on or off) or the smallest frequency separation between breaks (channel width or channel guard band). A better approximation should follow an even number of samples, e.g. 5, 7, 9, etc. since these would correspond to the harmonics present on the square wave function.

Considering the frequency parameters, this approximation implies that the bin width should be at least a third of the occupied channel or channel separation width, for the emission of interest with the narrower bandwidth within the scanned frequency range.

In an equivalent manner, the revisit time should be at least a third of the briefest event of interest, be it the presence or absence of an emission.

In many instances, these ideal sampling periods can hardly be met, and extrapolation using statistical models and reducing the confidence level will be needed to estimate the reliability of the obtained results.



Considering the idea of a system that can be initialized with no previous information about the spectrum usage conditions, the best approach would be to consider an initial data acquisition phase when each band is scanned with the smallest possible bin width, and with the smallest possible revisit time, for a brief time period, one hour or less.

This would allow for the creation of a baseline and the beginning of a process of increase on the revisit time and bin width, allowing for longer monitoring times for each band and wider bands. The repetition of such procedure with certain regularity would enable greater confidence in the defined parameters, including the definition of statistical models that may allow for a realistic estimation of the reliability of under sampled frequency bands.

Considering the user input, the optimization module should only require information on the highest possible level, e.g. wide frequency bands, areas and periods of interest. This information should be consolidated with the previously evaluated spectrum usage conditions and generate a schedule to perform the spectrum monitoring process. Inputs and outputs for this module are listed in the following table:

Table 4. Optimization module trigger events, input and output.

Trigger Events: User input; New measurements

Input	Output
<ul style="list-style-type: none"> ▪ Location Single or multiple monitoring station sites. ▪ Frequency range Frequency sweep range may be defined as start/stop, centre/width or simply all frequencies. It may include a minimum or a maximum bin width restrictions. ▪ Minimum detectable channel width Channels narrower than the defined values will be ignored. ▪ Minimum detectable channel power Channels that are not discernible with greater power difference from the noise than a minimum value will be ignored. ▪ Equipment characteristics Limit the evaluation to a specific monitoring network. 	<ul style="list-style-type: none"> ▪ Station IP direction ▪ Start frequency ▪ Stop frequency ▪ Bin width ▪ Antenna port ▪ Reference level ▪ AGC (fixed gain)

Table 4. Optimization module trigger events, input and output. (cont.)

Input	Output
<ul style="list-style-type: none"> ▪ Time Period for the measurement event on a schedule. It may be a single event, recursive or permanent (every time). ▪ Priority Used as a norm to select between conflicting schedules. As an initial proposal, one may consider 3 priority levels: Low, for tasks that might be dropped; Standard, for tasks that will be executed on a best effort basis; High, for tasks that will overwrite all other tasks and be executed. ▪ Time frames A recurrence time frame to visit a certain frequency range in case the desired time window cannot be met. e.g. try to monitor for an entire week at least once a year. ▪ Maximum revisit time ▪ Minimum emission presence duration ▪ Minimum emission absence duration 	<ul style="list-style-type: none"> ▪ Start timestamp ▪ Stop timestamp ▪ Interval (time between sweep runs)

A final aspect to be considered is that most of the available measurement equipment can provide results with some level of processing, usually at least averaging of several spectrum sweeps with the intent to provide a smoother result.

For the present system, we consider important to avoid such averaging and thus the parameter associated with the number of sweeps to be averaged should be kept fixed as 1, i.e. one should perform single sweeps with no averaging at the equipment level.

The rationale for this is that the noise reduction due to time averaging is equivalent in case of real-time or post-processing, with the advantage of better detection of fast signals and strong in favour of the post-processing alternative. This assumption may vary in accordance with the specific characteristics of the used measurement equipment, but such discussion is considered beyond the scope of the present study.

5.4.3.2 Module for measurement instrument configuration

The measurement instrument configuration module must take the sweep parameters scheduled by the optimization module and load into the measurement equipment using the required format.

There is no truly standardized API for measurement instruments, although most manufacturers provide some API that can interface with the most common programming languages, allowing the development of the configuration module with little effort.

Considering our reference case at [Anatel](#), there are several interfaces that allow the control of the measurement equipment provided by [CRFS](#). The best alternative for the planned systematic monitoring using this equipment is to have the onboard computer handling all data collection tasks using an application called RFEye Logger, that may be controlled using text files describing configuration scripts and HTTP POST methods. This application is available on all monitoring stations used by [Anatel](#)

For this specific equipment, the configuration module function translates the schedule created by the optimization module, previously described, and produce a set of RFEye Logger configuration files to be deployed and executed at specific times and specific stations. A detailed description of the configuration script syntax is presented on the application user manual [20].

The following table presents a brief resume of the input and output for this module.

Table 5. Configuration module trigger events, input and output.

Trigger Events: New configuration parameters from the optimization module

Input	Output
<ul style="list-style-type: none"> ▪ Scheduled configuration parameters Produced by the optimization module 	<ul style="list-style-type: none"> ▪ Set of RFEye Logger configuration file ▪ Interface daemon schedule

5.4.3.3 Instrument interface daemons

To execute the required spectrum sweep, an application needs to interface with the measurement equipment at the required moment and execute the required operations. This function shall be executed by an independent service referred here as measurement instrument interface daemon.

This interface service may be merged with the configuration module whenever the measurement equipment can only be controlled through an online remote control, that is not the case for most of the existing equipment at [Anatel](#).

For the specific case of the [CRFS](#) equipment, the daemon has the function of transferring the files to the specific station, at the needed time, and send the commands to stop measurements, reload the measurement script and start the new measurements.

The monitoring station control is conducted through HTTP methods described on the application user manual [20]. In short, one may reload the configuration file, pause and restart the logger application, reboot or soft restart the measurement equipment.

Due to security issues, the HTTP methods may only be accessed from the localhost or from the VPN, which means that an [SSH](#) remote access or VPN connection must be established prior to any action. Both connection alternatives are automated by the

manufacturer applications and the configuration module must only select the desired station and issue the commands as described in the manual.

File transfer is also restricted through the same connections. An FTP protocol may be used to transfer files to the appropriate system path, before reloading the configuration script into the RFEye logger application using the mentioned HTTP method. Another alternative is the use of the Linux utility [rsync](#)¹⁷, available on the monitoring stations as part of the original system automation.

The following table presents a brief resume of the input and output for this daemon.

Table 6. Interface daemon trigger events, input and output.

Trigger Events: Update on the daemon schedule

Input	Output
<ul style="list-style-type: none">▪ Set of RFEye Logger configuration file▪ Interface daemon schedule	<ul style="list-style-type: none">▪ Measurement data

5.4.3.4 Decode and conversion module

Measurement equipment manufacturers usually employ proprietary formats to store the measurement data. To each different format used, a specific decoding and conversion module must be created.

For the case of [CRFS](#) and RFEye Logger, the data is stored in a binary format fully documented by the manufacturer [21] and the decoding process is trivial.

After decoding, the data should be stored on the HDF5 format following the proposed standard presented in the section about “HDF5 Measurement data”. The use of a single format on the storage allows for the creation of a single and unified view of the dataset, facilitating the creation of the analytical and GUI modules.

Considering the asynchronous nature of the data transfer from the measurement equipment to the repository, it is advisable that the decoding and conversion process operates into single files, with no data aggregation at this step. The rationale behind this strategy is that some delay should be allowed between file processing in order to allow the transfer of the complete set of sequential files before proceeding with the indexing, aggregation, and analysis, avoiding the need of reprocessing in order to account for previously missing data.

Table 7. Decode and conversion module trigger event, input, and output.

Trigger Events: New measurement data file

Input	Output
<ul style="list-style-type: none">▪ Measurement result file (.bin)	<ul style="list-style-type: none">▪ HDF5 measurement data file

¹⁷ <https://rsync.samba.org/>



5.4.3.5 Indexing modules

Once the data is in HDF5 format, it must be indexed to allow the operation of the information retrieval by the users.

The initial indexing must simply extract the relevant metadata already stored as attributes on the HDF5 files. Table 8 enumerates basic metadata that should always be available.

Other metadata may be present and should also be indexed if available, e.g. detected channel boundaries.

Table 8. Indexing module trigger events, input, and output.

Trigger Events: New HDF5 measurement data file

Input	Output
<ul style="list-style-type: none"> ▪ HDF5 measurement data file 	<ul style="list-style-type: none"> ▪ Measurement data index <ul style="list-style-type: none"> - Filename - Equipment ID - Site location (ID, Latitude / Longitude) - Start time - Stop time - Initial frequency - Final frequency

5.4.3.6 Data aggregation modules

Following indexing, files might be aggregated to allow easier manipulation and analysis.

The data aggregation also allows for more efficient use of the storage and HDF5 compression functionalities, although it also represents a risk, since larger files are more prone to data corruption and loss.

A balance between those aspects must be experimentally adjusted by system tuning after implementation since aggregation may also be implemented without actual changes on the measurement data content of the original HDF5, by using available functionality of this format, such as creating external links between files and updating the relevant metadata.

One important variation of the aggregation procedure is required when it is needed to splice data that already have been analyzed. This includes, for example, the combination of spectrum power histograms and channel data to create a unified view of the complete dataset.

Table 9. Data aggregation module trigger events, input, and output.

Trigger Events: Update on the HDF5 file index by the indexing module

Input	Output
<ul style="list-style-type: none"> ▪ HDF5 measurement data file HDF5 file index 	<ul style="list-style-type: none"> ▪ Aggregated HDF5 measurement data file ▪ Update on the measurement data index.

5.4.3.7 Analysis modules

Several analyses may be performed on spectrum monitoring data. In the present study we are going to concentrate the initial development on the essential tasks for spectrum monitoring by fixed stations, that is, those related to the spectrum usage characterization, including emission detection and occupancy measurements.

An ancillary analysis of the emission detection and occupancy measurement is the signal clustering and classification. On ideal conditions, signal classification provides better identification of users, but such algorithms would require an extensive ground truth database, something not available. As an alternative the clustering based on the spectrum usage patterns may be a useful tool, since it may allow for the discrimination of different users sharing a single band, avoiding errors on the occupancy measurement and providing alerts to anomalous behaviour.

The specific algorithms corresponding to different analysis are reviewed in the chapter about the “State of the Art”.

Table 10. Analysis module trigger events, input and output.

Trigger Events: Update on the HDF5 file index by the aggregation module

Input	Output
<ul style="list-style-type: none"> ▪ HDF5 measurement data file ▪ HDF5 file index 	<ul style="list-style-type: none"> ▪ Update on HDF5 measurement data file ▪ Update on the measurement data index

5.4.3.8 Web GUI modules

The description of the Web GUI was left to the end of this section since the functionalities are derived from the requirements and products of the previously described modules.

The Web GUI should provide CRUD functions to all system data, including input of configuration parameters, access to logs, reference and measurement data. Additionally, the Web GUI should include standard functionalities such as user identification and access control.

The more advanced features of the Web GUI are closely related to the analysis modules, since it may require specialized controls, such as to allow the interactive

exclusion or inclusion of wrongly detected channels, the annotation of classified emissions, etc.

Figure 15 provides a view of a navigation flow that can be used as an initial approach to the system interface.

Although the system could possibly be simplified down to a single page if necessary, the proposed navigation flow includes all mentioned functionalities to access data tables defined on the reference relational model described in section 5.4.1 and annex 10.3.

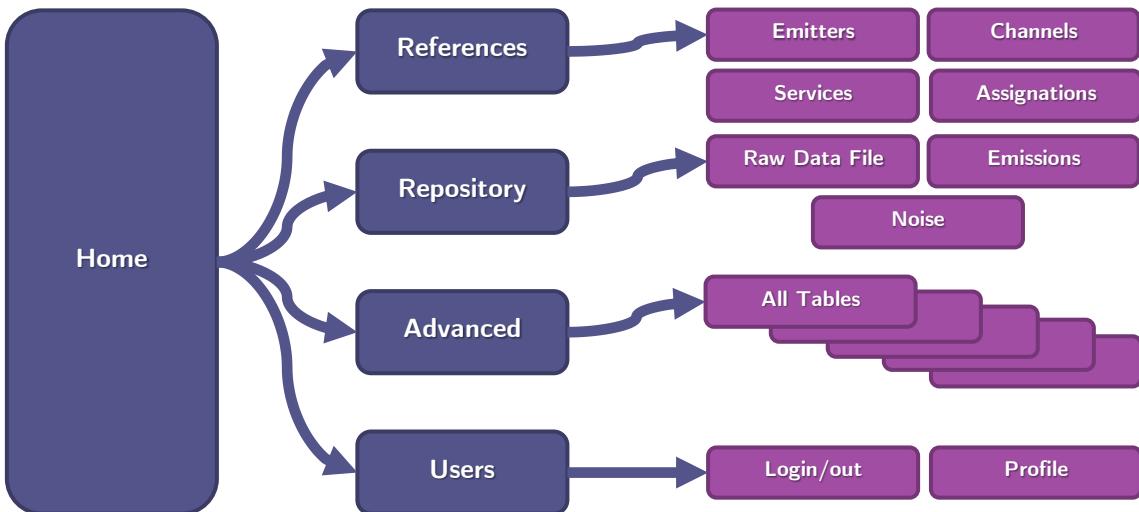


Figure 15. Navigation flow for the Web GUI

5.5 Technologies used

A key technology used on the current project is the HDF5 format and associated APIs. This topic was much discussed in the chapter about the “State of the Art” and complemented by the description on the section about the “HDF5 Measurement data”. Bearing this in mind, we will not further review this technology at this point, where we concentrate our attention on the discussion about the technologies used to develop the described modules, daemons and the Web GUI.

One important aspect is that, due to the complexity of the system, the current project scope is limited to the description of the architecture and the development of a prototype for the proof of concept, thus the present analysis includes not only the discussion about the technologies that shall be used on the more immediate system implementation but also the technologies that may be used on future implementations that may fully realize the proposed architecture.

The system requirements presented in section 3.3 define the use of several technologies, including a Linux distribution, the use of open software frameworks and standards. From these requirements, one may address various aspects of the system development to narrow down the alternatives to an actual and viable option.

5.5.1 Development Languages

When looking for a specific programming language for the development of the proposed system modules, one is also faced with a wide variety of choices, making almost impossible to compile a full list of alternatives.

Limiting our search to languages that include engineering and scientific packages that may be useful to facilitate the analysis and presentation of spectrum measurement data, the number of alternatives becomes much more restricted.

In fact, maybe the most used development environment on academic research for telecommunication engineering are Matlab¹⁸ and Labview¹⁹, but both alternatives must be excluded in the present case due to the open-source requirements. The next best alternative is Python.

This choice for Python is strengthened by the fact that the master course does not include specific programming topics and related subjects only included references and practices using R, Python, Weka and Tableau from which Python is the alternative with a larger number of telecommunication engineering packages and most suitable for the intended development.

Python is also an interesting alternative due to its availability on most serverless cloud platforms, allowing for future integration of the developed system on such deployment architecture, allowing for greater flexibility and scalability, something that cannot be achieved using R, for example.

Another benefit of the use of Python is the availability of several alternatives for the development of the system GUI, a topic that will be discussed in more detail in the following section.

Finally, Python is the most accepted language by the spectrum monitoring community within [Anatel](#) since it is the open programming language choice by several engineers and already have been used to develop a few tools that are currently in use to support regulatory enforcement activities. Such familiarity is a strong motivator since reduces the risk of non-acceptance by the organization.

During the initial stages of development though, it was not possible to achieve the desired performance for the decoding of binary files and storage using HDF5 format using only Python. Although such limitation most likely was a consequence of poor code optimization, a choice was made to develop this specific module using C++ language, allowing for the direct use of the API distributed by the HDF5 group and the flexibility of the C language to directly and efficiently map memory positions into different data types.

The use of C++ serves an additional purpose of demonstrating the flexibility of the proposed architecture in terms of integrating different development technologies into a single system, taking advantage of the benefits of each alternative.

¹⁸ <https://www.mathworks.com/products/matlab.html>

¹⁹ <https://www.ni.com/pt-br/shop/labview.html>



5.5.2 Application framework

As previously presented, the choice of Python enables the selection between a wide variety of frameworks for the user interface development while keeping the same language base, facilitating the future system maintenance.

Concentrating the initial search on graphical controls that enable the visual interaction between the user and the data one is driven to the concept of creating a dashboard that integrates all needed information about a dataset within a single page view.

In fact, one may say that the dashboard concept is the idea behind most of the reviewed applications described in section 4.1.1 and is a strong tool to promote system usability.

Considering available Python dashboard applications, the foremost choices are Bokeh²⁰ and Dash²¹. Focusing on the case of [Anatel](#), the agency has been using dashboards on numerous applications mostly provided by specific tools or developed in house using Qlik²² or SaS²³.

Both Qlik and SaS are proprietary alternatives and thus do not conform with the open-source requirement. Even more important, they share a restriction similar to that observed on Bokeh, that is the limited availability of controls to manipulate heat maps such as presented in Figure 4, Figure 9 and Figure 10, a common format used on the presentation of spectrum monitoring data.

On this last feature, Dash represents the most interesting alternative, an alternative made more interesting by the fact that Dash has a focus on instrumentation and scientific analytic interfaces while the other discussed alternatives are focused on business analytics. This difference is reflected in the type of data and presentation formats available on each framework.

One important aspect to consider is that although the availability of the needed controls is relevant to the selection of a framework, it is not essential to have such controls, since different approaches may be used to integrate various graphing libraries into a single GUI in order to create the desired dashboard. One may even encapsulate the Dash application, that runs using Flask²⁴, within a more sophisticated and powerful web server employing Django²⁵ or similar frameworks.

Since the current project has its scope limited to the development of the basic integration and on the measurement data organization, the selection of a Web GUI framework will be left open at this point, pending further discussion by the team tasked with the development of the final release version of the proposed system.

²⁰ <https://bokeh.pydata.org/en/latest/>

²¹ <https://plot.ly/dash/>

²² <https://www.qlik.com>

²³ <https://www.sas.com/>

²⁴ <https://palletsprojects.com/p/flask/>

²⁵ <https://www.djangoproject.com/>

5.5.3 Automation framework

Although a complete discussion about the system automation is beyond the scope of the present work, such automation is an essential part of the proposed system by monitoring and triggering the various process and modules.

Currently, [Anatel](#) spectrum monitoring network is not equipped with any relevant system automation. Files are transferred using [rsync](#)²⁶ and monitoring is centralized using a conventional network monitoring solution based on SNMP messages generated by the monitoring stations. Any actions or configuration of the stations need to be manually performed on each individual station using remote terminal connections through [SSH](#).

To the full deployment of the proposed system, the implementation of system automation is critical, not only to ensure that all system modules are running properly but also to update and reload the measurement scripts on the stations at the scheduled time.

The automation will have the additional benefit of enabling advanced management actions over the network, such as by implementing new security measures to restrict the access to administrative privileges over the monitoring stations and also by simplifying update operations.

There are several choices that may be used to automate the process including Ansible²⁷, Rudder²⁸, SaltStack²⁹, Puppet³⁰, Chef³¹, Cake³² and GitLab³³. The first two alternatives, Ansible and Rudder, are the most promising considering [Anatel](#) spectrum monitoring network architecture, based on Linux SBC configured via [SSH](#), that has little room for the installation of agents or programs that may interfere on the data acquisition process.

5.6 Pilot implementation

The pilot implementation focused on the development of the essential modules to demonstrate the basic functionalities of the proposed information management system architecture.

These functionalities include:

- the conversion of the raw binary data into the HDF5 format, as a demonstration of a decode and conversion module for the RFEye Node 20-6 equipment model;

²⁶ <https://rsync.samba.org/>

²⁷ <https://www.ansible.com>

²⁸ <https://www.rudder.io/en/discover/what-is-rudder/>

²⁹ <https://www.saltstack.com/>

³⁰ <https://puppet.com/>

³¹ <https://www.chef.io/products/chef-infra/>

³² <https://cakebuild.net/>

³³ <https://about.gitlab.com/>

- the automatic indexing of these files, as a demonstration of an indexing module;
- the aggregation of information from multiple data files, as a demonstration of a data aggregation module;
- the detection of emissions as a demonstration of an analytical module;
- the clustering of emissions as a demonstration of an analytical module;
- the identification of patterns on emission profile as a demonstration of an analytical module;
- the creation of visualizations of the aggregated data for emissions and noise information as a demonstration of Web GUI module;
- the creation of a web interface as a demonstration of Web GUI module;
- the creation of a reference database to structure the data used by the system;

As previously discussed, these modules were initially developed using Python 3.6, but due to performance issues, the data conversion and emission detection functionalities were integrated into a single module coded using GNU C++11.

The file processing automation employed the Linux subsystems inotify³⁴ and associated Python interfaces. This enables the creation of a simple and efficient solution and could be replicated to automate other tasks without the need for a central orchestrator. On the other hand, if a general-purpose automation system is deployed to manage the network, this simple event-driven automation procedure could be replaced to simplify the overall system maintenance.

The coding was conducted using Microsoft Visual Studio Code³⁵ running under Redhat Enterprise Linux 7³⁶ in a virtual machine created using Hyper-V³⁷ and Windows 10³⁸ professional.

This virtualization environment approximates the possible deployment within [Anatel](#), although different virtualization technologies and Linux flavour might be used. Such differences had driven the development to avoid hardware-specific alternatives, such as GPU acceleration.

For the demonstration of a Web GUI, the idea proposed is that the analytical components should plot their results into PNG images using matplotlib Python library. These images may be used as tiles to create a broader view of the spectrum over time, much in the same way as mapping software are able to represent the earth with distinct levels of accuracy using various zoom levels.

³⁴ <http://man7.org/linux/man-pages/man7/inotify.7.html>

³⁵ <https://code.visualstudio.com/>

³⁶ <https://www.redhat.com/en/resources/whats-new-red-hat-enterprise-linux-7>

³⁷ <https://docs.microsoft.com/en-us/virtualization/hyper-v-on-windows/>

³⁸ <https://www.microsoft.com/en-us/windows/>

The plot results may also be accessed via an interface such as the one created during the CMS course with Joomla!³⁹. The use of Joomla! was motivated by its regular use at [Anatel](#), which could reduce the acceptance risk. Another important feature is the availability of Fabrik⁴⁰, an application builder extension for Joomla! that enables fast interface building from basic SQL structures.

Although the use of a CMS to create the WebGUI do overload the system with unnecessary complexity, it is a useful demonstration tool for the flexibility of the proposed architecture.

³⁹ <https://www.joomla.org/>

⁴⁰ <https://fabrikar.com/>

6 Implementation

This chapter describes the pilot implementation for the proposed information system architecture.

Firstly, in section 6.1, the implemented modules are presented in detail, discussing the algorithms used for data analysis and key system functions.

Later, in section 6.2 are described the tests performed and on 6.3, examples of the results obtained for a sample dataset.

The chapter ends with a discussion about the implemented modules in section 7, lessons learned and changes to be made on the proposed architecture for future iterations of the system development.

6.1 Development of the proposed solution

The complete development of the proposed solution is a task too complex to be achieved in the timeframe of the present master project and due to such limitation, as previously discussed, the present work centred on the description of the system architecture, main data structures and organization models. At last, on the development of a pilot implementation that may serve as a first cycle on the development process of a fully functional system.

For the present demonstration, the system was named “Spectrum Cortex” as a reference to its function as an aggregator to the information coming from multiple radio spectrum sensors and on the association of this information into meaningful abstractions and representations, much in the same fashion as observed in the sensory and association areas of the cerebral cortex in mammals.

All code is available at a public repository⁴¹ and electronically attached to the annexe 10.5 in compressed binary format for conciseness. The code is extensively commented and mostly self-explanatory, including comments that aid the use and deployment.

The following subsections will describe all coded components, providing more detail on the most important aspects of some of the implemented algorithms.

⁴¹ <https://github.com/FSLobao/Spectrum-Cortex>



6.1.1 Ancillary components

A few ancillary components were created to allow greater flexibility on the implementation and reusability of the code. These components are briefly described in the following table.

Table 11. Ancillary system components

File Name	Description
cortex_lib.py	Ancillary tools shared by different modules including shared classes and methods for specific type and functions such as logging and debugging.
cortex_names.py	Constants used by python modules
h5_spectrum.py	Constants used by the Python modules that define the naming convention used for the HDF5 structure.
h5_spectrum.hpp	Constants used by the C++ modules that define the naming convention used for the HDF5 structure.
crfsbin.hpp	Constants used by the C++ modules that define the naming convention used for the CRFS bin file structure.

Changes in the “.py” ancillary components will directly affect the following instance of the program execution. Changes in the “.hpp” modules will demand a recompilation of the main program to take effect.

There is no input or output to these modules from the user’s perspective since there is no direct interaction with them.

6.1.2 Data format conversion module

Data conversion is performed by a single component which code is presented in the file “**decode.cpp**”. The program is designed to be executed from a command line, which allows it to be called by any system automation process. It decodes a [CRFS](#) “.bin” file in the format described in [21] and translate it into the proposed HDF5 format.

The decoding itself is a straightforward implementation. The data file is loaded into memory as a byte array and sequentially interpreted using C++ type recasting. The data is stored on various vector structures according to its content, using atomic types compatible with the HDF5 format already described. Required unit conversions and analysis are performed by [online algorithms](#) for increased performance.

At the end of the input data array, depending on the options selected, additional transformations and analysis are performed. The resulting data vectors are stored on a new file using the HDF5 C++ APIs.

Not all data options from the [CRFS](#) “.bin” format were implemented on the current version, only those directly need to retrieve the spectrum trace data along with location and time of execution. Text notes and audit properties were also retrieved and stored within the HDF5 logbook dataset.

Most notably are missing the decoding and conversion of time capture and occupancy objects within the original “.bin” format.

The time capture might be needed in the future to allow more advanced data analysis. In the case of occupancy, the original algorithm from [CRFS](#) is limited, based on a fixed threshold and providing only the occupancy percentage of each bin for a user-selected period of time. Such a simplified result is a useful alternative when local storage and/or communication resources are extremely limited, a situation not usually observed on [Anatel](#) monitoring stations.

6.1.3 Data processing automation module

The automation created in Python is presented on the code **“inbox_watchdog.py”**. This program should be running as a service on the server.

The [CRFS](#) RFeye logger application running on each node is configured to store the results every time the file size is larger than a user-defined value. Once the file is closed, it is transferred through the VPN using [rsync](#)⁴² to a specific folder on the server.

The automation program monitors the configured folders using Linux subsystems inotify⁴³. Once a file is created, the process becomes active and records the file name on a processing queue. Since the file transfer may take a long time, the process keeps monitoring the status of the file and only starts processing once the file is not changed for a previously defined period.

Each file is converted by an independent process thread, allowing multiple files to be processed simultaneously and providing greater scalability.

All processes are monitored by the automation program until completion, errors and success are logged as standard output and may be redirected to a log file as part of the service initialization.

One interesting aspect of the implemented automation is that it is purely event-driven, i.e. it takes little processing time when idle and activation is immediately following the event signalled by the operating system, without monitoring cycle delays.

A visual representation of the described process is presented in Figure 16. At this, illustration is presented the repository structure used to segment files according to their processing stage.

The automation itself provides the following services: the file move operations between the indicated folders; the call to the process “decode.o”, responsible for the file conversion and other operations; and the monitoring of each process thread such as to move the original file to a folder when an error is detected or to another, when the “decode” process concludes with success.

⁴² <https://rsync.samba.org/>

⁴³ <http://man7.org/linux/man-pages/man7/inotify.7.html>



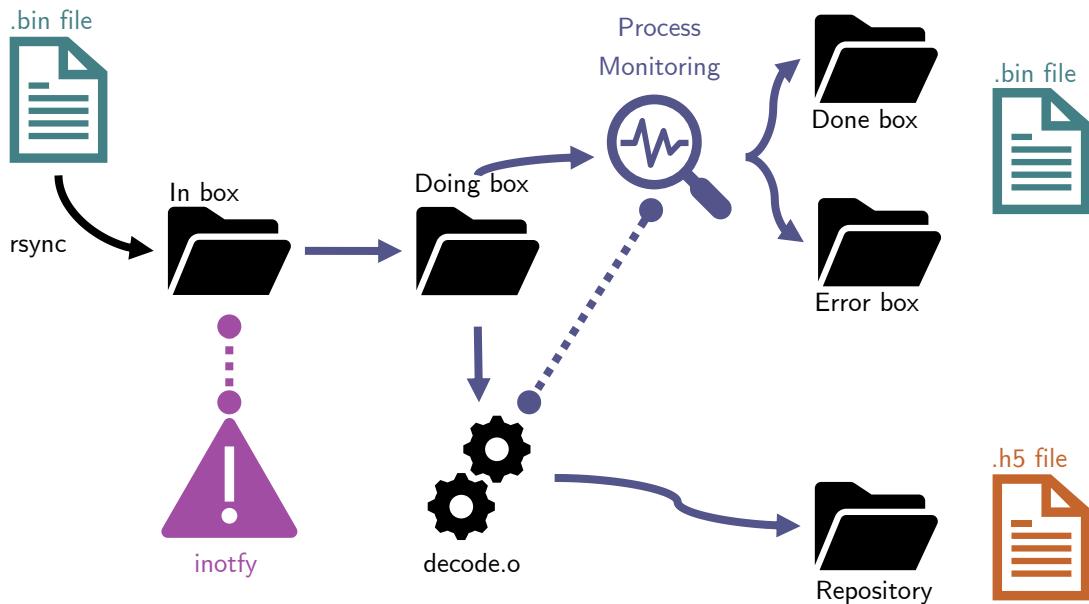


Figure 16. Automation of the file conversion process.

6.1.4 Indexing module

Indexing of the created HDF5 files is performed by the code presented at “[index_h5_files.py](#)”.

The same automation created to run the decode program can be configured to automatically index the HDF5 files incoming into the repository.

The indexing takes place by the use of three tables: a file index table including for each file a row containing the filename, initial and final frequency, start and stop timestamps; a site index table, including reference to the station name, latitude and longitude where the measurement was performed; and a channel index table, containing information of the detected channels including detailed information about the channel boundaries and assigned name.

For the pilot implementation, the index information is stored using pandas⁴⁴ dataframe and HDF store library. For the integration with the Web GUI, these tables should be stored within the SQL database as discussed later in this chapter, when discussing the Web GUI.

6.1.5 Data aggregation modules

Aggregation of the multiple HDF5 files is performed by the services implemented as presented on the code listed below:

- “[merge_geolocation.py](#)” takes as input the HDF5 files, the file index and the site index and compute the site latitude, longitude and altitude parameters. It uses statistics for each coordinate from individual files,

⁴⁴ <https://pandas.pydata.org/>

including average and variance to estimate the combined average and variance.

- “**merge_noise_profile.py**” takes as input the HDF5 files and the file index and splice the power level histograms for the noise data available into a single object by adding the hit count for each frequency and power level bin.
- “**merge_channel_index.py**” takes as input the HDF5 files, the file index and channel index and perform the merge of the channel index information. This includes identifying coinciding channels that might have different names, computing new boundaries and reassigning names. The merging algorithm is the same as the one used in the last phase of the emission detection algorithm, described later in this chapter.

The “merge_channel_index.py” program needs to be executed prior to the other channel merge algorithms since the channel index is essential to the following tasks

- “**merge_channel_spectrogram.py**”, takes as input the HDF5 files, the file index and the merged channel index and splice the corresponding spectrogram data creating a single representation of the channel for the entire merged period
- “**merge_channel_profile.py**” takes as input the h5 files, the file index and the merged channel index and splice the power level histograms for each channel into a single object by adding the hit count for each frequency and power level bin.

The activity log is also merged, including computation of the transmission and silence events duration, through the file boundaries, when files are adjacent.

Adjacency is determined by the time difference between the timestamp of the last trace in a file and the timestamp of the first trace in the following file. If this difference is smaller than two standard deviations computed from the time difference between traces within a single file, the two files are considered adjacent, otherwise it is assumed that there is no continuity between files.

6.1.6 Analysis modules

Three analysis modules were created for this pilot implementation. One is to perform the automatic channel detection and segmentation, other is used to identify anomalous behaviour of the emission within a channel by computing the self-similarity of the trace information and the third is to compare channels and group them into clusters of similar emissions.

One important aspect of the analysis modules is that they may not be as easily integrated into a working flow as the previous modules. The emission detection may be performed on every measurement performed, but the computational cost of the other



modules is prohibitive, especially because the complexity of the computation increases significantly when the number of channels or number of traces considered increase.

Such complexity implies that these analyses may be performed only under demand, considering a limited group of channels or traces specifically indicated by the user or further optimization need to be performed to allow the continuous analysis of all bands. Other strategies, such as the creation of standard references for classification instead of simple clustering, may also be explored in the future to solve this problem.

Two of these three algorithms are presented in greater detail in annexe 10.6, that may be used as a reference to better understand the implemented algorithms, prior to inspecting the source codes and associated comments.

6.1.6.1 Emission detection

The channel detection is the first stage of signal processing, performing the feature extraction needed to allow the meaningful analysis of the recorded information about emissions.

It is coded in C++ and expected to run as an [online algorithm](#) along with the initial file format conversion by the “**decode.cpp**” program. The source code is presented in the file “**level_differential_detector.hpp**”.

The used algorithm for automatic emission detection was discussed in the chapter about “State of the Art” and it is based on the idea of a Bayesian detector, i.e. detect emission by the variation it causes on statistical measurements, e.g. arithmetic mean, over a region of the spectrum.

Due to its complexity and specificity of the implementation, not easily found in the literature, this algorithm is presented in greater detail at annexe section 10.6.1.

Once channels have been identified, all traces corresponding to active emissions on each channel are consolidated into the same spectrogram, discarding the traces where the emission was not present. This information is stored as an EM Spectrum object within the corresponding “Channel” group. The remaining information is stored within the “Noise” group.

6.1.6.2 Emission self-similarity

The traces corresponding to emissions within each channel may be compared between themselves to identify variations on the power level distribution that may be associated with changes on modulation or on the transmission power. Such analysis is especially useful to differentiate various users that may share the same band or to isolate the occurrence of an interference event.

The approach of comparing traces allow for stronger characterization of the signals when compared to alternatives such as the extraction and comparison of isolated features such as occupied bandwidth and channel power since more subtle characteristics are also considered. e.g. an AM and FM modulated signals might have the same bandwidth and total channel power but the distribution of power within the band is quite different and easy to spot by observation of the spectrum trace, given enough resolution on the trace presentation.

To perform this analysis was coded the program “**compute_channel_inner_distance.py**”. This takes as input the HDF5 files and channel index and output graphical representations of the spectrogram with markers on the peak dissimilarities.

The mentioned program works by unstacking the bidimensional array of the spectrogram into a one-dimensional array. The dissimilarity is identified by locating the discords signalled by the peak of the matrix profile distance computed over this one-dimensional array, using as the query size the number of bins within the channel at a single trace.

The matrix profile algorithm was presented in section 4.1.5. The Python implementation, “matrixprofile-ts” is well documented with examples on Github⁴⁵ and may be installed with the default Python package installer, pip.

6.1.6.3 Emission clustering

As previously discussed, automatic emission clustering may be useful to identify similar emissions and thus create metadata that can be associated with known classifications and provide more comprehensive information retrieval capability.

Initial tests using the matrix profile algorithm to perform this task did not provide a meaningful result due to excessive noise on the trace data. A second more successful attempt employed a simple root-mean-square error (RMSD) computation as a distance measure between the mean trace of the two channels to be compared. The choice for the RMSD instead of a simple Euclidian distance is to avoid differences that would result from variations in the length of each channel trace, a common situation since channels boundaries are automatically detected.

Prior to the distance computation, a series of operations are needed to ensure that the results are insensitive to variations created by the measurement and processing conditions. These operations include scaling and cropping of the traces and is presented in more detail in the annex section 10.6.2.

This module is coded in the program named “**compute_channel_distance.py**” and generate as output dendrogram visualizations of the resulting clusters. A debugging visualization of the compared waveforms may also be produced with the selection of this option, performed with a configuration constant within the code.

6.1.7 Reference database and WebGUI

Although most of the modules described may operate autonomously through a cascade of event-driven calls, a central and unifying part of the system is the user interface and reference database described hereon.

To address the user requirements, the spectrum data must be associated with meaningful information about the spectrum usage, such as the assignment, allocation and allotment of spectrum bands.

⁴⁵ <https://github.com/target/matrixprofile-ts>



For the pilot implementation, we propose that such relations are described by a structured database represented by the relational model presented on 10.3. This model was translated into SQL that is presented in the file “**create database.sql**”, available at a public repository⁴⁶ and electronically attached to the annexe 10.5 in compressed format for conciseness.

As previously discussed in section 5.6, a demonstration Web GUI was created using this database structure with Joomla!⁴⁷ CMS and Fabrik⁴⁸. This implementation was performed without coding, using only the standard administrative interface of the mentioned applications to create the navigation flow described in section 5.4.3.8 based on the database structure already created.

To enable the data visualization, the mentioned simplified WebGUI depends on previously generated graphical representations of the spectrum data. To create such images to represent individual channels, an example module was created using Python and Matplotlib. The code is presented under the name “**plot_channel_profile.py**” and works by plotting all data on representing each channel on the repository into “.png” files that can be loaded into the WebGUI.

Although apparently precarious, the alternative of representing the complete dataset into “.png” files is not much different from what is done by web mapping applications, that uses image tiles to quickly produce a representation of the maps without the need of complex vector graphics editing and styling in real-time that could result on fuzzy or lower quality presentations.

6.2 Trials

Trials were conducted using a sample dataset collected by a measurement station installed on [Anatel](#) headquarters in Brasília, DF, Brazil. The data comprises a period of about 8 hours, on a regular Wednesday afternoon in April, starting at about 16:30 until after midnight. This data was collected in preparation for an actual measurement campaign and thus has no usage restrictions.

This specific data is useful to the current exercise since it was scanned with high frequency and time resolution, about 390Hz per bin and a bit more than one trace was collected every second. It includes only a small spectrum band from 450MHz to 470MHz. This band is allocated to several services, most of them using narrowband transmissions with different modulation types and usage profiles.

Although limited in time and frequency, this sample takes about 1.3GB of storage space. The complete dataset is comprised of 35 files, each containing about 40MB of data and describing a bit less than 800 traces, each with 51200 frequency bins.

This limited sample data is considered enough to achieve the current project objective that is limited to the architecture design and initial experimentation within a pilot implementation.

⁴⁶ <https://github.com/FSLobao/Spectrum-Cortex>

⁴⁷ <https://www.joomla.org/>

⁴⁸ <https://fabrikar.com/>

6.3 Results

Considering the proposed HDF5 format and conversion modules, the results can be inspected by use of the available HDF5 tools. The most intuitive tool available is HDF View⁴⁹. This application provides a visual interface to the HDF5 file, including a tree view of the hierarchical structure and the representation of datasets as tables and as graphical objects. The GUI of this application loaded with an HDF5 file produced by the implemented code is presented in Figure 17.

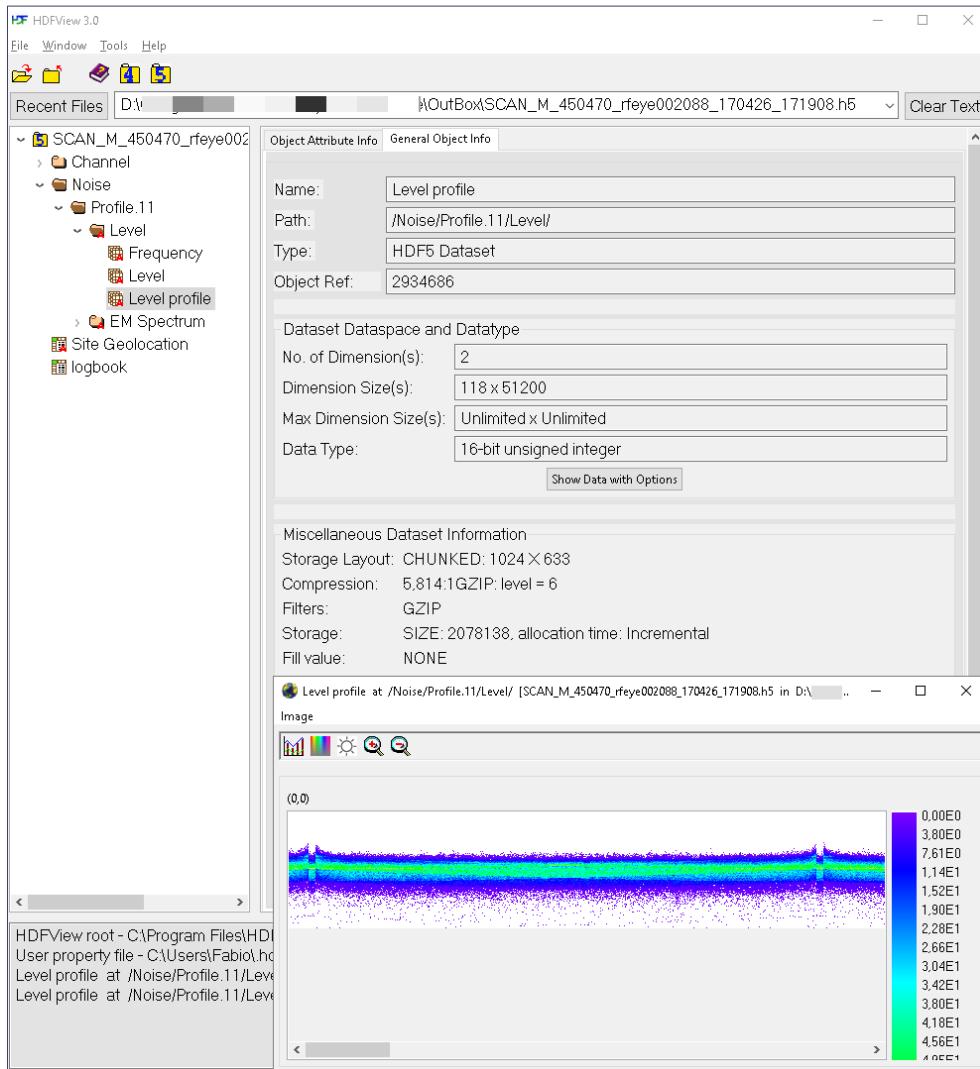


Figure 17. HDF View⁵⁰ interface presenting a converted data file with the noise level profile dataset

The figure above displays the level profile object for the noise. A small section of the dataset is presented as an image on the lower part. Channels that were removed by the detection algorithm can be visualized by the remaining out of band emission components. The dataset itself is a bidimensional matrix with 51200 columns and 118 rows using a 16bit integer data type. In this case, the stored value corresponds to the number of hits at each specific bin that represents a power level and frequency. There is

⁴⁹ <https://www.hdfgroup.org/downloads/hdfview/>

⁵⁰ <https://www.hdfgroup.org/downloads/hdfview/>

no maximum dimension set and the dataset is chunked into segments with dimensions of 1024x633. A GZIP level 6 compression is applied.

Concerning the emission detection module, part of the result was already illustrated in Figure 17, on the graphical visualization of the noise with the missing emission components.

The code created for the detection algorithm included a debugging output in graphical format using gnuplot⁵¹. An example of such output is presented in Figure 18.

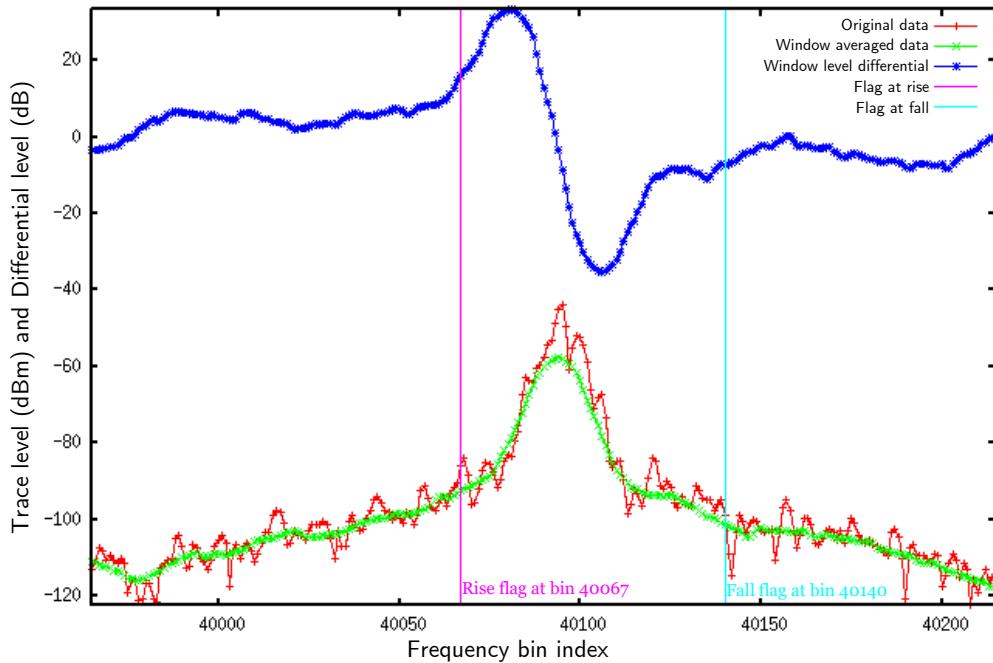


Figure 18. Detection algorithm debugging output.

Figure 18 presents a region including about 300 frequency bins from the raw data decoded from the RFEye Logger binary file. Centred on the frequency bin index axis one can see the peak of the original data that reaches about -40dBm. This rise corresponds to an emission. At the illustration edges, the power level drops down to the noise level below -110dBm.

The window averaged data presents a much smoother trace as expected since it represents the arithmetic mean value over a moving window of about five of the original data points.

The window level differential is presented in dB and would be the same as expected from the derivative of the presented averaged data function. This is not arithmetically correct though since an offset of several bins is introduced between the values used in the operation. As discussed, this offset is intended to compare the central peak region with another region representing the noise floor, thus increasing the differential value.

⁵¹ <http://www.gnuplot.info/>

The flags that delimit the emission are created by searching for a sequence of three bins that mark a local minimum on the averaged trace data, both before the positive differential peak (rise flag) and after the negative peak (fall flag). These flags are later consolidated considering various traces to define the channel boundaries for each file. When files are merged, these boundaries are again revised and merged into a single description of the channel as described in section 10.6.1.

The emission detection process on the sample data was able to detect activity on 151 independent channels. After data conversion into the proposed HDF5 format, is possible to visualize each individual channel using the Python module described in section 6.1.7.

With this module is possible to create visualizations such as illustrated in Figure 19 that present two emissions with distinctive characteristics.

The colour on these images represent the hit count as a histogram, lighter tones represent fewer observations of that power level at that frequency. In this case, the colour is presented only for a qualitative inspection and no numerical scale was created in association with the used colour range.

Frequencies on the horizontal axis are presented in relation to the peak value and the solid trace represents the arithmetic mean value of all traces used to create the histogram.

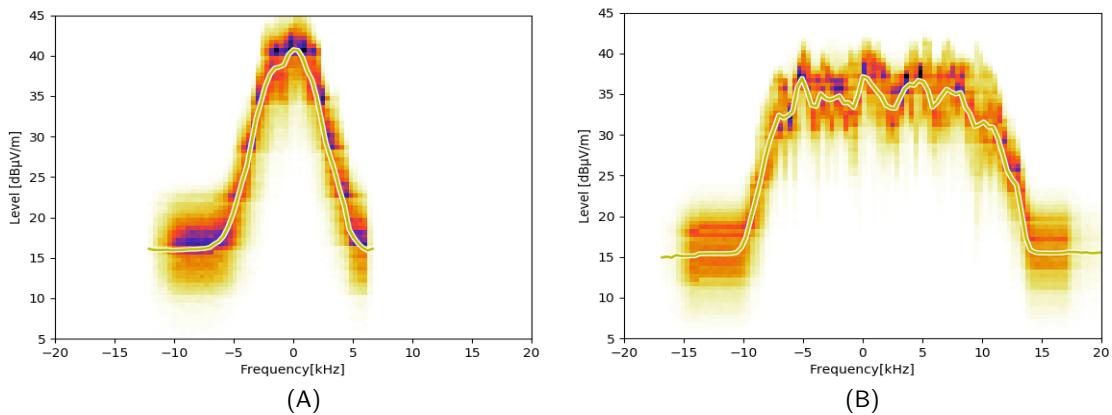


Figure 19. The power level of emissions (A) 12.5kHz emission and (B) 25kHz emission.

On the above picture, the traces from several files were merged, since the channel detection is performed on each individual file, the channel limiting frequencies might be different for each data subset due to changes in the noise and emission characteristics. Such change is observed at the edges of the frequency axis, where the number of hit counts is smaller and the trace is more noise affected.

This kind of visualization may allow for interesting analysis, such as presented in Figure 20, where one can see how the histogram hint to the presence of multiple users sharing a specific channel. In this case, one can see three emissions: a high-power, i.e. with peak level at about 60dB μ V/m, with narrowband emission that is very frequent and dominates the average trace and display the darker colours of the histogram; a lighter shadow representing another emission detected with slightly lower power, i.e. with peak level at about 55dB μ V/m, and a much wider bandwidth; and a third emission with much lower power, i.e. with peak level at about 30dB μ V/m, and also narrow bandwidth.

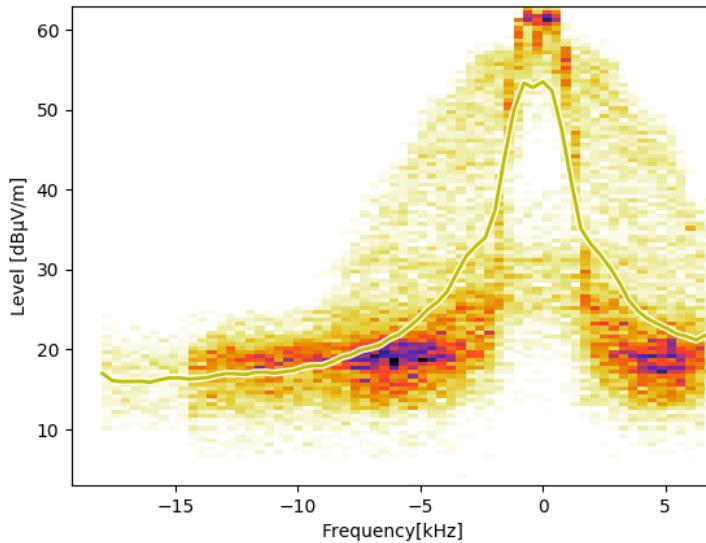


Figure 20. Channel with three distinct emissions overlapped

Although interesting to observe such superposition of emissions as presented in the figure above, this form of manual analysis does not correspond to the expected increase of the analytical level in the system since it still requires a human interpretation of the graphical visualization.

To automate such analytical process, we proposed the use of a metric of self-similarity based on the matrix profile distance, as described in section 6.1.6.2. The result from the coded module is presented in Figure 21.

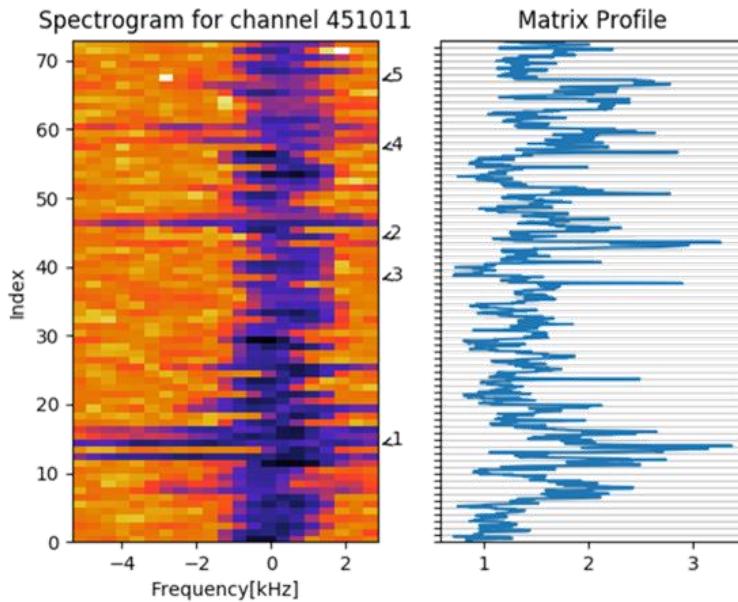


Figure 21. Matrix profile distance (right) analysis of a spectrogram (left) for a model case.

On the left side of the above figure is presented the spectrogram for a channel named 451011, where activity was detected at the frequency of 450.011MHz. At this channel, 73 recorded spectrum traces included discernible emissions.

The spectrogram presents the frequency on the horizontal axis and the trace index on the vertical axis. The colour of each point is determined by the power level. Brighter colours from red to yellow and white on the lower end corresponding to the minimum level and thus to the noise floor. The level increases from red to blue and violet until black, where is the highest detected emission level represented. The absolute values of the colour scale are not important for the present analysis.

On the right side is presented the matrix-profile distance, unitless as per definition. The peaks correspond to the maximum discord within the sequence and thus signal the emission traces where relevant changes on its more recurrent behaviour were detected. Flags were added with numbers to point to a few of the maximums, presented in decreasing order of the matrix-profile peak value.

Inspecting both graphs one can see that the matrix-profile algorithm was able to identify regions where there is a significative change on the power level over the observed band. Indicating the detection of wideband emissions on the traces with an index around 15, 47 and 60. It also points to some low power emissions on traces 40, 45 and 55. This result corresponds to similar conclusions as observed in Figure 20, although the presentation needs improvement and more experimentation in order to allow such results to become a real analytical tool.

Although promising, such result is more difficult to replicate in complex situations, with more traces, higher noise level and lower discrepancies, such as presented in Figure 22. Also, it was observed that, regardless of all optimization provided by the provided library, the matrix-profile computation still became awfully slow when processing thousands of traces, overtaxing the system performance.

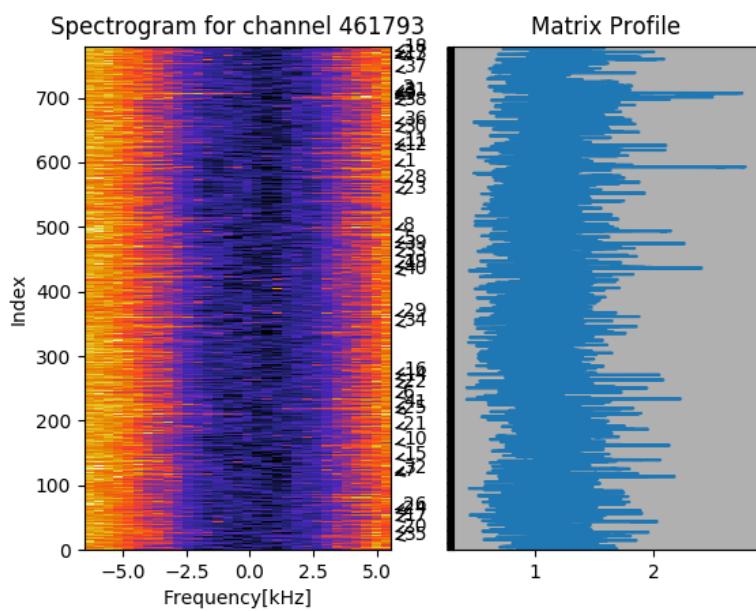


Figure 22. Matrix profile distance (right) analysis of a spectrogram (left) for a complex scenario

The other proposed analytical algorithm attempts a different approach for comparing traces. In short, the proposal is to create clusters of similar emissions using a distance metric computed between every pair of detected signals by using a normalized version of the spectrum traces, employing the algorithm described in section 10.6.2.

The use of the word “normalization” must be understood here on a broader and more common-sense meaning of scale transformation than on the strict mathematical sense of an adjustment on the statistical distribution of values within a sequence.

This normalization process can be inspected on the example presented in Figure 23.

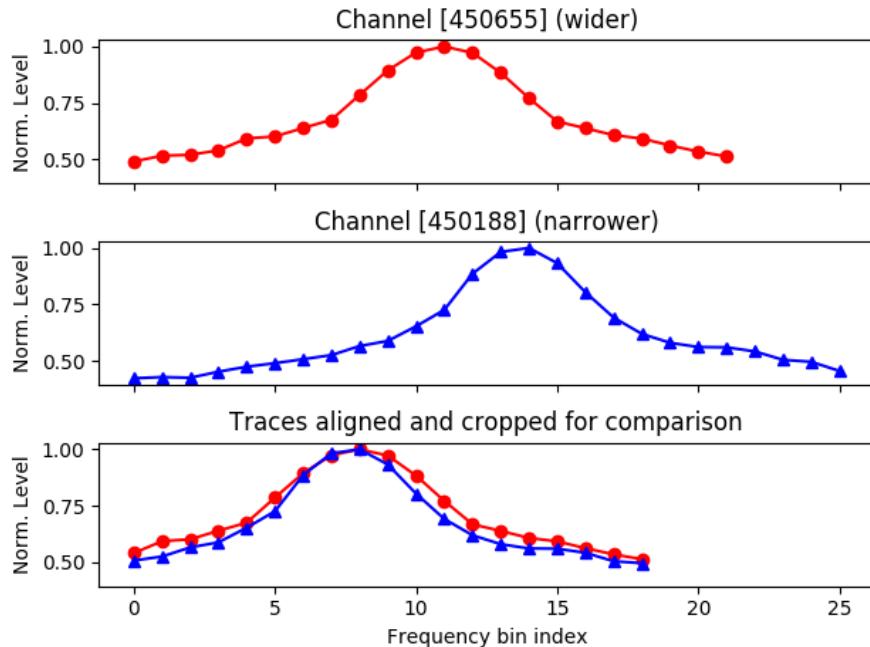


Figure 23. Normalized channel representation for comparison of two traces.

On the above picture, two traces to be compared are presented on the top and middle graphs.

All traces are rescaled by their respective maximum values, such as that the peak is found at the value 1. It is notable that the first trace on the top has a slightly smaller dynamic range than the second trace in the middle since the lower value is a bit higher than the one observed for the second trace

The horizontal scale represents the frequency bin index on the numerical array and is only presented for the third and lower graph. The first trace, on the top, has more points than the second trace and thus is identified as “longer trace”. This difference becomes clear on the third and lower graph when both graphs are adjusted and superimposed within the same scale. At this, it becomes clear how much alike are both traces and that the differences initially observed were only a product of the variation on the number of frequency bins and the shift due to channel boundary definitions.

The proposed adjustment procedure attempted to obtain the minimum distance measurement between the two compared traces. The distance used itself is the root mean square deviation (RMSD) computed for all frequency bin on the aligned and adjusted trace as presented on the lower graph.

In order to better inspect the effects of the described adjustment, Figure 24 and Figure 25 illustrates different cases with the same graphical representation.

While in Figure 23 very similar traces were compared, in Figure 24 we have a case with more distinct emissions. In this second case, due to modulation effects and characteristics, the first emission, on the top, appears to have the central carrier suppressed.

Nevertheless, the cross-correlation algorithm minimizes the error shifting the trace in the middle to the left, more to the beginning of the index, taking advantage of the higher level on the beginning of the trace on the top to reduce the overall error.

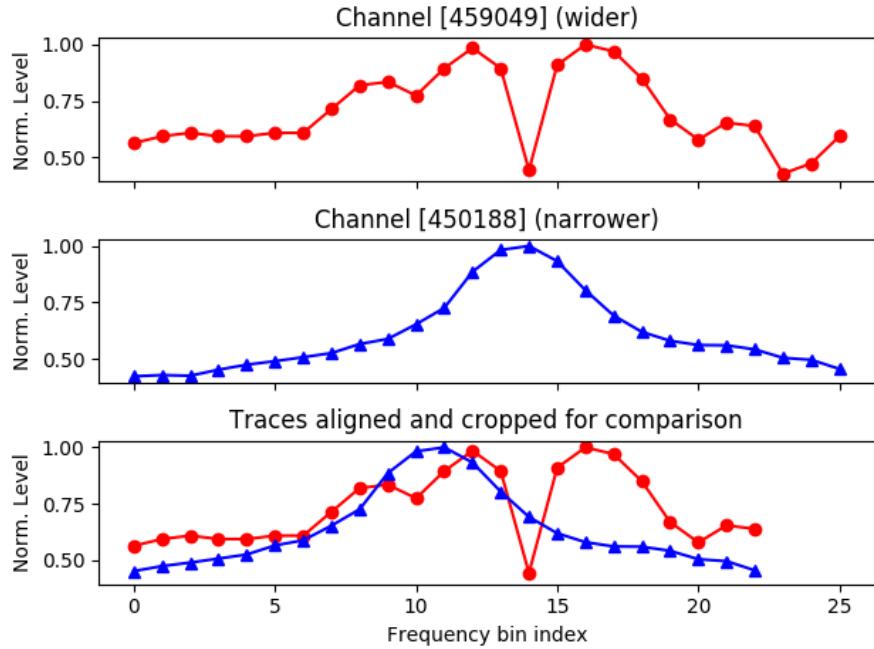


Figure 24. Normalized channel representation for comparison of two narrowband traces

On the third example presented on the image below, a wideband emission on the top graph is compared with a narrow band emission on the middle graph. After adjustment, the trace corresponding to the wideband emission becomes approximates a flatline on the top. The edges, when the power from the wideband emission drops to the noise floor, are out of the range. The distance between the traces becomes significant in this case.

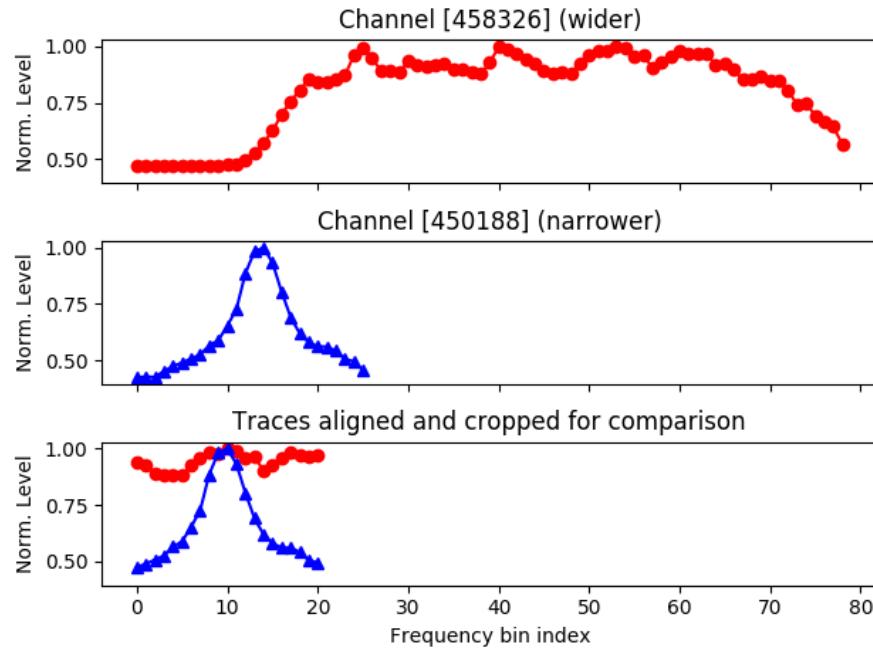


Figure 25. Normalized channel representation for comparison of a narrowband and a wideband trace.

After computing all differences, the resulting distance matrix for 151 channels on the sample data is presented in Figure 26. This image is just illustrative and allows only for a qualitative analysis of the results by identifying the expected variations and patterns, e.g. channels that present the same pattern, defining clear vertical and horizontal lines, result in low RMSD at their crossings.

The matrix is computed and presented in this case on the compressed form, i.e., less than half of the possible combinations are computed since the values are mirrored around the diagonal and the error on the diagonal itself is zero since it represents the comparison of a trace with itself.

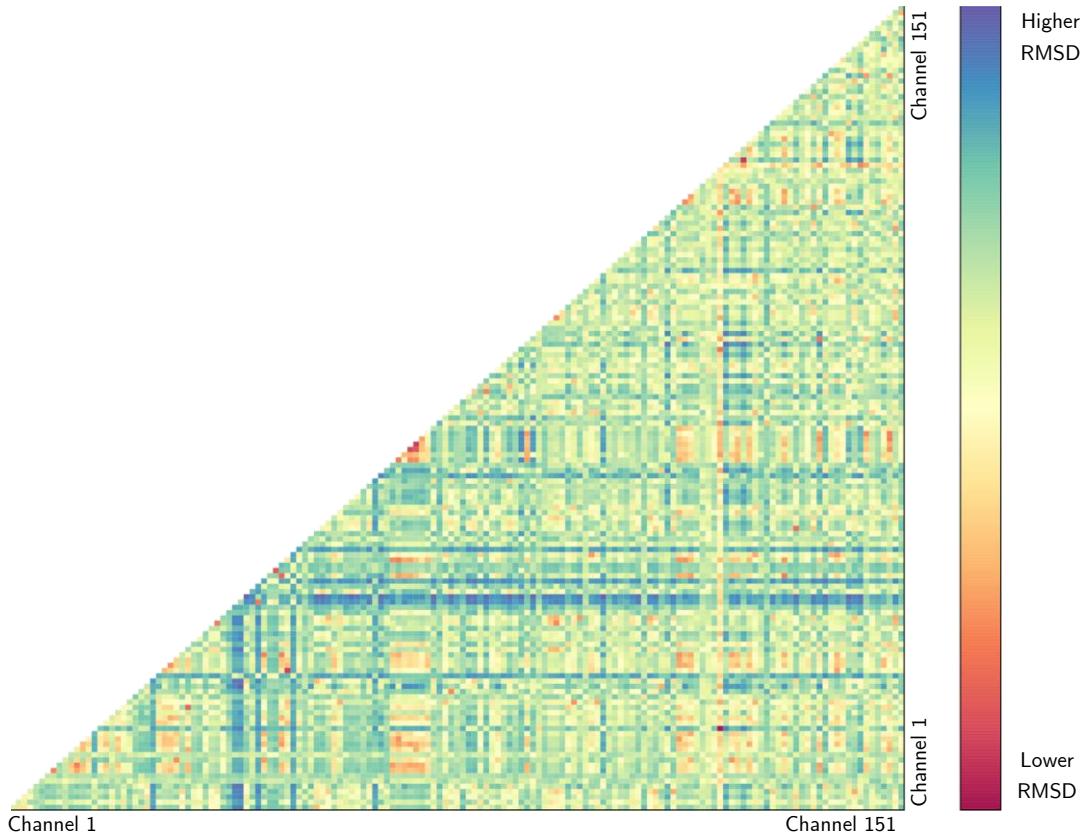


Figure 26. Distance matrix graphical visualization for 151 channels.

A more informative presentation is obtained after running the clustering algorithm and producing a dendrogram representing the different channel types grouped by their spectrum trace similarity translated into the smaller distance. Such representation for the 151 channels is presented in Figure 27 for qualitative purposes.

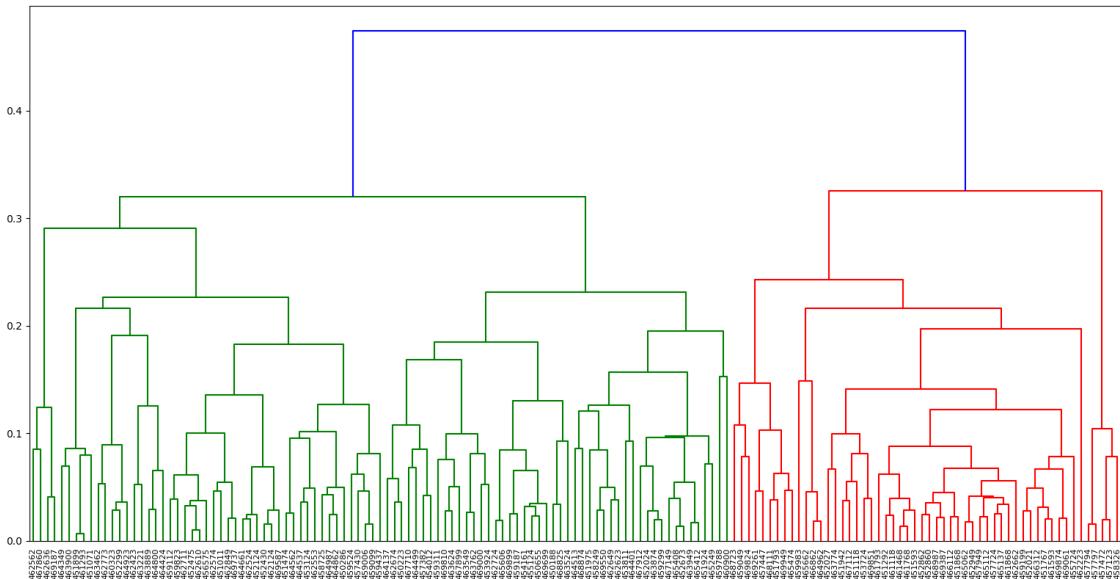


Figure 27. Complete dendrogram representing clusters for the 151 channels.

In fact, one can realize that small groupings, such as those including only two channels, are not representative. When one comes to the top two groups, there is a clear

separation between 97 narrowband emissions and 54 wideband emissions, with further subgroupings that might be of some interest.

This reduced dendrogram along with examples is presented in Figure 28.

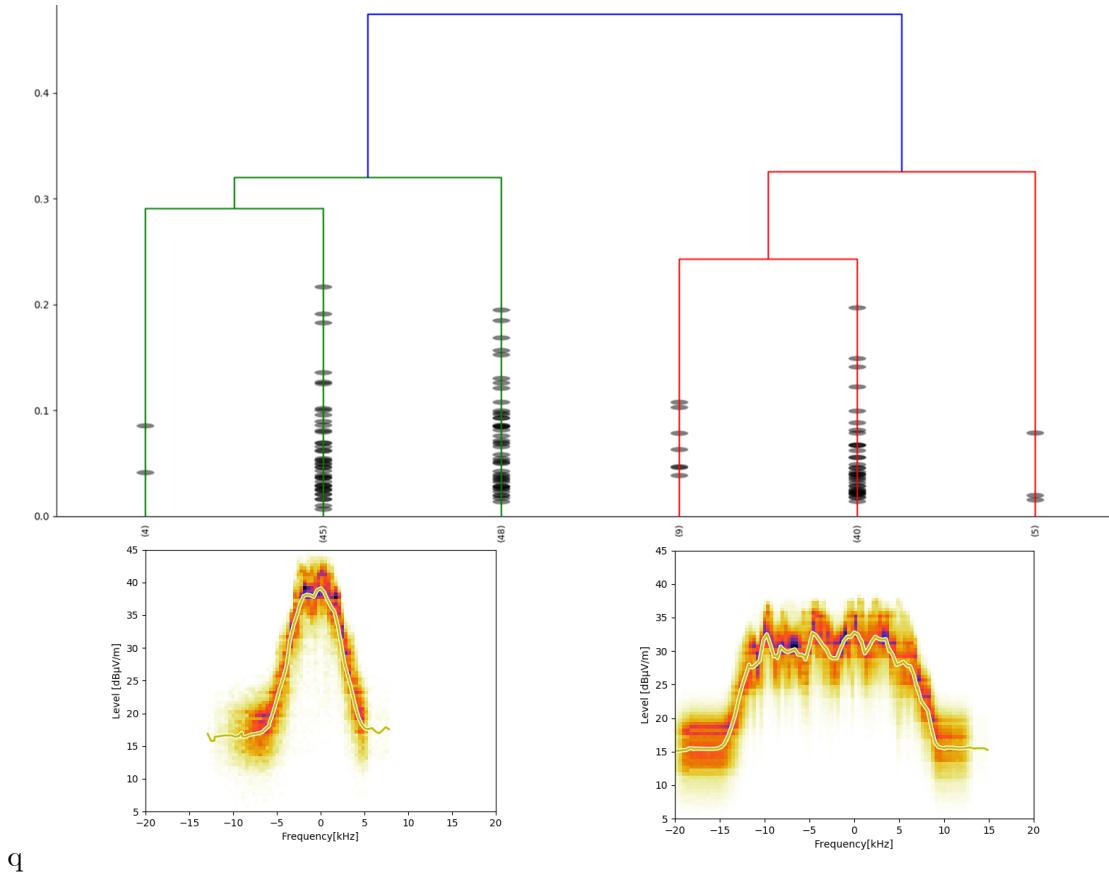


Figure 28. Dendrogram top part with most representative groups.

As previously discussed, the proposition considers that such analytical capabilities may be used to enable easier data retrieval, for example, a user may quickly find all emissions of a certain type within a band or generate statistics concerning the separate types.

All this user interaction will be provided through the WebGUI. A demonstration WebGUI was created in Joomla!⁵² as part of the CMS course. The main pages of the created GUI are presented in annex 10.7.

The integration of the created WebGUI with the remaining system modules was left for future activity on the project and for the presented demonstration example, the database was loaded with dummy data.

⁵² <https://www.joomla.org/>

7 Discussion

Beginning our discussion with the proposed architecture, the idea of using event-driven microservices is considered well fitted to the problem due to the complexity of the required implementation. Such a strategy allows for progressive development through several cycles with small deliveries associated with each independent service. This is expected to reduce the complexity by splitting the larger problem into simpler modules, allowing fast delivery of value to the users.

Also, the microservices architecture allows for the integration of existing solutions, most notably, the use of automation systems such as Ansible⁵³, Rudder⁵⁴, SaltStack⁵⁵, etc.

The realization of the pilot implementation was a best-effort attempt to create the foundations of the proposed system architecture, allowing to perceive the value of the development template, but although several important aspects were tackled, many remained open or were implemented only on a primary and prospective manner that may require a complete revision before reuse on a production environment

One critical architectural feature and likely the most important contribution of this project is the definition of a working format for data storage and exchange of spectrum monitoring data. There are numerous alternatives for digital data serialization and storage and the selection of the HDF5 format was a combination of factors but only the starting point since it requires the creation of a structure or schema that describe the objects and their relations within the desired domain.

On this perspective, even though this work presents a viable structure and object description, any definitive proposal should be in line with an accepted ontology standard, something that does not exist at this moment, at least, not anything created by the consensual agreement of trusted references of the domain.

At this, the creation of an operational pilot implementation is instrumental to provide a better and practical understanding of the proposed format that may work as a basis to the creation of a true ontology for this domain, an important and desirable outcome for this project.

It is important to highlight that the pilot implementation employed header files to define all naming conventions, any change on the used ontology may be easily deployed at an updated version.

One major drawback observed on the technological aspect of the development was the use of two distinct environments, one based on gnu C++11 and the other on Python 3.6. Although the use of C++ allowed for the creation of a very efficient solution to the file format conversion and emission detection problems, the required change of mindset whenever the development needs to move from one environment to the other resulted in a relevant and perceptible decrease in work efficiency, most likely resulting in worst code

⁵³ <https://www.ansible.com>

⁵⁴ <https://www.rudder.io/en/discover/what-is-rudder/>

⁵⁵ <https://www.saltstack.com/>



quality than would have been achieved if a single development environment was used. Even worse, the diverse development environment resulted in an increased risk of non-acceptance since maintenance costs are also higher.

Such drawbacks may outweigh the benefits of the use of C++ for transcoding of the binary files and the emission detection task. Later developments, after a full system deployment is achieved, may revisit these C++ components and recode them into Python, that has greater support among the engineers within [Anatel](#) and greater flexibility on the available analytical tools.

One important aspect is that the perceived performance issues that had driven the decision to move the development to gnu C++11 may be overcome by scaling the processing over a cloud platform or using other enhanced processing tools, enabling the use of Python even though it may be less efficient than the equivalent C++ code.

Reviewing the implemented modules, the used detection algorithm was able to produce the desired results, it detected emissions with no information about the noise floor or channel assignation, using as input only the expected standard deviation for the noise and some exclusions, such as minimum and maximum acceptable channel width.

One strong limitation of the implemented detection algorithm is that it is not linked with an external database where the parameters may be stored and changed as needed. Those parameters are defined as constants within the header and code files. The required change from constants to a database connection is trivial but will require a new development cycle.

The connection of the detection module to a database, most importantly, will allow the system to store detected channels and reuse these detections, such that there will be less variation on the channel boundaries between files, allowing for better channel segmentation and thus better operation of the remaining analytical modules.

Another strong limitation on the pilot implementation is the limited amount of sample data used to conduct the trials. This limitation may result in a possible restriction on the use of the existing code at different conditions.

Although more extensive trials could be performed, one important aspect to consider is that, as discussed, the full system integration with a central repository is expected to provide a significant increase in the flexibility and adaptability of the existing algorithm, an increase that may be sufficient to solve any issue associated with detection of odd emissions that may not easily be identified by the existing algorithm on the current implementation.

Considering the identification of discords in the self-similarity of the emission, the use of the standard matrix-profile distance over the flattened spectrogram proved to be less than satisfactory, although effective in several cases. As previously presented at the “Results”, the main difficulty is associated with the performance when the number of traces becomes exceptionally large. This is a strong limitation for the processing of a dataset that is continuously accumulating data.

At the current implementation, the alternative to solve this problem is to segment the dataset into meaningful segments, e.g. one dataset every 15 minutes. This approach

has the disadvantage that it makes it harder to correlate discords between different time periods and to really provide an understanding of the nature of the discord.

Another alternative to solve this limitation is to employ different distance metrics, such as the one used for clustering. Applying a clustering algorithm to group similar emissions within the same channel may enable to create template models for the most recurrent pattern and for the most relevant discords. Each template could then be used as a reference for the analysis performed over the incoming data, i.e. not only signalling the discords within a single time block as currently implemented but allowing for a meaningful tagging of the emission throughout the complete dataset.

The use of the developed clustering algorithm for the self-similarity analysis instead of the matrix-profile distance would allow for the creation of a more realistic representation of the emission types within a channel and thus for better operation of the clustering algorithm between several channels. To better understand this statement, we need to review the current implementation of the clustering algorithm.

The clustering analysis is performed based on the distance between the average trace for the emissions within each channel. This assumes that the mean power level within each bin is an acceptable representation of the power level of that channel, something that, for example, holds true to the emissions presented in Figure 19 but fails to describe the emissions presented in Figure 20.

This limitation on the distance measure could be solved by the change on the self-similarity analysis as proposed, which would result in a description of the channel presented in Figure 20 as being used by three distinct emissions and a more reliable clustering of each of these emissions.

Furthermore, the change on the self-similarity analysis could allow for the development of a more meaningful analysis of the temporal behaviour of the emissions, allowing for the creation of other distance metrics and visualizations.

As an example of an analysis based on the temporal behaviour, one could associate each emission type defined by a cluster with an identifier, e.g. a single character. This association process should include a similar representation for the time periods when there is no activity on the channel. With this, the sequence of silence and emissions within a channel may be translated into a string array. The time behaviour of different channels may then be compared and channels grouped into clusters using a metric such as the Levenshtein distance [85].

Considering the performance obtained on the clustering module, the efficient processing of the emission traces might become an issue if such analysis is performed in a systematic scope as proposed.

To this end, is worth mentioning that, for the used dataset, the results obtained by the individual comparison of all frequency bins on the power level distribution within each channel did not bring about relevant benefits to the clustering results when compared with a simpler approach employing only a few attributes that might be easily computed from the traces, such as the power variance within band, rise and fall attack rates, or several $\beta\%$ occupied bandwidth.



Such an approach, using clearly defined attributes, have the benefit of being more transparent and easier to translate into concepts that the users may understand and use on their work.

More trials should be performed using different strategies in an attempt to ensure that the best alternative is used in each case. One may even propose a mixed approach, where a high-level clustering is based solely on a few essential parameters and a fine-grained clustering might be created around black-box approaches, such as the one used or employing neural networks.

It becomes clear at this point that the pilot implementation was a valuable step on the system development, but only the surface of this problem was scratched at this point. More development is needed, especially the integration of the various modules into an interface that allows the user to interact with the data and provide the needed feedback for complete system development, a component that was left on the first experimental stages.

8 Conclusions

Considering the volume of raw data produced by the spectrum monitoring network and the fact that only a fraction of this data is relevant in itself, is paramount the use of automation and signal processing tools to gather, analyse, index and consolidate data into its meaningful content, allowing for efficient use of measurement, IT and personnel resources.

This document presented a system architecture to address this problem by the management of spectrum monitoring data acquisition, storage, analysis and retrieval. The proposed architecture envisions the integration with existing spectrum monitoring networks using available interfaces and the exchange of data using file formats native to each equipment manufacturer.

Automation of this integration should employ existing applications, e.g. Ansible⁵⁶, Rudder⁵⁷, SaltStack⁵⁸, etc., requiring the creation of specific modules for these interfaces using the documentation provided by the various manufacturers.

The deployment of an automation solution is expected to allow more secure and efficient management of the spectrum monitoring network and most importantly, the execution of automatically generated measurement scripts that allow the optimization of the essential acquisition parameters.

Concerning the measurement data repository, the proposed solution is expected to employ files created with a standard data structure using HDF5 format. The proposed structure attempts to harmonize various ITU standards related to spectrum monitoring and existing ontologies related to the measurement and use of the radio spectrum.

Concerning the data management, it was also proposed a relational database structure to enable the indexing of the HDF5 files and associated metadata that may be useful for data retrieval, including data acquired from different sources and services, e.g. spectrum management information such as the [assignment](#), [allocation](#) and [allotment](#) of spectrum bands and channels.

To enable the increase of the analytical level of the spectrum monitoring network in comparison to existing solutions, the architecture proposes the creation of modules to: enable the automatic detection of emission using a Bayesian energy detection method; identify discords between emissions detected within a single channel using the matrix-profile distance; and create clusters to group similar emissions using RMSD as a distance measure between the power level trace that describes each emission.

A pilot project partially implemented the proposed architecture employing as reference the spectrum monitoring network in use by Brazilian National

⁵⁶ <https://www.ansible.com>

⁵⁷ <https://www.rudder.io/en/discover/what-is-rudder/>

⁵⁸ <https://www.saltstack.com/>



Telecommunications Agency, [Anatel](#). The development effort concentrated on the creation of the required data structures and the described analytical modules.

With the pilot implementation was possible to identify critical aspects of the proposed architecture, enabling to develop a better understanding of the problem that is crucial for future development cycles of this information management system.

8.1 Theoretical implications

This project was initially proposed in the field of innovation, as an attempt to employ an array of existing technologies into a new application. As such, no relevant theoretical implications were expected.

Nevertheless, during the initial development, it became clear that the domain of spectrum management lacks a comprehensive ontology. Even more alarming was the fact that existing standards for data exchange were employing different terminology for similar concepts and that there was no coherent definition for classes, attributes and their relations.

The design of a new data structure using HDF5 and underlying domain ontology was crucial to the development of the proposed solution but, more importantly, it may serve as the foundation for a new standard for the digital exchange of spectrum monitoring data and an ontology for the spectrum management domain.

This standardization should be carried out within a forum such as the International Telecommunication Union, ITU, and its eventual adoption by various participating manufactures may become a key element in the development of more efficient alternatives for spectrum management, enabling the efficient, automatic and seamless integration of different solutions.

Additionally, such new standards may be used to enable open data initiatives related to spectrum monitoring, allowing the reuse of the collected information and consequent value generation from this unexplored public resource.

8.2 Practical implications

The current project was constructed around the data management problem affecting [Anatel](#)'s spectrum monitoring network and the most immediate implication is the possible use of the developed concepts and implemented codes to solve this problem.

The deployment of the proposed solution on a production scenario will require further development of the code already developed, but the essential initial steps were already taken.

If a successful implementation of the system core is achieved, further steps may be taken to enable further development of the concepts here presented, including the publication of the monitoring information at open data portals and the promotion of new standards for radio spectrum measurement data, with the engagement of partners that may share the benefits and costs of system maintenance.

8.3 Limitations and further research

The implementation of the proposed system architecture is extraordinarily complex since it demands a combination of signal processing, data management and IT skills. Due to this complexity and the limited time and resources available to the current project, only the foundational aspects to the complete system development could be tackled.

Although limited, the achieved results are promising. With additional development cycles and more importantly, greater interaction with users to aid in the development of the WebGUI, it will be possible to deploy into production a functional system to manage the existing spectrum monitoring data repository.

Future development may also consider other alternatives, more importantly, the use of time capture data, an alternative that was not explored in the pilot implementation. This may allow for the development of important features, such as below noise floor detection of coherent signals, and advanced signal identification and measurements with modulation analysis.

Other aspects that must be considered as soon as possible on the development of the proposed system architecture is the integration of an automation framework. This will allow greater efficiency on the use of the spectrum monitoring network and better employment of the available human resources.

The deployment of a complete and functional system will provide additional experience on the manipulation of the proposed HDF5 data structure and associated ontology, building the hands-on knowledge necessary to support the proposition and discussion about the creation of a standard ontology and digital exchange format within the ITU community.

The creation of such standards encompasses more than the ITU community and will also require the engagement of the engineering and scientific community through articles and conferences. The engagement of these groups is exceedingly important since they will take great benefits by the adoption of such standards and the creation of open data repositories containing measurement information about the radio spectrum.



9 References

- [1] ETSINF, ‘Guía: Estructura y Contenidos Recomendados’. Escola Tècnica Superior d’Enginyeria Informàtica, ETSINF, València, 2019.
- [2] A. Downer, ‘Hierarchy of Needs Pyramid Parodies’, 2017. [Online]. Available: <https://knowyourmeme.com/memes/hierarchy-of-needs-pyramid-parodies>. [Accessed: 08-Aug-2019].
- [3] L. Sachs and B. Aiello, *Agile Application Lifecycle Management: Using DevOps to Drive Process Improvement*. Addison-Wesley Professional, 2016.
- [4] J. Sutherland, *Scrum: the art of doing twice the work in half the time*. Currency, 2014.
- [5] ‘Best Practices for Microservices’, *MuleSoft Whitepaper*. 2004.
- [6] V. F. Pacheco, *Microservice Patterns and Best Practices: Explore patterns like CQRS and event sourcing to create scalable, maintainable, and testable microservices*. Packt Publishing, 2018.
- [7] M. Cohn, *User stories applied: For agile software development*. Addison-Wesley Professional, 2004.
- [8] S. M. Baby and M. James, ‘A Comparative Study on Various Spectrum Sharing Techniques’, *Procedia Technol.*, no. 25, pp. 613–620, 2016.
- [9] W. K. Jones, ‘Use and Regulation of the Radio Spectrum : Report on a Conference’, *Washingt. Univ. Law Rev. Commun. Futur.*, vol. 1968, no. 1, 1968.
- [10] ‘Overview of ITU’s History’, *ITU.int*, 2018. [Online]. Available: <https://www.itu.int/en/history/Pages/ITUsHistory.aspx>. [Accessed: 23-Sep-2018].
- [11] ‘National Telecommunication Agencies’, *ITU.int*, 2018. [Online]. Available: <https://www.itu.int/en/ITU-D/Statistics/Pages/links/nta.aspx>. [Accessed: 14-Sep-2018].
- [12] ‘Regional Telecommunication Organizations’, *ITU.int*, 2018. [Online]. Available: <https://www.itu.int/en/council/Pages/rto.aspx>. [Accessed: 14-Sep-2018].
- [13] H. Bakker, ‘Key principles of market regulation in telecommunications’, in *ITU Regional Workshop on “Competition in Telecommunications Market”*, 2016.
- [14] R. Trautmann *et al.*, *Spectrum Monitoring Handbook*, 2011th ed. Geneva: ITU.int, 2011.
- [15] J. Rivera and R. van der Meulen, ‘Gartner Says Advanced Analytics Is a Top Business Priority’, *gartner.com*, 2014. [Online]. Available: <https://www.gartner.com/newsroom/id/2881218>. [Accessed: 26-Oct-2018].
- [16] J. O. M. Andres, ‘Business Analytics Class Notes, Information and Knowledge’, València, 2018.

- [17] J. Manco-Vásquez, M. Lázaro-Gredilla, D. Ramírez, J. Vía, and I. Santamaría, ‘A Bayesian approach for adaptive multiantenna sensing in cognitive radio networks’, *Signal Processing*, vol. 96, no. PART B, pp. 228–240, 2014.
- [18] J. Talukdar, B. Mehta, K. Aggrawal, and M. Kamani, ‘Implementation of SNR estimation based energy detection on USRP and GNU radio for cognitive radio networks’, *Proc. 2017 Int. Conf. Wirel. Commun. Signal Process. Networking, WiSPNET 2017*, vol. 2018-Janua, pp. 304–308, 2018.
- [19] V. Valenta, R. Marsalek, G. Baudoin, M. Villegas, M. Suarez, and F. Robert, ‘Survey on Spectrum Utilization in Europe: Measurements, Analyses and Observations’, pp. 6–10, 2010.
- [20] ‘RFeye Logger User Guide’, CRFS Limited, Cambridge, UK, CR-001205-UG-6, 2018.
- [21] ‘RFeye Node File Storage Specification’, CRFS Limited, Cambridge, UK, CR-002421-TN-1, 2018.
- [22] *Instrução Normativa Nº 4 de 12 de abril de 2012*. SLTI, MP, Brasil.
- [23] *Decreto Nº 8.777, de 11 de maio de 2016*. Casa Civil, Brasil.
- [24] *Lei Nº 12.527, de 18 de novembro de 2011*. Casa Civil, Brasil.
- [25] *Lei Nº 13.709, de 14 de agosto de 2018*. Casa Civil, Brasil.
- [26] *Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)*. European Parliament and of the Council, 2016.
- [27] ‘The MIT License’, *Massachusetts Institute of Technology*. [Online]. Available: <https://opensource.org/licenses/MIT>. [Accessed: 10-Jul-2019].
- [28] *Padrões de Interoperabilidade de Governo Eletrônico*. SLTI, MP, Brasil, 2017.
- [29] ITU-R SM.2039-0, ‘Spectrum monitoring evolution’, *International Telecommunication Union*. ITU.int, Geneva, 2013.
- [30] ITU-R SM.2355-0, ‘Spectrum monitoring evolution’, *International Telecommunication Union*. ITU.int, Geneva, 2015.
- [31] M. Knott, ‘Machine Learning and RF Spectrum Intelligence Gathering’. CRFS Whitepaper, 2017.
- [32] B. Boiko, *Content Management Bible*, 2nd ed. Wiley Publishing, 2004.
- [33] E. Keathley, *Digital Asset Management: Content Architectures, Project Management, and Creating Order out of Media Chaos*, 1st ed. Berkely, CA, USA: Apress, 2014.
- [34] Joint Committee for Guides in Metrology (JCGM), ‘International vocabulary of metrology — Basic and general concepts and associated terms (VIM)’, *International Organization for Standardization*, no. 3. 2008.
- [35] Joint Committee for Guides in Metrology (JCGM), ‘Evaluation of measurement data — Guide to the expression of uncertainty in measurement’, *International Organization for Standardization*. 2008.

- [36] A. Pfeiffer, I. Bausch-Gall, and M. Otter, ‘Proposal for a Standard Time Series File Format in HDF5’, *Proc. 9th Int. Model. Conf. Sept. 3-5, 2012, Munich, Ger.*, vol. 76, pp. 495–506, 2012.
- [37] S.-A. Dragly *et al.*, ‘Experimental Directory Structure (Exdir): An Alternative to HDF5 Without Introducing a New File Format’, *Front. Neuroinform.*, vol. 12, no. April, pp. 1–13, 2018.
- [38] P. Greenfield, M. Droettboom, and E. Bray, ‘ASDF: A new data format for astronomy’, *Astron. Comput.*, 2015.
- [39] J. L. Teeters *et al.*, ‘Neurodata Without Borders: Creating a Common Data Format for Neurophysiology’, *Neuron*. 2015.
- [40] G. J. Smethells, ‘PDB and HDF5 Compared’, 2005. [Online]. Available: <https://wci.llnl.gov/codes/pact/benchmarks.html>. [Accessed: 25-Jan-2019].
- [41] ITU-R SM.2117-0, ‘Data format definition for exchanging stored I/Q data for the purpose of spectrum monitoring SM Series’, *International Telecommunication Union*. ITU.int, Geneva, 2018.
- [42] L. A. Wasser and E. Webb, ‘Introduction to Hierarchical Data Format (HDF5) - Using HDFView and R’, *neonscience.org*, 2018. [Online]. Available: <https://www.neonscience.org/about-hdf5>. [Accessed: 25-Jan-2019].
- [43] A. Collette, *Python and HDF5: Unlocking Scientific Data*. O'Reilly Media, Inc., 2013.
- [44] ITU-R SM.1809, ‘Standard data exchange format for frequency band registrations and measurements at monitoring stations’, *International Telecommunication Union*. ITU.int, Geneva, Apr-2007.
- [45] ITU-R SM.668-1, ‘Electronic exchange of information for spectrum management purposes’, *International Telecommunication Union*. ITU.int, Geneva, 1997.
- [46] ITU-R SM.1393, ‘Common formats for the exchange of information between monitoring stations SM Series Spectrum management’, *International Telecommunication Union*, ITU.int, Geneva, 1999.
- [47] ITU-T Y.4500.12, ‘oneM2M base ontology’, *International Telecommunication Union*. ITU.int, Geneva, 2018.
- [48] ITU-T X.1500, ‘Overview of cybersecurity information exchange’, *International Telecommunication Union*. ITU.int, Geneva, 2011.
- [49] H. Rijgersberg, M. van Assema, and J. Top, ‘Ontology of Units of Measure and Related Concepts’, *Semant. Web* 4, no. 1, pp. 3–13, 2013.
- [50] Q. Zhou, A. J. G. Gray, and S. McLaughlin, ‘ToCo: An ontology for representing hybrid telecommunication networks’, in *16th Extended Semantic Web Conference, Portoroz, Slovenia*, 2019.
- [51] T. Cooklev, ‘Making Software-defined Networks Semantic’, *2015 12th Int. Jt. Conf. E-bus. Telecommun.*, vol. 06, pp. 48–52, 2015.
- [52] R. B. Normoyle, ‘Overview of the joint open architecture spectrum infrastructure (JOASI) ontology for spectrum interoperability’, *Proc. - IEEE Mil. Commun. Conf. MILCOM*, pp. 1774–1778, 2013.



- [53] L. S. Todor Cooklev, ‘A Comprehensive and Hierarchical Ontology for Wireless Systems’, in *Wireless World Research Forum Meeting 32*, 2014, p. 5.
- [54] S. Li, M. M. Kokar, and D. Brady, ‘Developing an Ontology for the Cognitive Radio: Issues and Decision’, in *SDR ’09 Technical Conference and product Exposition*, 2009.
- [55] Y. Chen, M. M. Kokar, J. J. Moskal, and D. Suresh, ‘Mapping spectrum consumption models to cognitive radio ontology for automatic inference’, *Analog Integr. Circuits Signal Process.*, no. June 2018, pp. 1–13, 2017.
- [56] D. Smirnov and P. Stuetz, ‘Ontology-Based Modelling of Sensor and Data Processing Ressources Using OWL: A Proof of Concept’, *First Int. Conf. Appl. Syst. Vis. Paradig.*, 2016.
- [57] S. Kianoush, M. Raja, S. Savazzi, and S. Sigg, ‘A Cloud-IoT Platform for Passive Radio Sensing: Challenges and Application Case Studies’, *IEEE Internet Things J.*, vol. 5, no. 5, pp. 3624–3636, 2018.
- [58] L. Shi, P. Bahl, and D. Katabi, ‘Beyond Sensing: Multi-GHz Realtime Spectrum Analytics’, *12th {USENIX} Symp. Networked Syst. Des. Implement.*, vol. 1, pp. 20–29, 2015.
- [59] L. N. T. Perera and H. M. V. R. Herath, ‘Review of spectrum sensing in cognitive radio’, *2011 6th Int. Conf. Ind. Inf. Syst. ICIIS 2011 - Conf. Proc.*, pp. 7–12, 2011.
- [60] K. Moessner, B. G. Evans, and X. Liu, ‘Comparison of reliability, delay and complexity for standalone cognitive radio spectrum sensing schemes’, *IET Commun.*, vol. 7, no. 9, pp. 799–807, 2013.
- [61] J. Chan, A. Wang, A. Krishnamurthy, and S. Gollakota, ‘DeepSense: Enabling Carrier Sense in Low-Power Wide Area Networks Using Deep Learning’, pp. 25–27, 2019.
- [62] A. Selim, F. Paisana, J. A. Arokiam, Y. Zhang, L. Doyle, and L. A. DaSilva, ‘Spectrum Monitoring for Radar Bands Using Deep Convolutional Neural Networks’, *2017 IEEE Glob. Commun. Conf. GLOBECOM 2017 - Proc.*, vol. 2018-Janua, pp. 1–6, 2018.
- [63] A. Fehske, J. Gaeddert, and J. H. Reed, ‘A new approach to signal classification using spectral correlation and neural networks’, *2005 1st IEEE Int. Symp. New Front. Dyn. Spectr. Access Networks, DySPAN 2005*, pp. 144–150, 2005.
- [64] R. P. Adams and D. J. C. MacKay, ‘Bayesian Online Changepoint Detection’, *arXiv Prepr. arXiv0710.3742*, 2007.
- [65] A. Nielsen, *Practical Time Series Analysis*. O’Reilly Media, Inc., 2019.
- [66] R. Hyndman, X. Wang, and K. Smith, ‘Characteristic-based clustering for time series Data’, *Data Min. Knowl. Discov.*, vol. 13, pp. 457–476, 2005.
- [67] A. W. C. Fu, E. Keogh, L. Y. H. Lau, C. A. Ratanamahatana, and R. C. W. Wong, ‘Scaling and time warping in time series querying’, *VLDB J.*, vol. 17, no. 4, pp. 899–921, 2008.
- [68] S. Gharghabi, S. Imani, A. Bagnall, A. Darvishzadeh, and E. Keogh, ‘An Ultra-Fast Time Series Distance Measure to allow Data Mining in more Complex Real-World Deployments’, *IEEE Int. Conf. Data Min.*, 2018.

- [69] S. Gharghabi, S. Imani, A. Bagnall, A. Darvishzadeh, and E. Keogh, ‘Matrix Profile XII: MPdist: A Novel Time Series Distance Measure to Allow Data Mining in More Challenging Scenarios’, *Proc. - IEEE Int. Conf. Data Mining, ICDM*, no. January, pp. 965–970, 2019.
- [70] S. R. Fleurke, H. G. Dehling, H. K. Leonhard, A. D. Brinkerink, and R. Den Besten, ‘Measurement and statistical analysis of spectrum occupancy’, *Eur. Trans. Telecommun.*, vol. 15, no. 5, pp. 429–436, 2004.
- [71] ITU-R SM.2028, ‘Monte-Carlo Simulation methodology for the use in sharing and compatibility studies between different radio services or systems’, *International Telecommunication Union*. ITU.int, Geneva, 2017.
- [72] M. Höyhtyä *et al.*, ‘Spectrum Occupancy Measurements: A Survey and Use of Interference Maps’, *IEEE Commun. Surv. Tutorials*, vol. 18, no. 4, pp. 2386–2414, 2016.
- [73] C. Rossant, ‘Should you use HDF5 ?’, *cyrille.rossant.net*, 2016. [Online]. Available: <https://cyrille.rossant.net/should-you-use-hdf5/>. [Accessed: 08-Feb-2019].
- [74] C. Rossant, ‘Moving away from HDF5’, *cyrille.rossant.net*, 2016. [Online]. Available: <https://cyrille.rossant.net/moving-away-hdf5/>. [Accessed: 08-Feb-2019].
- [75] K. Hinsen, ‘Konrad Hinsen ’ s Blog On HDF5 and the future of data’, *blog.khinsen.net*, 2016. [Online]. Available: <http://blog.khinsen.net/posts/2016/01/07/on-hdf5-and-the-future-of-data-management/>. [Accessed: 08-Feb-2019].
- [76] ‘HDF5 User ’ s Guide’, 2016. [Online]. Available: <https://portal.hdfgroup.org/display/HDF5/HDF5+User%27s+Guide>. [Accessed: 08-Feb-2019].
- [77] V. Boddapati, A. Petef, J. Rasmusson, and L. Lundberg, ‘Classifying environmental sounds using image recognition networks’, *Procedia Comput. Sci.*, vol. 112, pp. 2048–2056, 2017.
- [78] S. Rajendran, W. Meert, D. Giustiniano, V. Lenders, and S. Pollin, ‘Distributed Deep Learning Models for Wireless Signal Classification with Low-Cost Spectrum Sensors’, pp. 1–13, 2017.
- [79] I. F. Akyildiz, W. Y. Lee, M. C. Vuran, and S. Mohanty, ‘NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey’, *Comput. Networks*, 2006.
- [80] B. F. Lo, ‘A survey of common control channel design in cognitive radio networks’, *Physical Communication*. 2011.
- [81] E. Hossain, D. Niyato, and D. I. Kim, ‘Evolution and future trends of research in cognitive radio: a contemporary survey’, *Wirel. Commun. Mob. Comput.*, no. October 2013, pp. 1530–1564, 2012.
- [82] V. Prithiviraj and L. Hanumath Bhujanga Rao, ‘A novel energy detection technique for cooperative spectrum sensing in cognitive radio’, *J. Green Eng.*, vol. 2, no. 2, pp. 179–186, 2012.



- [83] F. Salahdine, ‘Spectrum Sensing Techniques For Cognitive Radio Networks’, *arXiv Prepr. arXiv1710.02668*, pp. 1–20, 2017.
- [84] D. Cabric, A. Tkachenko, and R. W. Brodersen, ‘Spectrum Sensing Measurements of Pilot , Energy , and Collaborative Detection’, *IEEE Milcom*, pp. 23–25, 2006.
- [85] G. Navarro, ‘A guided tour to approximate string matching’, *ACM Comput. Surv.*, vol. 33, no. 1, pp. 31–88, 2001.
- [86] WRC-15, *Radio Regulations Volume 1, Articles*, 2016th ed. Geneva: ITU.int, 2016.
- [87] ITU-R SM.1880-2, ‘Spectrum occupancy measurements and evaluation’, *International Telecommunication Union*. ITU.int, Geneva, 2017.

10 Annexes

On the following sections are included additional information that is ancillary to the understanding of this project, mostly in sections 10.1, 10.2, 10.7 and 10.8 and some that might be considered essential to a complete understanding of the presented work but was too complex or burdensome to a more general and high-level understanding, as intended for the main body of the text. This is the case of sections 10.3, 10.4, 10.5 and 10.6 that present the used models and produced algorithms and codes.

The order in which these annexes are presented is defined by the order in which they are first mentioned within the previous chapters.



RFEye Node

20-6

Intelligent Wideband Receiver



The original RFeye Node is still the benchmark for cost-effective real-time 24/7 ITU-compliant spectrum monitoring and radio geolocation.

The RFeye Node 20-6 is a complete spectrum monitoring system designed for remote deployment in distributed networks both indoors and outdoors, including in hostile environments. Packaged in a compact, rugged and a weatherproof housing, it has been optimized for size, weight and power (SWaP) and is simple to connect to power and network.

The Node's unique architecture is capable of supporting multiple concurrent tasks and missions, including ITU-compliant measurements. Timing and synchronization features allow correlation of data between multiple Nodes for accurate DF and geolocation of target signals using AOA, TDOA and POA techniques. The Node 20-6 is available with optional on-board SSD for logging of very large data sets.

⁵⁹ Reproduced from <https://pages.crfss.com/hubfs/datasheets/node-20-6-datasheet.pdf>.with license.

RFeye Node

20-6 Specifications

Single channel receiver		Processor sub-system
Switchable RF inputs	4 x SMA connectors	Marvell 88F6281 @ 1 GHz
Frequency		Main memory
Range	10 MHz to 6 GHz	512 MB DDR2
Noise figures at maximum sensitivity		System disk
10 MHz to 3 GHz	8 dB typical	512 MB
3 GHz to 6 GHz	11 dB typical	
Phase noise		I/O
Receiver input at 2 GHz	-91 dBc/Hz at 20 kHz offset, typ.	Network 1 x 1 GigE, with POnE
Signal analysis		Universal Serial Bus 2 x USB 2.0
Instantaneous bandwidth	20 MHz	2 x IEEE1394 expansion ports configurable as:
Tuning resolution	1 Hz	GPS antenna input 1 x SMA passive or active (3.3 VDC)
Internal frequency reference (pre-calibration)		Cellular modem antenna 1 x SMA
Initial accuracy	better than ±2 ppm typ.	Cellular modem (internal) LTE*/HSPA+ /GSM (MIMO not supported)
Stability	better than ±1 ppm typ.	* region variants, consult CRFS
Ageing	better than ±2 ppm per year	
Programmable sweep modes		Data storage
Sweep speed - fast synth	45 GHz/s @ 1.2 MHz RBW	External flash disk via USB interfaces
Sweep speed - high quality synth	18 GHz/s @ 1.2 MHz RBW	Optional internal storage 512 GB SSD option
User programmable modes	free run continuous, single timed, user trigger and adaptive	
Trigger-on-event modes	user defined masks, actions and alarms	System software
Sampling		Boot firmware U-Boot
Resolution	12 bits per channel (I&Q)	Operating system Linux, kernel v 2.6
Rate	40 MS/s I&Q	RFeye Node Control Protocol NCP Server (NCPd)
Third order intercept points with AGC		Node Apps (optional) Logger, Recorder, Threshold, Stations, Survey
< 1 GHz	+21 dBm typical	
1 GHz to 6 GHz	+22 dBm typical	
Local oscillator		Size, weight and power
Re-radiation	-90 dBm typical	Dimensions (w, h, d) 170 x 60 x 125 mm (6.7 x 2.4 x 4.9 inches)
Frequency references		without IP67 rated end plate
Selectable	Internal, GPS or external	Weight 1.4 kg (3.1 lbs)
External input	10 MHz ±1 kHz	with IP67 rated end plate or 2 kg (4.4 lbs)
Output	10 MHz	DC power or POne 10 to 48 VDC
Power consumption		
Typical	15 W	
Maximum	25 W	
Environmental		
Operating temperature	-30 to +55 °C (-22 to 131 °F)	
Storage temperature	-40 to +70 °C (-40 to 158 °F)	
Ingress protection	IP67 (with optional end plate)	



Cambridge RF Systems, Cambridge Research Park,
Building 7200, Beach Drive, Cambridge, CB25 9TL, UK
+44 1223 859 500 crfs.com

CRFS and RFeye are trademarks or registered trademarks of CRFS Limited. Copyright © 2017 CRFS Limited. All rights reserved. No part of this document may be reproduced or distributed in any manner without the prior written consent of CRFS. The information and statements provided in this document are for informational purposes only and are subject to change without notice. Document Number CR-000122-DS-12, July 2018.



FS 576625

10.2 Message template used on the market survey

From: Fabio Santos Lobao
Sent: DD Month 20AA HH:MM
To: <NAME>
Subject: Information request about how spectrum monitoring data is stored, analyzed and distributed

Dear <Mr./Mrs. NAME>

I am writing to ask for your help in providing some information about how you manage spectrum monitoring data at <ORGANIZATION NAME>. To give you some ideas of what exactly I am looking for I added some suggestive open questions at the end of this message and some additional information on the following paragraphs. All information provided will be kept anonymous unless you explicitly ask me otherwise.

Stepping back to give you some context, I am not sure if you remember me, but we met on some of the ITU-R, WP-1C meetings. I work for Anatel, the Brazilian telecommunications regulatory agency and usually participate in delegations representing Brazil.

Now, I am on a sabbatical absence from Anatel doing a master course on information management in Spain. To my dissertation, I am researching on tools to improve the information management of spectrum monitoring measurement data and would like to include a review on how different spectrum management authorities are handling the data generated by monitoring equipment.

Since this review is not really the core of my work, but just a quick and small qualitative survey to aid in the motivation arguments, I am just sending this message to a few people that I had contact during my years of contribution to ITU, this is why I am not using the international offices and ITU administrative infrastructure to conduct this survey.

Please feel free to forward this message to others that you believe might be able to contribute to this discussion and, just to have a target timeframe, try to answer it before DD/MM/20AA.

In case you need additional information, feel free to contact me through WhatsApp or Telegram by the number +NN NNN NN NN NN

Follows some suggestive questions that might help you to understand what type of information I am looking for. You do not need to answer them explicitly and you can just describe how spectrum monitoring personnel do their job, even just sending me presentations or document examples that might give me some ideas on how things are done at your office.



- For how long do you store the raw spectrum monitoring data collected? Is it just until a report is completed?
- How do you build the reports? Using a commercial text editor? Automatically by the spectrum monitoring software? Which applications do you use?
- How do you store the reports? Do you have any sort of a content management system or a digital asset management system? Documents are electronically published in any way or to any group?
- Measurement results stored only on the resulting reports (e.g. as graphical images and tables) or are they also stored in any easily reusable format? (e.g. in a relational database with associated metadata or as files accessible through a management system of any sort)
- Is the measurement data indexed or searchable through parameters such as frequency, time, location, measurement station, etc?
- Is measurement data related to an emitter associated with the license parameters or stored on the same system?
- Do you apply any sort of automatic analysis?
- Do you apply any sort of automatic optimization process to guide the measurement procedures?
- How spectrum monitoring data is used to aid spectrum management activities?
- How information is shared between spectrum management and spectrum monitoring departments?

Many thanks in advance and best regards

Fábio Santos Lobão

Anatel, Brazilian National Telecommunications Agency

Regulatory Enforcement Support Department

WhatsApp/Telegram +NN NNN.NN NN NN

www.anatel.gov.br

10.3 Detailed relational model

On this annex is presented a detailed description of the proposed relational model for the indexing of radio spectrum measurement information derived from spectrum monitoring.

To better understand the proposed model, it is important to understand the employed concept of tags. Tags are used as an alternative to a complete description of all objects or qualifiers that may be applicable to the data. This simplifies the relational model and the user interface. The lists of tags that may be used to qualify different entities on the model are kept on separate tables and presented on the bottom part of the diagram.

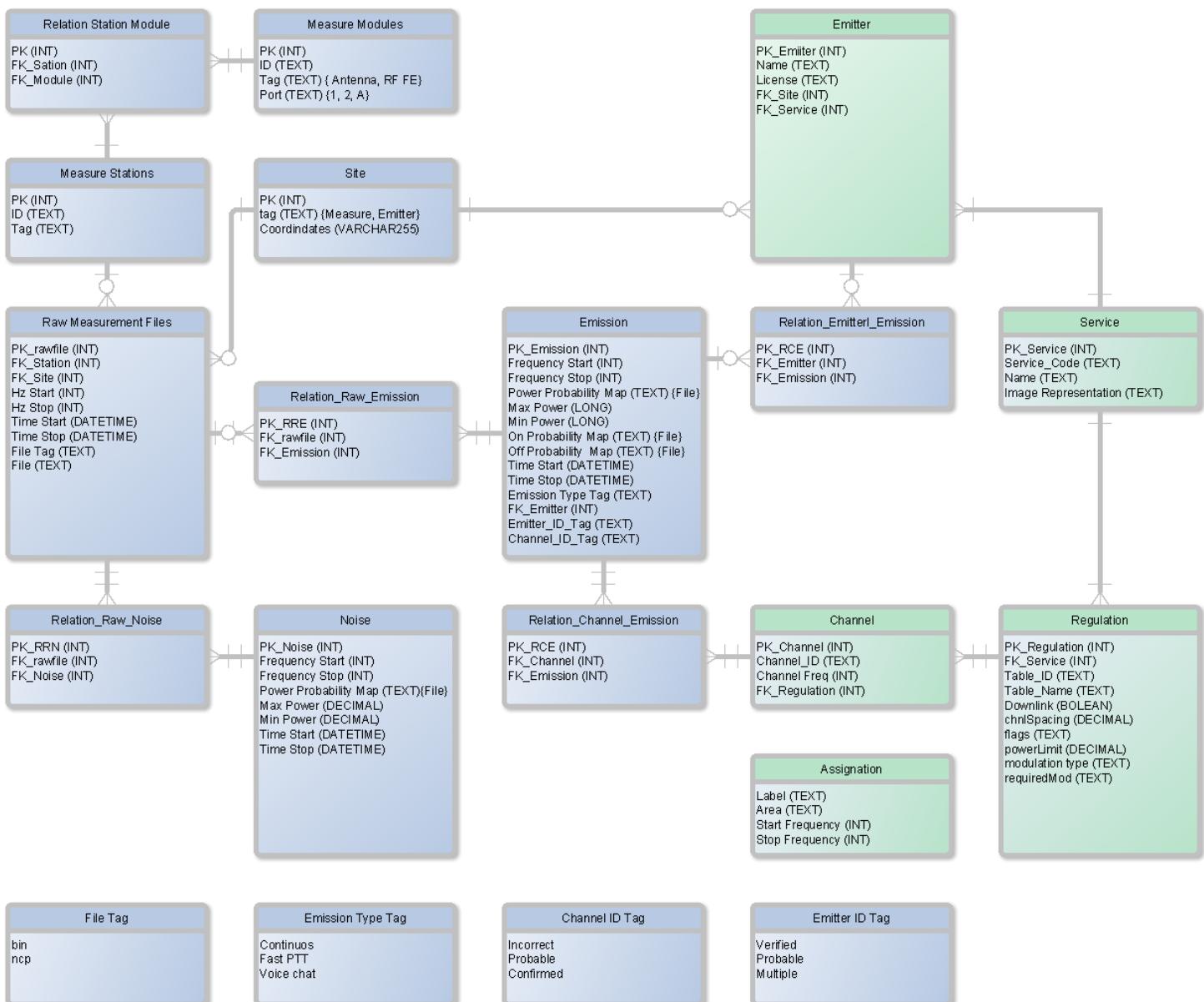


Figure 29. The relational model for the measurement index

10.4 Detailed HDF5 file structure description

This annex presents a detailed description of the proposed HDF5 file format.

The following figures present a tree view of the HDF5 file structure with all level depicted. Figure 30 presents the structure on the root group and the groups associated with raw data, including I/Q and spectrum sweep.

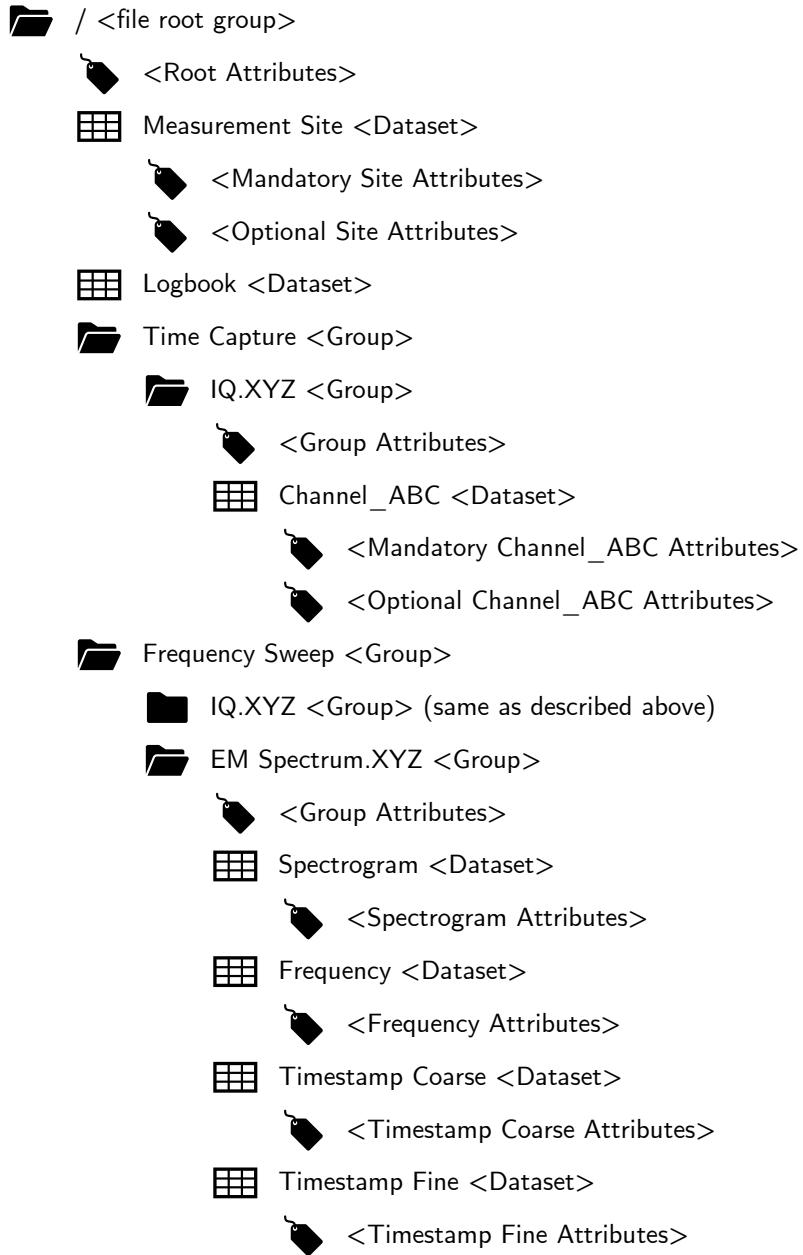


Figure 30. Tree view of the HDF5 file structure including root elements and raw data groups.

Figure 31 complement the previous view with additional groups associated with analytical data extracted from the raw measurement information. This includes noise and channel data groups.

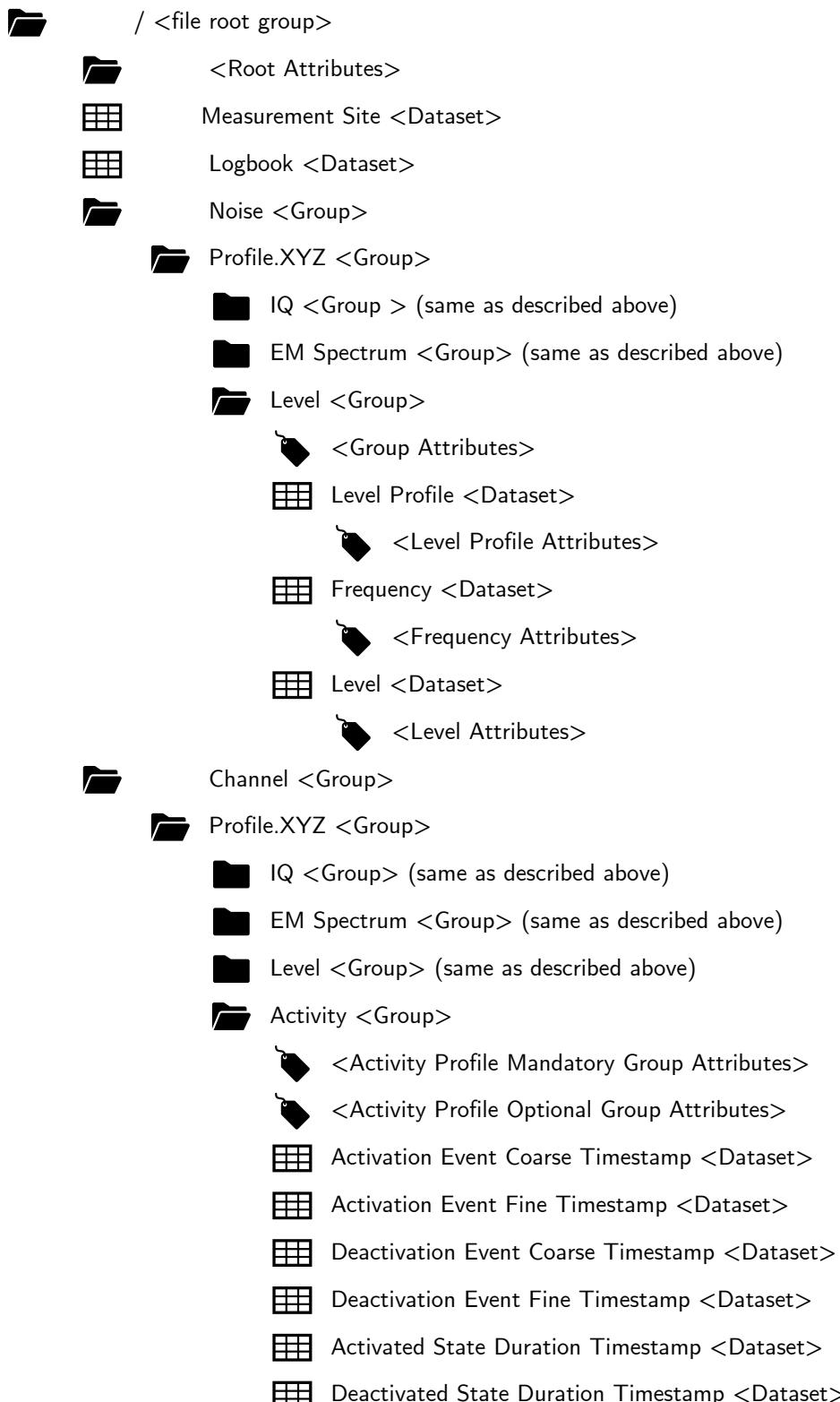


Figure 31. Tree view of the HDF5 file structure including root elements and analytical data groups.

A full detailed description of all attributes and datasets is presented on the following figures, where the following conventions are used:

- Rectangles delimited by dashed lines represent groups. The group name is presented on the top right corner including the full path, using a backslash to separate parent groups.
- Rectangles delimited by a continuous bold line represent datasets, the name of the dataset is presented at the top centre part of the rectangle.
- Rectangles delimited by a continuous thin line represent a set/list of attributes or characteristics. This is not an HDF5 object but only a presentation aid to present several objects in an ordered manner. The exact objective of the set or list is presented at the centre top part of the rectangle.
- Parentheses are used to delimit comments on the item, such as the standard measurement unit. They are not a part of the attribute or dataset.
- Chevrons (angled brackets) are used to present the data type description of an attribute or dataset.
- Braces are used to delimit examples of values for the attribute or dataset

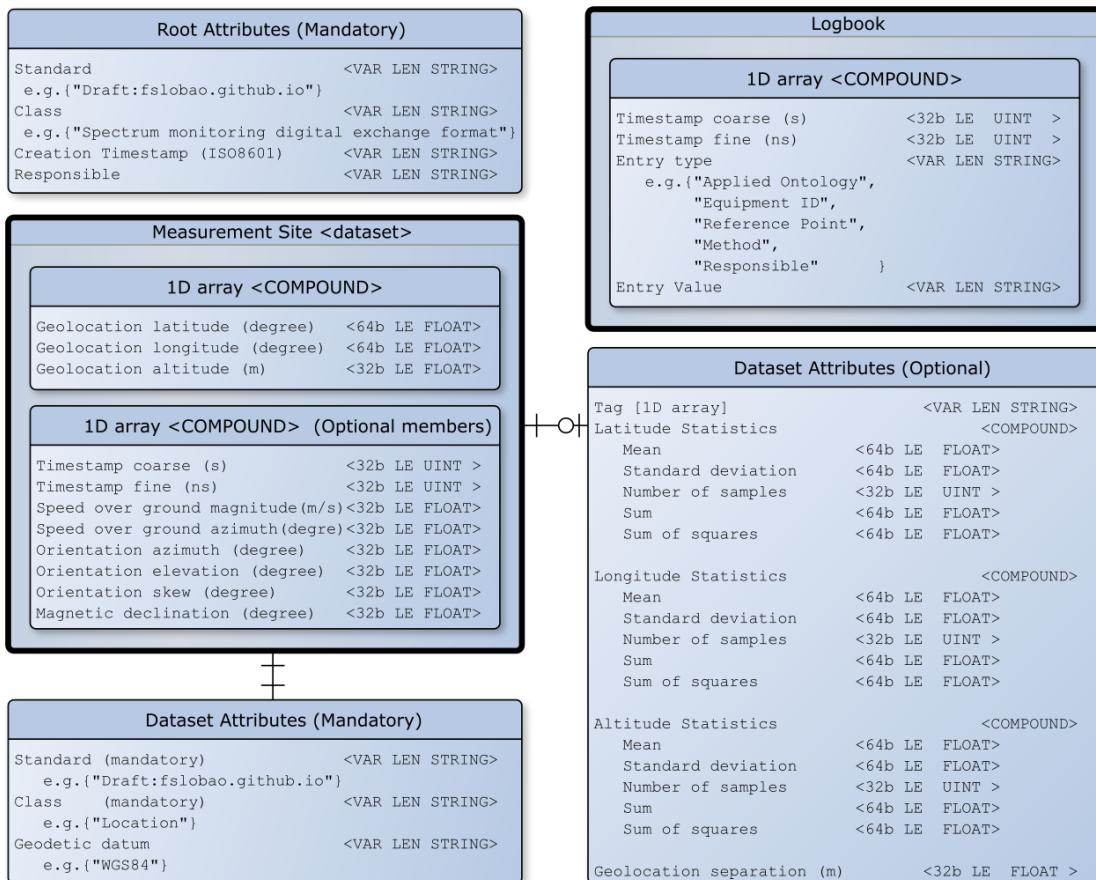


Figure 32. A detailed description of root objects on the proposed HDF5 format.



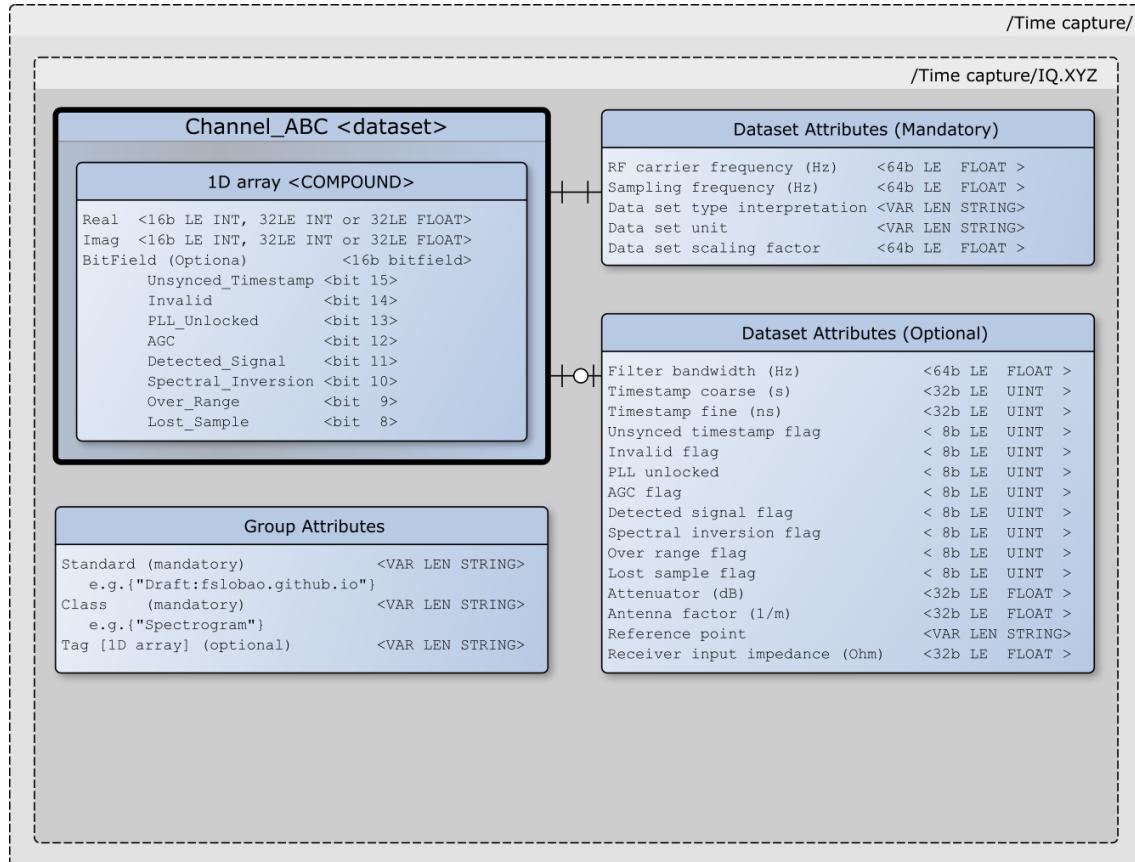


Figure 33. A detailed description of objects within the time capture group on the proposed HDF5 format.

Back to
section 5.4.2

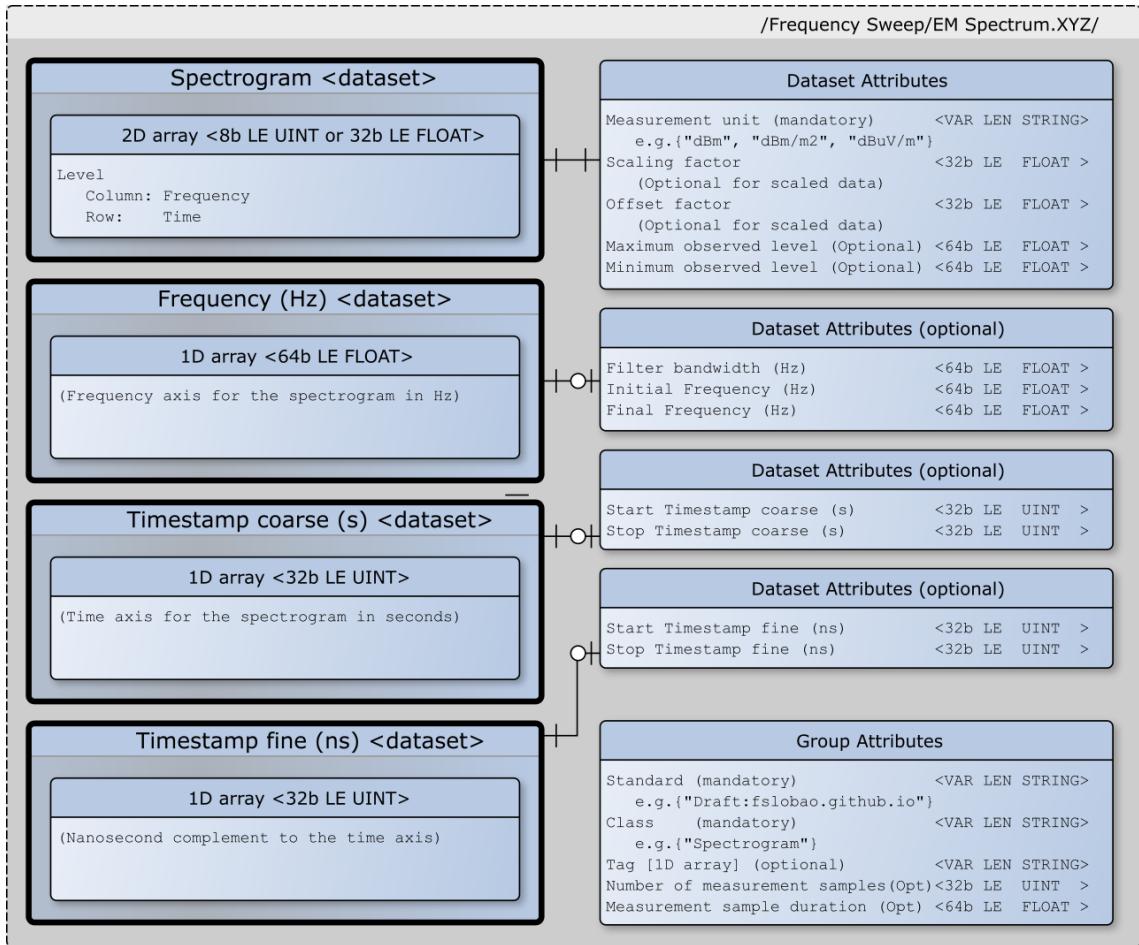


Figure 34. A detailed description of objects within the frequency sweep group on HDF5 format.

 Back to
section 5.4.2

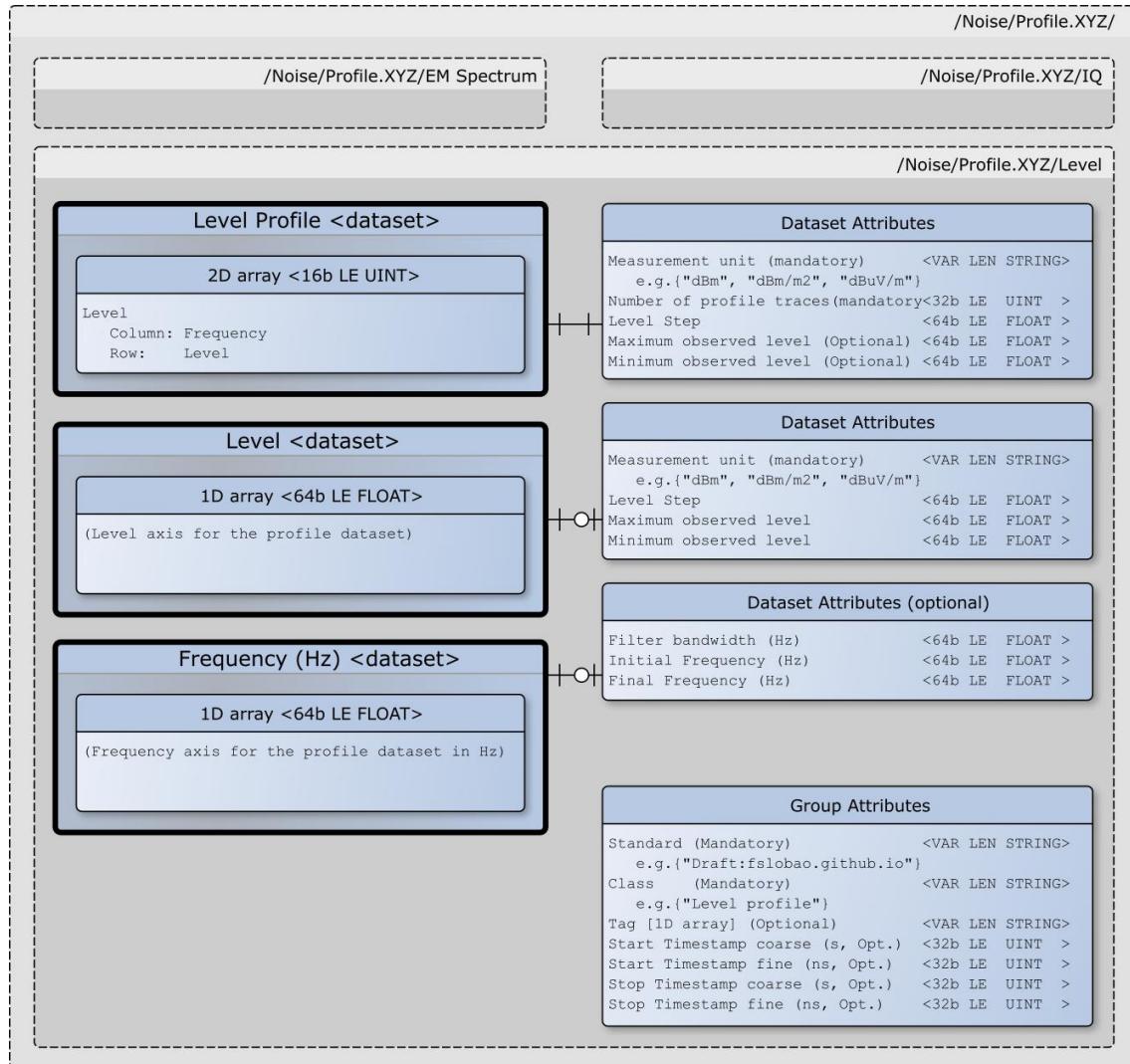


Figure 35. A detailed description of objects within the noise group on the proposed HDF5 format.

Back to section 5.4.2

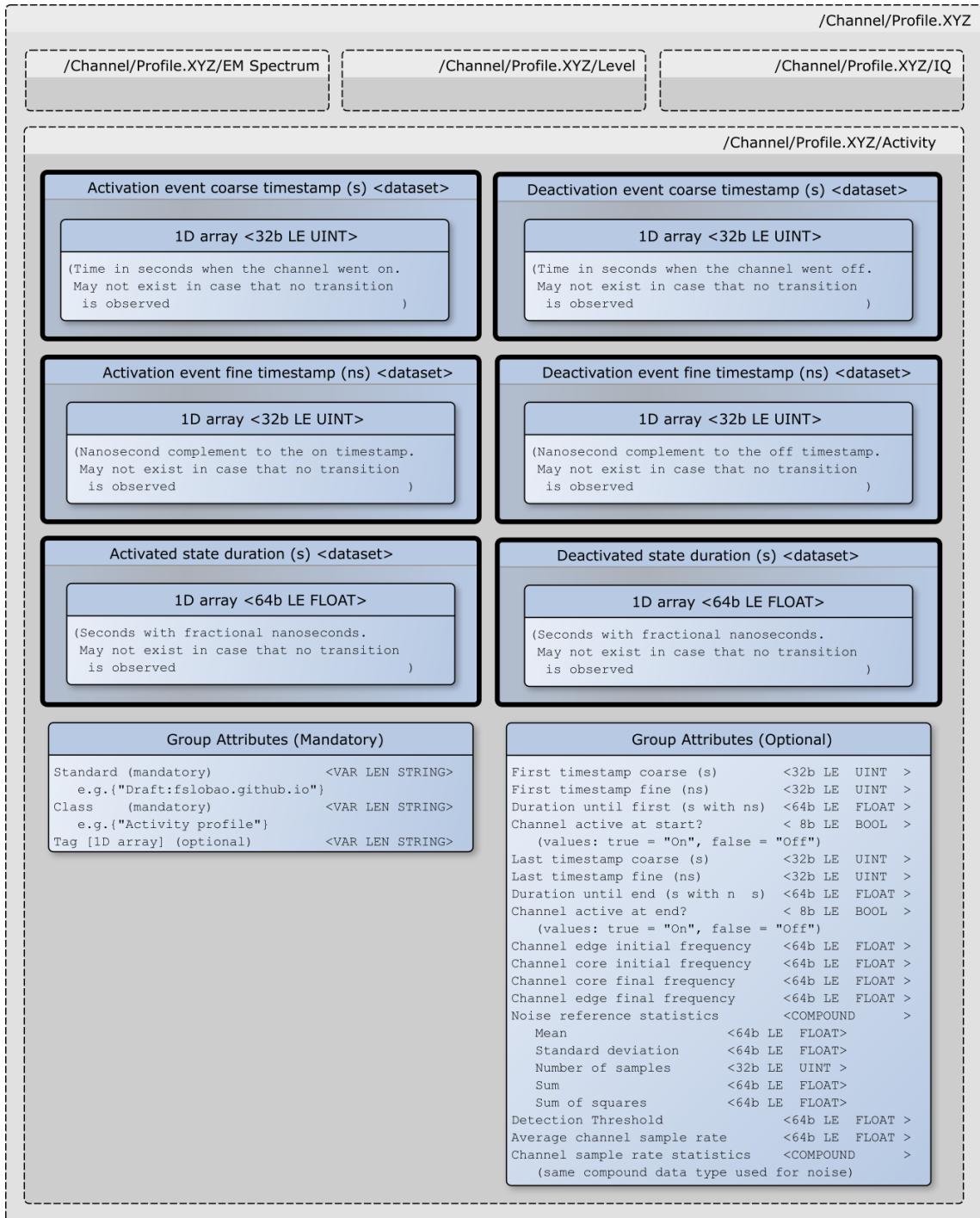


Figure 36. A detailed description of objects within the channel group on the proposed HDF5 format.

Back to
section 5.4.2

10.5 Code of the pilot implementation

On this annex is included an embedded object containing an electronic compressed copy of all code developed for this project.

This information is also available at a public repository⁶⁰ where additionally sample data is available.

To access the compressed data, click on the pin icon below. Depending on the used PDF reader application and operating system, the reader may be required to save the object on a local folder where it should be decompressed using a ZIP⁶¹ compatible program.

Some PDF reader applications may block the access to the attached file, requiring reconfiguration of security features or the use of different means to access the data.



⁶⁰ <https://github.com/FSLobao/Spectrum-Cortex>

⁶¹ <https://support.pkware.com/display/PKZIP/APPNOTE>

10.6 Description of analysis algorithms

This annex describes in greater detail the algorithm created for two of the analysis modules created as part of the pilot implementation.

10.6.1 Emission detection

The channel detection module is coded in C++ and run as an [online algorithm](#) along with the initial file format conversion by the “**decode.cpp**” program. The source code is presented in the file “**level_differential_detector.hpp**”.

The idea behind the automatic emission detection was discussed in the chapter about “State of the Art” and it is based on the idea of a Bayesian detector, i.e. detect emission by the variation it causes on statistical measurements, e.g. arithmetic mean, over a region of the spectrum.

The first step is to create a set of flags at each analysed trace using an online algorithm. This algorithm computes the difference between the arithmetic mean power level over adjacent bands. Whenever the absolute value of the difference in the mean power level is larger than a certain threshold a flag is created.

A truly Bayesian detector would ideally employ other measurements, such as the variance to define the detection threshold. At the present demonstration case, the use of a fixed value is acceptable since the power level variance of the noise is approximately constant over the considered bands, even though the noise mean value may change enough to make it difficult to use a fixed threshold level for the detection.

It is important to highlight that the use of the differential power instead of the absolute value of the power level, as observed on traditional energy detectors, allows for the easier definition of a threshold detection value. Such threshold is thus insensitive to variations in the background noise level that naturally occurs between bands and throughout time periods.

The arithmetic mean is an acceptable estimator for the noise power since the power distribution is gaussian in nature. Non-Gaussian noise might be observed in some use scenarios, but still, the used estimator is the most commonly employed in the reviewed literature, as presented in chapter 4.

The windows used to compute the arithmetic mean differential are not truly adjacent. A guard band is set between the windows with the objective to increase the peak resulting values, reducing the influence of the power-up and down ramps that naturally occur on the edge of any radio emission and strengthening the comparison between the noise and the peak level of the emission.

The exact frequency value for each flag is defined by the bin before the power level ramps up or after the power level ramps down. This is performed by identifying the negative overshoot of the spectrum trace before the positive and negative peaks on the differential result.



The power level between flags is computed by adding all bins between two flags, including from the start of the trace to the first flag and from the last flag to the end of the acquired trace data.

Each single spectrum trace is processed independently, resulting in a series of flags to each trace. These flags delimit regions identified as emissions and regions identified as noise.

The segment with the lowest total power within the trace is identified as noise and used as the noise reference level. All segments in which the total power does not exceed the detection threshold above the identified reference noise level are also identified as noise. Other segments are identified as emissions.

Unnecessary flags, such as those delimiting adjacent regions of the same type are deleted. At each flag deletion, the power density of the newly merged region is updated. At this point, other exclusions may be applied, including deleting regions that are too small.

Although not implemented in the current version, this flag edition process should include the exclusion or inclusion of regions defined on a reference database. This would allow for the user to signal specific requirements, excluding a sensitive region from the evaluation or to generate statistics about channels that are not usually present and might not be detected within observation time presented in a single file.

Considering the various traces within a single file all identified regions are consolidated into a single list of identified emissions. Regions that include the same frequency bins are merged by defining four limits, an inner intersection region and an outer union region.

At this point, to ensure that all existing emissions were detected, the inner region of each occupied channel is revisited and re-evaluated. The outer region data of each trace is recorded as representing the channel data at that specific trace.

As described in the proposed HDF5 format, each channel where an emission was detected is stored as an independent object and all empty channels are stored together, composing a background noise object.

For these objects, the datasets stored correspond to: the spectrogram, i.e level for each time and frequency as a bidimensional array; and the level profile, i.e histogram with the number of traces detected for each level and frequency as a bidimensional array.

For the channels where an emission was detected are also stored the time and duration of each event when the emission was observed and when it was not. The duration is computed by assuming that whenever two consecutive traces display the same channel usage status, with or without an emission, such status was maintained between those traces.

This assumption of continuity of an event can only be sustained if there are no events with duration shorter than the time between spectrum sweeps, a condition that must be evaluated over the optimization module.

10.6.2 Emission clustering

This module is coded in the program named “**compute_channel_distance.py**” and performs scaling and cropping operations needed to ensure that distance between two traces, computed with the RMSD, is insensitive to variations created by the measurement and processing conditions.

The algorithm runs through all the listed channels and removes those that are too small or have too few samples. Such cases might be produced by any number of possible spurious or anomalous conditions and should be disregarded from the clustering analysis. Since such cases are exceptions by themselves, they might be grouped as a separate category for later consideration.

For each channel, the arithmetic mean trace is computed as representative of the detected emission. The mean trace for each channel is rescaled to have the peak value equals to 1, this reduces the effects of different attenuation due to propagation or equipment setup effects. It is important to highlight that this is not a process of standardization or min-max rescaling since it is important to maintain unchanged the ratio between the maximum and the minimum values as it describes relevant features of the emission.

After the scaling of each channel, the comparison process is started taking channels in pairs and performing further adjustments.

The first adjustment in the comparison process is to match the scale from both traces such as that they present the same range, e.g. from 1 to 0.25. This is equivalent to increase the noise level or attenuate the stronger emission such as it can be viewed on the same level in relation to the noise as the weaker emission.

After the level adjustment is performed a cross-correlation of the traces, such as to identify the shift needed between them that allows for the greatest similarity, i.e. minimum error.

Once identified the needed shift, it is performed such as to allow the greatest possible intersection between traces. The traces may need to be shifted prior to any comparison since similar emissions might be positioned differently within the channel boundaries, especially when such boundaries are automatically created as in the present case.

Once both traces are adjusted for the lowest possible error, all bins that fall outside the intersection area are discarded. This will happen when the compared channels are represented by a different number of bins or due to the shift made. This ensures that both traces have the same length and the difference between them is the lowest possible.

This procedure of normalization and adjustments is repeated to every pair of traces and the distance is computed resulting in a distance matrix that may be used as the basis for hierarchical agglomerative clustering. Standard Python libraries were used to perform this clustering task.



10.7 WebGUI demonstration with Joomla!

On this annex are presented a few screenshots of the demonstration interface created with Joomla!⁶².

This interface was created as part of the CMS course and serves as a demonstration to the WebGUI and the flexibility of the proposed architecture to integrate different platforms into a system with low implementation cost.

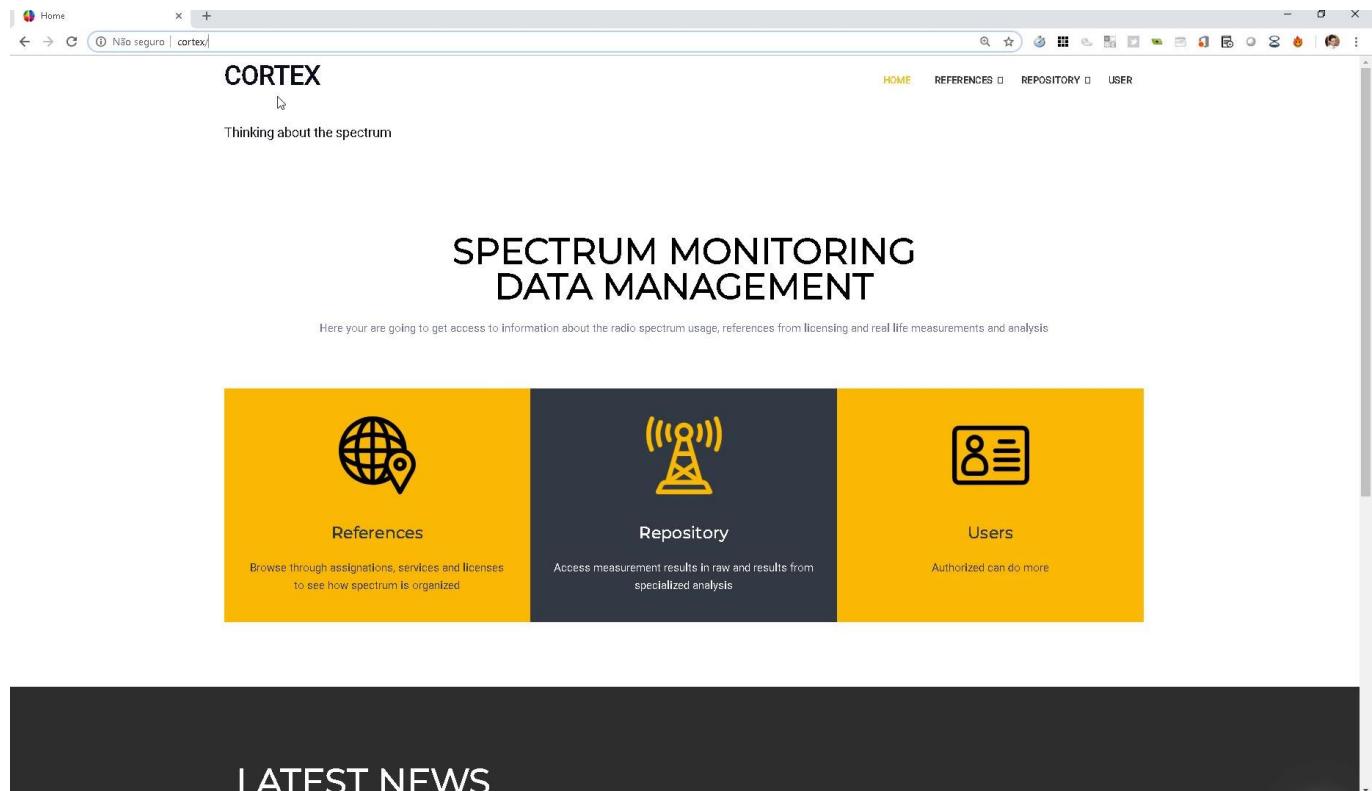


Figure 37. Home page of the system WebGUI provides access to open data repositories and user identification screen.

⁶² <https://www.joomla.org/>

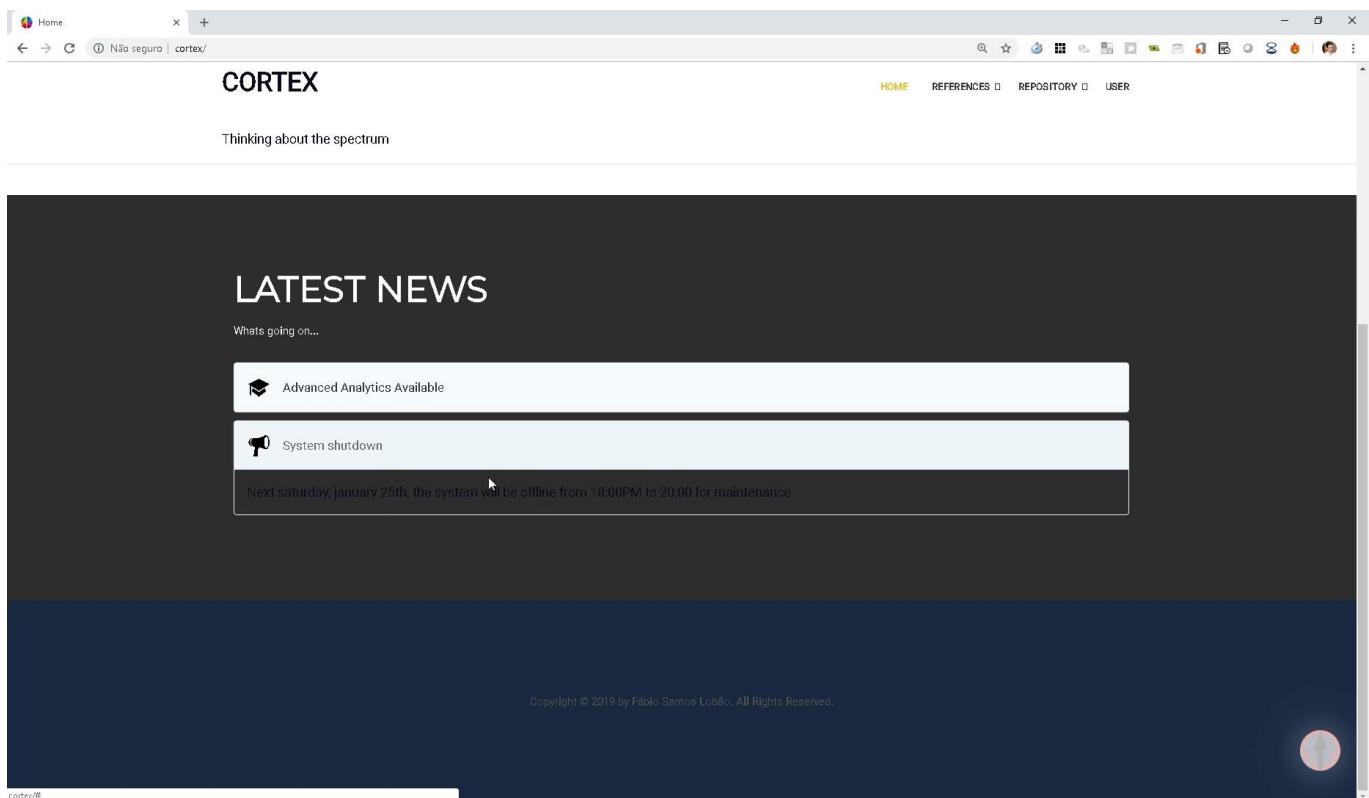


Figure 38. The lower part of the WebGUI homepage also provides an area for information

The screenshot shows the CORTEX References page. At the top, there is a navigation bar with links for HOME, REFERENCES, REPOSITORY, and USER. Below the navigation bar, a banner reads "Thinking about the spectrum". The main content area features a large heading "REFERENCES" and a subtitle "Basic References to Find and Manage Spectrum Data". Below this, there are four circular icons with corresponding labels and descriptions:

- SERVICES**: An image of a stadium with signal beams, with the text "Which are the telecommunication services provided".
- EMITTERS**: An image of a tall antenna tower, with the text "Who is known putting signal in the air".
- CHANNELS**: An image of a hand holding a remote control, with the text "Which are the regulated channels".
- ASSIGNATION**: An image of a spectrum assignment chart, with the text "What is the band assignation".

Figure 39. References page give access to the information on reference tables of the system

Figure 40. Repository page gives access to raw data files, information on detected emissions and noise.

Figure 41. User identification employs standard interfaces.



Intelligent Radio Spectrum Monitoring

The screenshot shows a web browser window for the CORTEX application. The URL is `cortex/index.php/my-account/profile`. The page title is "CORTEX". The top navigation bar includes links for HOME, REFERENCES, REPOSITORY, ADVANCED (which is highlighted in yellow), and USER. A dropdown menu under ADVANCED contains items such as Sites, Stations, RawData Files, and various system tables like Assignation, Channels, Emissions, Emitters, Modules, Noise, Regulations, Services, RT Station Module, RT Channel Emission, RT Emitter Emission, RT RawData Emission, and RT RawData Noise. On the left, there's a sidebar titled "Thinking about the spectrum". The main content area is titled "Profile" and displays basic user information: Name (Super User), Username (lobao), Registered Date (Saturday, 12 January 2019), and Last Visited Date (Friday, 18 January 2019). Below this is a "Basic Settings" section with fields for Editor, Time Zone, Frontend Language, Backend Template S..., Backend Language, and Help Site, all of which show "No information Entered". At the bottom, there are "User Actions" and "Log Options" buttons.

Figure 42. Using a profile with administrative privileges allow the user to edit any of the system tables.

The screenshot shows a web browser window for the CORTEX application. The URL is `cortex/index.php/advanced`. The page title is "Advanced". The top navigation bar includes links for HOME, REFERENCES, REPOSITORY, ADVANCED (highlighted in yellow), and USER. A button labeled "Edit Page" is visible in the top right. The main content area is titled "ADVANCED TABLE EDITING" and contains a sub-instruction: "This allow edition of individual tables on the system database." Below this is a warning: "TAKE EXTREME CARE ON HANDLING THE TABLES". There are three columns of icons representing different database tables: "Assignation" (with a person icon), "Raw data file" (with a document icon), "Stations" (with a location pin icon); "Services" (with a person icon), "Emissions" (with a signal icon), "Modules" (with a stack icon); "Regulations" (with a document icon), "Noise" (with a speaker icon), "Association Stations and Modules" (with a handshake icon); "Channels" (with a speech bubble icon), "Association RawData and Emission" (with a handshake icon), "Association RawData and Noise" (with a handshake icon); "Emitters" (with a signal icon), "Association Emission Emitter" (with a handshake icon), "Association Emission Channel" (with a handshake icon). At the bottom, a dark footer bar contains the text "Copyright © 2019. All rights reserved." and a small logo.

Figure 43 Each table on the database is also accessible through an advanced editing page

The screenshot shows the CORTEX interface with a map of Valencia, Spain, in the background. The map includes various locations like Valencia, Alaquàs, Torrent, and Catarroja. In the foreground, there is a table listing individual data files. The table has columns for Initial Frequency, Final Frequency, Initial Time, End Time, File Attributes, Download, Site Identification, Site Tags, Station Identification, and Station Attributes.

Initial Frequency	Final Frequency	Initial Time	End Time	File Attributes	Download	Site Identification	Site Tags	Station Identification	Station Attributes
88 000 000	108 000 000	2017-04-27	2017-04-28	bin	Download	Valencia 2	Measurement	CRFS RFEYE002087	Fixed TDOA Active
450 000 000	470 000 000	2017-04-30	2017-05-01	bin	Download	Valencia 5	Emitter	CRFS RFEYE002087	Fixed TDOA Active
450 000 000	470 000 000	2017-04-26	2017-04-27	bin	Download	Feria Valencia	Measurement	CRFS RFEYE002088	Fixed TDOA Active
88 000 000	108 000 000	2017-05-01	2017-05-02	bin	Download	Feria Valencia	Measurement	CRFS RFEYE002088	Fixed TDOA Active
450 000 000	470 000 000	2017-04-28	2017-04-29	bin	Download	Valencia 3	Measurement	CRFS RFEYE002089	Fixed TDOA Active

Figure 44. Individual data files may be retrieved using a map and list interface.

The screenshot shows the CORTEX interface with an advanced search dialog open. The dialog allows users to define search criteria using elements, conditions, and values. In the example, a filter is applied where the 'Hz Start' value is less than 200000000. The main table below the dialog lists data files, similar to Figure 44.

Initial Frequency	Final Frequency	Initial Time	End Time	File Attributes	Download	Site Identification	Site Tags	Station Identification	Station Attributes
88 000 000	108 000 000	2017-04-27	2017-04-28	bin	Download	Valencia 2	Measurement	CRFS RFEYE002087	Fixed TDOA Active
88 000 000	108 000 000	2017-05-01	2017-05-02	bin	Download	Feria Valencia	Measurement	CRFS RFEYE002088	Fixed TDOA Active
450 000 000	470 000 000	2017-04-28	2017-04-29	bin	Download	Valencia 3	Measurement	CRFS RFEYE002089	Fixed TDOA Active
450 000 000	470 000 000	2017-04-30	2017-05-01	bin	Download	Valencia 5	Emitter	CRFS RFEYE002087	Fixed TDOA Active

Figure 45. Data may be accessed by any information through advanced queries.

10.8 The relation between the project and the MUGI subjects

The following list present in alphabetical order various courses offered on MUGI and briefly mention their contribution to the present work.

Only subjects with high impact to the present work are mentioned. This list was created only to provide a subjective linkage between the courses and some knowledge asset that was used during this project, it does not relate with the intrinsic relevance, quality or importance of each subject to the information management field.

- ARI: on the understanding of algorithms and architectures for data indexing and retrieval;
- BAN: to support the motivation argument that resulted in the relationship between business analytics and spectrum monitoring services;
- CMS: as a reference for the understanding of the data asset management problem. Also provided the support for the development of the experimental WebGUI in Joomla! and the presented relational database model;
- CPD: on virtualization technologies that were used to create a development environment that is equivalent to the final deployment scenario;
- DGP: on the selection of the methodology used for the project development;
- EXM: on the use of analytical tools to model and analyse large datasets. Also provided room for initial experimentation with the used sample dataset;
- FDI: provided insight into the importance of open data initiatives;
- IAP: as reference for the microservice architecture and web service connection to reference data providers;
- MLD: on the legal and ethical implications associated with information systems, especially concerning data protection regulations;
- SEN: as reference for cloud services and scaling alternatives that allow the efficient processing of large volumes of data. Also, the reference to automation framework alternatives and the possible deployment of the spectrum monitoring network data management system as a cloud service;
- SIA: on security issues concerning systems distributed over the internet, such as observed in the case of the used spectrum monitoring sensor network;
- TII: on criteria and techniques used for scientific research, especially those concerning qualitative analysis and an initial survey of user requirements;
- WSO: to the underlying understanding of ontologies and their importance for data management.



11 Glossary

Allocation	“(of a frequency band): Entry in the Table of Frequency Allocations of a given frequency band for the purpose of its use by one or more terrestrial or space radiocommunication services or the radio astronomy service under specified conditions. This term shall also be applied to the frequency band concerned.” [86].
Allotment	“(of a radio frequency or radio frequency channel): Entry of a designated frequency channel in an agreed plan, adopted by a competent conference, for use by one or more administrations for a terrestrial or space radiocommunication service in one or more identified countries or geographical areas and under specified conditions.” [86].
Anatel	Brazilian National Telecommunications Agency (<i>Agência Nacional de Telecomunicações</i>), http://www.anatel.gov.br/institucional/ .
Assignment	“(of a radio frequency or radio frequency channel): Authorization given by an administration for a radio station to use a radio frequency or radio frequency channel under specified conditions.” [86].
CRFS	UK based equipment manufacturer and solution provider of systems for spectrum monitoring, management and geolocation. https://www.crfs.com
CRUD	Create, Read, Update and Delete. Acronym to the four basic functions associated with persistent data storage.
DTT	Digital Terrestrial Television. A broadcast communication service that provides television content using terrestrial transmitters.
DVB-T	Digital Video Broadcasting, an international standard used to provide digital television services. The “T” refers to the specific standard used for <u>DTT</u> . This is the standard used by most of the countries, including all of Europe, most of Africa, Asia and Oceania.
ePING	Brazilian electronic government interoperability standard. http://eping.governoeletronico.gov.br/
Intermodulation	Emissions generated by the non-linear combination of two or more other emissions, at frequencies that are harmonic to the sum and difference of the originating emissions.



ISDB-T	Integrated Services Digital Broadcasting, an international standard used to provide digital television services. The “T” refers to the specific standard used for DTT . This is the standard in use in Brazil and most of South America. Originally created in Japan is also adopted in a few countries in Asia and Africa.
LDAP	Lightweight Directory Access Protocol, a standard protocol for the access and maintenance of directory information service. It most commonly is used to store usernames and passwords.
MIB	Management Information Base, is an object composed of one or more attributes, presented in a hierarchical format and that is used to organize the information accessed using a protocol such as SNMP.
Occupancy ratio	The ratio between the time a certain frequency is used (occupied) in relation to a reference observation period. e.g. while monitoring a channel for 1 hour, it is observed that it is in use for a total of 30 minutes. Its occupancy ratio is 50% of that 1 hour. Computation and statistical details are presented on ITU references [87] and [14].
OFDM	Orthogonal Frequency-Division Multiplexing, a method used for wideband digital communication that employs multiple carrier frequencies orthogonally modulated and ordered in a very compact arrangement.
Online Algorithms	A class of algorithms that are executed synchronously with the input data stream, producing a corresponding output data stream. e.g. compute the arithmetic mean temperature every minute by adding the measured value for that minute to a sum variable, increment a counter for the number of values added to the sum and output the ratio between the sum and the counter.
SBC	Single Board Computer. A computer system integrated into a single board that does not require modules or daughter boards to operate. Usually has a small form factor and is suitable for use on automation and small appliances.
SFTP	SSH File Transfer Protocol or Secure File Transfer Protocol is a standard communication protocol from IETF that allows for secure file management, including access and transfer through an unsecured communication channel by using added security features inherited from the SSH protocol.
SNMP	Simple Network Management Protocol is a communication standard defined by the IETF to provide tools, manage and monitor devices operating within an IP network. See RFC3411 and RFC3418 .

SSH	Secure Shell, is a communication protocol from IETF that allows for the secure use of services from network devices over an unsecured network.
SSO	Single Sign-On, an access control method where the user is required to perform a single authentication procedure to gain access to several applications and services.
VPN	Virtual Private Network, a private communication network created between devices across another network, allowing the participating devices to provide and consume services unrestricted by the underlying network characteristics.
Web GUI	A graphical user interface available through an internet browser, such as Chrome, Firefox, Internet Explorer or Safari.