METROLOGY IN THE CLOUD THROUGH MICROSERVICES WITH VARIABILITY MANAGEMENT

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ANIL ÇETİNKAYA

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN
COMPUTER ENGINEERING

SEPTEMBER 2024

Approval of the thesis:

METROLOGY IN THE CLOUD THROUGH MICROSERVICES WITH VARIABILITY MANAGEMENT

submitted by **ANIL ÇETİNKAYA** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Naci Emre Altun
Dean, Graduate School of **Natural and Applied Sciences** ______________

Prof. Dr. Halit Oğuztüzün
Head of Department, **Computer Engineering** ______________

Prof. Dr. Halit Oğuztüzün
Supervisor, **Computer Engineering, METU** ______________

Dr. M. Çağrı Kaya
Co-supervisor, **Computer Science and Engineering, Chalmers University of Technology** ______________

Examining Committee Members:

Prof. Dr. Hürevren Kılıç
Computer Engineering, Atılım University ______________

Prof. Dr. Halit Oğuztüzün
Computer Engineering, METU ______________

Assoc. Prof. Dr. Hande Alemdar
Computer Engineering, METU ______________

Assoc. Prof. Dr. Pelin Angın Ülküer
Computer Engineering, METU ______________

Assoc. Prof. Dr. Gül Tokdemir
Computer Engineering, Çankaya University ______________

Date:

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Anıl Çetinkaya

Signature :

ABSTRACT

METROLOGY IN THE CLOUD THROUGH MICROSERVICES WITH VARIABILITY MANAGEMENT

Çetinkaya, Anıl
Ph.D., Department of Computer Engineering
Supervisor: Prof. Dr. Halit Oğuztüzün
Co-Supervisor: Dr. M. Çağrı Kaya

September 2024, 86 pages

This thesis introduces a comprehensive microservice-based architecture tailored for the complexities of the Industrial Internet of Things (IIoT), with a focus on the metrology, in particular, calibration industry. The proposed architecture enhances the Internet of Measurement Things (IoMT) framework to effectively manage the digital transformation challenges encountered in metrology. A pivotal characteristic of the proposed architecture is the incorporation of methodological variability management mechanism, implemented through Textual Variability Model (TVM) integrated with a customized Continuous Integration/Continuous Deployment (CI/CD) pipeline. This approach ensures real-time adaptability, enabling seamless updates and continuous operation tailored to specific system requirements. The architecture's effectiveness is demonstrated through a detailed case study focusing on RF power measurement, showcasing its ability to manage the entire calibration process, from device communication to the generation of Digital Calibration Certificates (DCCs). Advanced uncertainty quantification techniques, including the Law of Propagation (LoP) and Monte Carlo Simulation (MCS), are embedded within the microservices, ensuring precision and reliability in calibration tasks. This thesis contributes to the broader field by illustrating how microservice-based architectures, enhanced with variability management and CI/CD mechanisms, can streamline processes, improve scalability, and support the evolving needs of complex, heterogeneous systems across various industries.

Keywords: Software Architecture, Microservices, Variability, Industrial Internet of Things, Digital Transformation

ÖZ

DEĞİŞKENLİK YÖNETİMİ ALTINDAKİ MİKROSERVİSLER YOLUYLA BULUTTA METROLOJİ

Çetinkaya, Anıl
Doktora, Bilgisayar Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. Halit Oğuztüzün
Ortak Tez Yöneticisi: Dr. M. Çağrı Kaya

Eylül 2024 , 86 sayfa

Bu tez, özellikle kalibrasyon endüstrisi olmak üzere metrolojiye odaklanarak Endüstriyel Nesnelerin İnterneti'nin (IIoT) karmaşıklıklarla başa çıkabilen kapsamlı bir mikroservis tabanlı mimariyi tanıtmaktadır. Önerilen mimari, metrolojide karşılaşılan dijital dönüşüm zorluklarını etkin bir şekilde yönetmek amacıyla Ölçüm Cihazlarının İnterneti (IoMT) çerçevesini geliştirmektedir. Bu mimarinin önemli bir özelliği, özelleştirilmiş bir Sürekli Entegrasyon/Sürekli Dağıtım (CI/CD) hattı ile entegre edilmiş Metinsel Değişkenlik Modeli (TVM) aracılığıyla uygulanan metodolojik değişkenlik yönetimi mekanizmasının dahil edilmesidir. Bu yaklaşım, gerçek zamanlı uyarlanabilirliği sağlayarak belirli sistem gereksinimlerine göre uyarlanmış kesintisiz güncellemeler ve sürekli çalışma olanağı sağlar. Mimarinin etkinliği, cihaz iletişiminden Dijital Kalibrasyon Sertifikalarının (DCC) oluşturulmasına kadar tüm kalibrasyon sürecini yönetme yeteneğini sergileyen RF güç ölçümüne odaklanan detaylı bir vaka çalışması ile gösterilmektedir. Kalibrasyon süreçlerinde hassasiyet ve güvenilirliği sağlamak amacıyla, mikroservislerin içine Yayılma Kanunu (LoP) ve Monte Carlo

Simülasyonu (MCS) gibi ileri düzey belirsizlik hesaplama teknikleri entegre edilmiştir. Bu tez, değişkenlik yönetimi ve CI/CD mekanizmaları ile geliştirilmiş mikroservis tabanlı mimarilerin süreçleri nasıl kolaylaştırabileceğini, ölçeklenebilirliği nasıl artırabileceğini ve çeşitli endüstrilerdeki karmaşık, heterojen sistemlerin gelişmekte olan gereksinimlerini nasıl destekleyebileceğini göstererek daha geniş bir alana katkıda bulunmaktadır.

Anahtar Kelimeler: Yazılım Mimarisi, Mikroservisler, Değişkenlik, Endüstriyel Nesnelerin İnterneti, Dijital Dönüşüm

To My Family

ACKNOWLEDGMENTS

I am deeply grateful to my Thesis Monitoring Committee, especially to my advisor, Prof. Halit Oğuztüzün, for his unwavering support, guidance, and encouragement throughout my study. I also extend my sincere thanks to Prof. Ali Doğru for his valuable contributions and mentorship. A special mention goes to my co-advisor, Dr. M. Çağrı Kaya, whose contributions have significantly shaped the direction and quality of my study. His consistent support and insightful guidance have been indispensable throughout this journey. These individuals have not only been remarkable in their dedication but have also set a standard of excellence that I aspire to follow. Their passion for research and commitment to mentorship have made them true role models in both my academic and personal development.

I would like to extend my gratitude to Associate Professors Pelin Angın and Gül Tokdemir for their helpful feedback and constructive comments on my thesis work. I also want to express my sincere appreciation to Dr. Erkan Danacı and Dr. Aliye Kartal Doğan for their guidance and support. Their expertise in the field of metrology greatly enriched our understanding throughout the research and the projects we contributed to. Additionally, I would like to thank Handan Sakarya, Murat Arslan, and Gülsün Tünay for their assistance during these projects. I also would like to thank Hasan Acar, Tuğberk İşyapar, Alper Karamanlıoğlu and Atakan Garipler for their friendship and support over the years, both personally and professionally.

Finally, I wish to express my profound gratitude to my family for their enduring patience, encouragement, and unwavering support throughout this journey.

The work described in this thesis is partially supported by The Scientific and Technological Research Council of Turkey (TÜBİTAK) under Grant No. 5200040 and in the framework of the 22RPT04 RFMicrowave2 project which has been supported by the European Partnership on Metrology, co-financed by the European Union's Horizon Europe Research and Innovation Programme.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

ABBREVIATIONS	
API	Application Programming Interface
ARP	AutoRFPower
BIPM	International Bureau of Weights and Measures
BPMN	Business Process Model and Notation
CI/CD	Continuous Integration/Continuous Deployment
CLs	Calibration laboratories
COSE	Component-Oriented Software Engineering
CSAPI	Cloud Services API
DCC	Digital Calibration Certificate
DCCs	Digital Calibration Certificates
DUT	Device Under Test
GKE	Google Kubernetes Engine
GPIB	General Purpose Interface Bus
GUM	Guide to the Expression of Uncertainty in Measurement
IaC	Infrastructure as Code
IEEE	Institute of Electrical and Electronics Engineers
IIoT	Industrial Internet of Things
IoMT	Internet of Measurement Things
IoT	Internet of Things
LoP	Law of Propagation of Uncertainty
MAC	Media Access Control
MCS	Monte Carlo Simulation
MII	Metrology Information Infrastructure

MS	Microsoft
MVC	Model-View-Controller
NI	National Instruments
NUM	NIST Uncertainty Machine
OCB	Oracle Crystal Ball
PaaS	Platform as a Service
PDF	Probability Density Functions
PoC	Proof of Concept
PTB	Physikalisch-Technische Bundesanstalt
QR	Quick Response
RF	Radio Frequency
ScoA	Scope of Accreditation
SPLE	Software Product Line Engineering
SoA	Service-oriented Architecture
SQL	Structured Query Language
SWR	Standing Wave Ratio
TÜBİTAK	Scientific and Technological Research Council of Türkiye
TVM	Textual Variability Model
UI	User Interface
UML	Unified Modeling Language
UUID	Universally Unique Identifier
VS	Visual Studio
XML	Extensible Markup Language

CHAPTER 1

INTRODUCTION

The microservice architecture represents a paradigm shift in software system design and construction methodology. Unlike traditional monolithic architectures, where all application components are tightly interwoven, microservices divide applications into smaller, self-contained services. These services operate autonomously and communicate across a network. Based on the principle of bounded context, each microservice is engineered to execute a specific functionality, allowing for independent development, deployment, and scalability. The adoption of this architectural approach presents several advantages, especially in the context of digital transformation, as it involves seamlessly integrating digital technology to revolutionize business operations.

One of the primary advantages of microservice architecture is its ability to enhance scalability and flexibility. By decoupling various functions into individual services, microservices allow organizations to scale only the parts of the application that require additional resources, without affecting the entire system. This modularity also facilitates easier maintenance and updates, as changes can be made to one service without requiring a complete overhaul of the application. Newman [5] highlights these benefits, emphasizing that this architecture allows for rapid development cycles and reduces the risk of system-wide failures when updates are deployed.

Microservices play a vital role in enabling organizations to be more agile and responsive to evolving market demands. As digital technologies progress, businesses must continuously adapt their software systems to encompass new functionalities, integrate with emerging platforms, and meet customer expectations. Bucchiarone et al. [6] discuss how microservices support this adaptability, particularly in industries like

Industry 4.0, where the ability to quickly integrate new technologies and processes is essential for maintaining competitiveness.

Furthermore, microservices contribute to the success of digital transformation by supporting the transition from legacy monolithic systems to more modern, cloud-native architectures. This migration is often a critical step in digital transformation, as it allows businesses to take full advantage of cloud computing's scalability, flexibility, and cost-effectiveness. Microservices enable this transition by allowing legacy components to be incrementally replaced with microservices, reducing the complexity and risk associated with large-scale system overhauls [7].

Variability management is a critical aspect of Software Product Line Engineering (SPLE) that focuses on systematically handling the differences and commonalities within software systems. By effectively managing variability, organizations can create a range of related products from a shared set of core assets, significantly enhancing software reuse, reducing development costs, and accelerating time-to-market. The importance of variability management in achieving the scalability and flexibility required for large-scale software systems is discussed in [8].

Incorporating variability management into microservices architecture has the potential to customize and adapt microservices to specific needs without disrupting the overall system. This approach can enhance the modularity and scalability of microservices, making them more robust and adaptable to changes in business requirements. However, integrating variability management within microservices architecture presents unique challenges and opportunities, as explored in [9]. This study identifies six key challenges in integrating variability management with microservices, with a focus on feature identification, variability modeling, and the customization of microservices. These challenges highlight the complexities of managing variability in dynamic, decentralized systems like microservices, where maintaining system coherence while enabling flexibility is crucial. To tackle these challenges, the authors provide an open-access repository of six microservice-based webshops as a case study.

The incorporation of variability management into microservices architecture, though it enhances modularity and adaptability, presents specific challenges. These challenges are particularly pertinent as industries such as metrology and calibration undergo digital transformation, compelled by the necessity to enhance efficiency, data accuracy, and competitiveness in a rapidly evolving global marketplace. With the metrology sector embracing automated, network-based solutions, the integration of variability management into microservices becomes essential to address the distinct and dynamic needs of this industry.

This transformation is imperative as the continuous technological advancements are reshaping every sector, pressing them towards reducing costs, speeding up business processes, and utilizing human resources more efficiently. The metrology and calibration industry finds itself at the forefront of this change, striving to keep pace with these demands [10, 11]. This shift is highlighted by the development and implementation of data standards coupled with the adoption of automated, network-based solutions. These steps are crucial for enhancing efficiency and data accuracy, thus allowing stakeholders to remain competitive in a swiftly evolving global marketplace [12]. In this context, the integration of variability management within microservices architecture becomes increasingly crucial. As the metrology and calibration sectors adopt these technological advancements, the challenges of managing variability within dynamic and decentralized systems become particularly relevant.

Any measurement equipment has a margin of error in its measurements and may deviate over time due to various reasons, including misuse and environmental factors such as temperature and humidity. To keep this drift in the results of the equipment within acceptable limits, devices need to be calibrated regularly. For some equipment, calibration is even a legal responsibility. The device under test is checked against certain test points, and measurement results are recorded during the calibration process. A "standard device", a trusted device usually calibrated by an upper-level body on the calibration pyramid [13], is also part of the calibration process [14] to be compared with the device under test. At the end of the process, the device under test becomes "calibrated", and a calibration certificate is generated containing device information (such as the manufacturer and model), the calibration lab's details, and the uncertainty values calculated by the laboratory.

The calibration process roughly involves the phases of data collection, uncertainty calculation, and calibration certificate generation. The overall process still requires manual handling of data. Although there are tools for providing automation, they mostly focus on certain parts of the process (such as device communication, and uncertainty calculation [15]), lacking an end-to-end perspective. Besides, traditional calibration certificates are paper-based or digitally signed PDF documents, which is considered the cheapest and safest way (e.g., protected from physical errors or numerical alterations). Recently, machine-readable digital calibration certificate (DCC) standards [16] have been proposed that can ensure stakeholder satisfaction and support the ongoing digital transformation in the field. Hence, there is a need for an application to handle the calibration process holistically: starting from data collection from the calibration equipment, performing uncertainty calculations, and providing the calibration certificate. Our application addresses these expectations while adopting state-of-the-art cloud-based application development to better contribute to the digital transformation of the metrology and calibration industry.

The adoption of Internet of Things (IoT) technologies in various industries has presented new prospects in big data analytics, machine learning, and cloud computing. This concept, known as the Industrial Internet of Things (IIoT) or Industry 4.0, has delivered notable advantages such as increased productivity, shortened development cycles, swift product customization, and improved resource efficiency [17]. Recognizing these benefits, numerous IIoT solutions have been put forth in several sectors, including agriculture, manufacturing, and telecommunications. Given these advantages, IIoT technologies offer promising solutions for tackling ongoing research challenges in metrology and the calibration industry.

The notion of the "Internet of Measurement Things (IoMT)" was introduced in [18] as a layered IIoT architecture designed to segregate physical equipment, cloud-based services, and applications. This architecture builds upon the Metrology Information Infrastructure (MII) [19] initiative and draws from the experiences of the Metrology.NET platform [20]. Both MII and Metrology.NET aim to establish community-driven standards and enhance automation within the metrology field. IoMT aims to advance these efforts into an IIoT-based framework.

The IoMT architecture consists of three layers: the *physical* layer, which includes calibration equipment typically found in calibration laboratories (CLs); the *MII Cloud*

Services layer, which hosts services for the calibration industry; and the *application* layer, which comprises various software used in metrology and calibration, such as calibration automation systems, asset tracking systems, and scope of accreditation (ScoA) editors.

This study demonstrates how microservice-based solutions improve the digitalization of the metrology and calibration industry through cloud-based, scalable, and maintainable applications, ensuring data accuracy, integrity, and standardization. We developed a cloud-based application capable of handling different stakeholders' needs and data, performing uncertainty calculations, and producing calibration certificates on the cloud. Two specialized microservices were developed to perform uncertainty calculations, each based on a different technique and implemented using different technologies and programming environments. Additionally, we have integrated microservices for the generation and management of DCCs, designed with the flexibility to accommodate future extensions. We leveraged Kubernetes [21] for orchestration.

Google Kubernetes Engine (GKE) [22] utilization enhanced with variability handling brings several benefits, including automated rollouts and rollbacks, simplified management of containerized services, and automatic scaling and load balancing capabilities. In cloud development, a rollout refers to the process of deploying a new version of an application or service. This usually involves gradually replacing the old version with the new one to ensure a smooth transition and minimal disruption. Conversely, a rollback refers to the process of reverting to a previous version of an application or service, typically to undo changes from a recent deployment that may have introduced issues or errors, ensuring the stability and continuity of the service. Therefore, GKE enables efficient management of application updates and the maintenance of system stability, which is crucial for meeting the dynamic requirements of the calibration industry.

The previous version of the AutoRFPower [1] application was a desktop application capable of communicating with the calibration equipment, collecting measurement data from them, and calculating uncertainties based on two techniques, namely, Law of Propagation of Uncertainty (LoP) and Monte Carlo Simulation (MCS) based on the Guide to the Expression of Uncertainty in Measurement (GUM) [23] and [24], respectively. Danaci et al. [2] presented a comprehensive approach for uncertainty evaluation in RF power measurements using the AutoRFPower software. The software incorporates LoP and MCS methods. The study validated the software's capability to accurately calculate measurement uncertainties and highlighted the advantages of using MCS for handling complex, non-linear relationships in uncertainty propagation.

In another previous work, a layered architecture is proposed to improve standardization and availability for the metrology and calibration industry applications in the context of the Industrial Internet of Things (IIoT), namely the "Internet of Measurement Things" (IoMT) architecture [25, 26]. Adhering to this layered architecture, the idea and initial steps of migrating the uncertainty modules of the AutoRFPower application to a cloud environment, with the name of "uncertainty-calculation-as-a-service" were explained in another previous work [27]. In the present paper, we enhance the IoMT architecture with microservices. The AutoRFPower application is migrated to the cloud environment based on the extended architecture in the scope of this work.

The previous work on this research direction and the contribution of this thesis can be summarized as follows:

- The IoMT architecture was presented as a specialization of IIoT architecture [18, 26].
- The AutoRFPower (ARP) software was developed as a desktop application to calculate uncertainties of RF power measurement devices automatically [1].
- Adhering to the IoMT architecture, the idea and initial steps of migrating the uncertainty modules of the ARP application to a cloud environment were explained [27].
- The validation of the utilized statistical techniques to calculate uncertainties using the ARP software was presented in [2]
- The previously presented IoMT architecture is adapted to the microservice architecture in [28]. Also, the modules of the ARP application are re-implemented as microservices, including the validated uncertainty calculation services, keep-

ing the business logic the same for the migrated ones. Furthermore, new functionalities are added to the up-to-date version of the application as newly implemented microservices, including variability handling with a customized Continuous Integration/Continuous Deployment (CI/CD) pipeline, DCC generation, and authentication. This study mostly focused on the software architecture aspect of the proposed microservice-bassed architecture.

- An enhanced version of the work presented in [28] is demonstrated in a detailed manner in a study [3] titled: "Uncertainty Calculation as a Service: Integrating Cloud-Based Microservices for Enhanced Calibration and DCC Generation". This study presented a broader background and literature review, implementation details of the uncertainty calculations, the workflows of the calibration and the authentication processes, a case study demonstrating the applicability of the approach, and the validation of the used uncertainty calculation techniques. The mentioned article focuses on the details of the calibration processes starting from gathering data from devices such as power sensors.

1.1 Motivation and Problem Definition

The continual progression of technology has significantly influenced various industries, especially in the realms of metrology and calibration. These sectors are pivotal in maintaining precise and reliable measurements critical to the functionality and integrity of various domains including security, healthcare, industrial automation, and communications. As digitalization becomes embedded within industry operations to enhance efficiency, minimize expenses, and expedite business processes, the disciplines of metrology and calibration are compelled to evolve in concert with these advancements. Nonetheless, the inherent complexity of calibration processes, compounded by the increasing sophistication of measurement instruments, poses formidable challenges.

Traditional calibration processes are often manual and fragmented, relying on paper-based certificates or digitally signed documents. While these methods are secure, they do not adequately meet the growing need for automation, data accuracy, and standardization required in an industry undergoing rapid digital transformation. Moreover, the rise of the Industrial Internet of Things (IIoT) has introduced new opportunities for leveraging big data analytics, machine learning (ML), and cloud computing to enhance productivity, customization, and resource efficiency in calibration processes. The concept of the Internet of Measurement Things (IoMT) has emerged as a framework for integrating physical calibration equipment with cloud-based services, yet many existing solutions lack a comprehensive, end-to-end approach to managing calibration data from collection to the generation of Digital Calibration Certificates (DCCs).

The motivation behind this thesis partly originates from real-world challenges encountered during a Scientific and Technological Research Council of Türkiye (Tübitak) 1505 project titled "A New Method and Software Development for Automatic RF Power Measurement and RF Power Meter Calibration." Tasked with the development of a software solution tailored for the automation of RF power measurements, alongside concurrent uncertainty analyses, we delved into the complexities of device communication, interfacing with a plethora of sophisticated instruments from leading manufacturers. However, unforeseen challenges soon emerged, extending beyond the technical difficulties of remotely controlling these devices. We faced significant compatibility issues between different brands, drivers, and systems, often due to manufacturers intentionally limiting compatibility to control market share, as well as the intricacies of operating in highly sensitive measurement environments.

For instance, certain device drivers utilized the same Dynamic-link Library (DLL) on the operating system, meaning that installing one driver could inadvertently disrupt the functioning of another. This necessitated specific installation sequences to ensure that multiple drivers, such as those for NI Max and Keysight IO Expert, could coexist without conflicts. Additionally, some hardware setups required computers with highly specific configurations tailored to particular devices from individual manufacturers. This complexity extended to the cables and connectors, which were sometimes recognized only by particular software or drivers. Consequently, laboratories were often forced to operate multiple computers, each dedicated to distinct sets of devices, further complicating workflow management and maintenance.

This fragmented setup also affected the reliability of the calibration process, as tests were often completed with different hardware setups, contributing to variability in the results. Additionally, operators with varying levels of expertise and eligibility to work with specific devices introduced further variability into the calibration processes. To address these challenges, we implemented systematic variability handling mechanisms, ensuring smoother device integration and operational reliability across diverse hardware configurations.

Furthermore, our engagement with the European Partnership on Metrology project, titled "Development of RF and Microwave Metrology Capability II (RFMicrowave2)," underscored the challenges of managing the complexities of metrology on an international scale. This project aimed to enhance the research and measurement capacities of emerging EURAMET countries by transferring theoretical and practical know-how between partners. However, dealing with National Metrology Institutes (NMIs) from multiple countries introduced additional complexity, particularly in tracking software configurations tailored to specific requirements and devices. This experience underscored the necessity of implementing CI/CD pipelines to manage the diverse and evolving needs of different stakeholders within the calibration sphere.

To navigate these multifaceted challenges, we pivoted towards adopting a microservices architecture. This architectural framework allowed us to dismantle the previously monolithic system into a series of modular components. Each component could be independently developed, deployed, and scaled.The adoption of microservices was pivotal, enabling us to encapsulate specific functionalities - such as uncertainty calculations and certification processes - into distinct, discrete services. This approach markedly enhanced the system's scalability, maintainability, and adaptability, making it robust against the shifting demands of the industry.

The problem addressed by this thesis is the gap in existing calibration systems that do not fully capitalize on the potential of a cloud-based microservices architecture for managing the complexities of calibration processes. Specifically, there is a need for a system that automates the entire calibration process while integrating advanced uncertainty calculation methods, managing data variability, and generating DCCs in a secure, standardized, and scalable manner. This thesis proposes an enhanced microservice-based cloud architecture designed to overcome these challenges by providing a comprehensive solution that integrates various measurement types and communication technologies, ensuring efficient and accurate calibration processes while aligning with the digital transformation trends in the industry.

1.2 Contributions and Novelties

The thesis articulates an approach to digitalization of metrology through the advent of an End-to-End Development Solution. This holistic solution encompasses the entire calibration process, commencing with direct device communication at the foundational physical layer and culminating in the standard-compliant generation of Digital Calibration Certificates (DCCs). This methodology streamlines the calibration process by establishing direct interfacing with measurement devices for data acquisition, subsequently channelling the acquired measurement data into cloud storage. It advances the process by employing sophisticated algorithms deployed in the cloud infrastructure to perform rigorous uncertainty calculations, a pivotal step for ensuring the precision and reliability of calibration data. The resultant data are then assimilated into DCCs, ensuring an integrated process that maintains data integrity. This approach notably mitigates risks associated with data tampering and operator errors, thereby significantly enhancing the reliability and accuracy of the calibration process.

Furthermore, the thesis introduces an Enhanced Microservice-Based Architecture, representing a significant evolution of the existing Internet of Measurement Things (IoMT) architectural framework, tailored for the calibration industry's intricate requirements. This proposed architectural framework leverages the principles of microservices to foster a modular, scalable, and seamlessly maintainable software ecosystem. Such an architecture not only facilitates the accommodation of the diverse and evolving needs of various stakeholders within the calibration sphere but also provides a robust framework for conducting complex uncertainty calculations and generating DCCs in a cloud computing environment. The proposed architecture ensures that applications developed within this framework are inherently more adaptable to changing industry demands due to its microservice-based design, which promotes modularity and scalability.

The work under discussion integrates two premier uncertainty quantification techniques, the Law of Propagation (LoP) and Monte Carlo Simulation (MCS), into a cloud-based infrastructure through designed microservices. These methodologies are implemented into the system with an emphasis on adaptability, allowing interoperability across diverse programming environments. This integration paradigm caters to the intricate needs of a wide array of calibration processes.

The architecture incorporates a Textual Variability Model (TVM) that dynamically manages the heterogeneity of calibration equipment and processes. This allows for real-time adjustments and configuration of the system based on the specific requirements of the measurement equipment and the environmental conditions, enhancing the system's adaptability and responsiveness. To further enhance this ability, a custom CI/CD pipeline, enriched with automated triggers, is implemented to enhance the system's adaptability and responsiveness. This integration of customized CI/CD pipelines and dynamic variability management helps to improve the current state of the calibration services while paving the way for digitalization and standardization in Metrology and the calibration industry as a whole.

Furthermore, the thesis introduces a microservice for the generation of DCCs in alignment with emerging industry standards. his service includes robust authentication mechanisms, including the deployment of UUID-based QR codes, thereby ensuring the integrity and authenticity of the generated certificates.

The practical application of this proposed system is demonstrated through a case study focusing on RF power measurement. This empirical evaluation compares the statistical methodologies employed against those utilized by the widely recognized simulation tool, Oracle Crystal Ball, showcasing the proposed architecture's capability in managing complex calibration tasks. This case study validates the architecture's ability to handle complex calibration processes and illustrates the benefits of the proposed system in terms of accuracy, scalability, and ease of integration with existing systems.

1.3 The Outline of the Thesis

The rest of this thesis is constructed as follows:

- In Chapter 2, a comprehensive background and literature review are provided, covering relevant concepts in software architecture, microservices, and variability management, including insights into service composition, heterogeneity, and interoperability challenges.
- Chapter 3, presents the ARP system's architecture, functionalities, and how it addresses metrological traceability by minimizing user errors. It discusses device integration, user interface designs, and its application in calibration and production environments.
- Chapter 4, introduces the "Proposed Microservice Architecture with Variability Management." It presents the integration of the IoMT framework with microservices and variability management. The chapter also discusses the utilization of CI/CD pipelines for automated deployment, and details the system workflow from device setup to DCC generation. Furthermore, the statistical techniques used for uncertainty calculations, including the LoP and MCS, are examined in the context of ensuring the accuracy and reliability of the system.
- Chapter 5 presents a system evaluation, beginning with a case study demonstrating the integration of measurement devices and uncertainty calculations in a cloud environment. The chapter validates the accuracy and reliability of the uncertainty calculation methods, specifically the LoP and MCS, by comparing the results with previous studies conducted in a local setting. It includes a critical evaluation of both methods, highlighting their advantages and limitations, particularly in complex RF power measurement scenarios. The chapter also discusses the findings, system constraints, and future directions, suggesting enhancements in digital metrology and calibration using microservices and cloud computing.
- Chapter 6, concludes the thesis with remarks on the contributions and limitations of the proposed work and outlines directions for future research and development.

CHAPTER 2

BACKGROUND AND RELATED WORK

2.1 Background

This section includes the introduction of notions that form the foundation for the suggested approach.

2.1.1 Calibration

Metrology is defined as "the science of measurement and its applications" [29]. Calibration is an essential aspect of metrology, playing a critical role in ensuring the accuracy and quality of measurements across different domains. In time, measurement equipment can develop errors due to external factors such as misuse, temperature variations, and humidity. These discrepancies necessitate regular calibration to uphold dependable performance.

Each measurement device inherently possesses a margin of error, which is known as measurement uncertainty. The purpose of calibration is to determine this uncertainty or verify that it is within acceptable limits. This process involves comparing the measurement device against a standard to identify any deviations. While calibration identifies and quantifies these variations, the subsequent step of adjustment corrects any discrepancies. Unlike calibration, which involves testing the device, adjustment focuses on fine-tuning the device to restore its accuracy.

2.1.2 Uncertainty

Measurement uncertainty is a crucial aspect of metrology and calibration as it represents the doubt associated with the result of a measurement. It reflects the range within which the true value is expected to lie. Understanding and managing measurement uncertainty is essential for ensuring the reliability and comparability of measurement results. Factors such as the instrument's precision, calibration method, environmental conditions, and the operator's proficiency contribute to measurement uncertainty [23]. Accurately calculating and reporting uncertainty is vital as it helps stakeholders make informed decisions based on measurement data, thereby reinforcing trust in the measurement process.

Quantifying uncertainty in the calibration process involves evaluating all potential sources of error and their combined effect on the measurement result. Standards such as GUM offer comprehensive frameworks for uncertainty calculation [23], [24]. Adhering to these standards allows calibration services to ensure that their uncertainty estimates are consistent and transparent, facilitating the comparison of measurements across different laboratories and applications.

While LoP is commonly used for calculating uncertainty, the MCS method offers a versatile and robust approach for uncertainty calculation in calibration processes. Unlike the strictly defined analytical approach of LoP, MCS uses computational algorithms to model and propagate uncertainties through a large number of simulated trials. This stochastic method excels in handling complex and non-linear relationships that may be difficult to address analytically, providing a detailed statistical representation of measurement uncertainties by generating and analyzing numerous random samples [30].

2.1.3 Microservices

In the realm of software architecture, a prominent approach for organizing software applications into a suite of services is service-oriented architecture (SoA) and microservices are a modular way of realizing SoA. SoA prioritizes the creation of loosely coupled, interoperable services that communicate over a network using standard protocols. This approach allows for modular design, enabling the reuse and integration of services into larger systems with ease. SoA examples include web services, which facilitate diverse applications to exchange data, and enterprise services, which streamline business operations across different platforms [31].

Microservices represent an evolution of the SoA paradigm, focusing on constructing software applications from a collection of small, self-contained, and independently manageable services. The aim is to divide applications into smaller, independently deployable services, each assigned to a specific business task. These services interact through lightweight, standardized protocols, ensuring seamless communication despite technological diversity. This modular approach allows for the utilization of various programming languages, databases, and technology stacks, as long as they adhere to common interfaces and protocols. This architectural strategy enhances scalability, agility, and maintainability, empowering teams to deploy updates and new features more frequently and reliably [32].

Additionally, microservices are typically designed to be stateless, which means that each client request is treated as an independent transaction, without relying on prior interactions. This stateless approach enhances scalability and resilience, allowing services to handle requests independently and be easily replicated across different environments. Both SoA and microservices contribute to the overall objective of creating flexible, maintainable, and scalable software systems by leveraging modular service composition. Moreover, already existing software components or systems can be leveraged to compose new applications. This composition can extend over the Internet, making it a crucial strategy for large-scale systems where building from scratch is both costly and time-consuming. Furthermore, this approach is not exclusive to large-scale solutions; domain-specific software development can also benefit from composition, reducing costs and accelerating time-to-market [33].

2.1.4 Variability Management

Variability in systems and software engineering refers to the ability of a system or software product line to be efficiently extended, changed, customized, or configured for use in a particular context, enabling the creation of different product variants that share a common core but differ in certain aspects to meet specific customer requirements or market demands [34].

Variability management plays a crucial role in software engineering by facilitating the efficient development and maintenance of a range of related software products. Effective variability management involves the identification, modelling, and management of the commonalities and differences among products. This process is essential for optimizing reuse and ensuring flexibility in product customization. Pohl et al. [35] emphasizes the significance of variability mechanisms, such as feature models, decision models, and configuration management, in systematically addressing the complexities associated with product variations. By leveraging these mechanisms, organizations can strike a balance between standardization and customization, thereby reducing time-to-market and enhancing product quality.

While feature models and decision models provide high-level strategies for managing variability, practical implementation also requires robust mechanisms for integration at different levels of granularity. The need for modular off-the-shelf connectors and their role in variability management is discussed in-depth in Kaya et al. [36]. This study introduces connectors as essential mechanisms required at different levels of granularity, from individual components to complete systems. These connectors are critical in creating reusable structures across varying levels, proving their value through multiple use cases.

Initially, the decomposition of components provided a structural view by mapping the relationships between required and published methods. However, the dynamic aspect, which indicates the activation order among these connections, was later addressed in the form of a collaboration diagram [37], inspired by UML. This dynamic perspective superimposed numbering on message flows, offering enhanced clarity in the sequencing of operations. Kaya [38] introduced a textual modelling language within the Component Oriented Software Engineering (COSE) framework, integrating process flow primitives into packages to represent distributed process models. Building on this foundation, Cetinkaya et al. [39] extended the framework by incorporating connectors into the process model, further enriching the COSE framework and providing a more nuanced approach to managing system interactions.

Kaya [40] later expanded the concept by incorporating SoA-inspired process flows, which aligned the framework with modern software architecture paradigms. This approach reintroduced connectors with more significant roles, allowing them to facilitate variability and integration effectively [41]. These enhancements established reusable connectors as enabling technologies for building software systems through integration, completing the structural spectrum. Other related works have advanced the configuration capabilities of connectors, integrating variability management more systematically into the development process [42, 43, 44].

2.2 Related Work

To the best of our knowledge, our approach is one of the pioneering efforts to address the calibration process holistically. Existing studies, tend to focus on specific aspects or phases of calibration. As a result, there are only a limited number of studies available for comparison.

IoMT was presented as a layered IIoT architecture for metrology and calibration industry applications [26]. The three layers are composed of the *physical layer*, the *cloud services layer*, and the *application layer* as shown in Figure 4.1. When the architecture was initially proposed, the cloud services layer lacked a microservices vision and corresponding implementations. This work enhances the IoMT architecture with the microservices perspective, and its applicability is shown with the AutoRF-Power application.

Zet et al. [45] describe an automated process for calibration and DCC generation using blockchain technology to improve accessibility among stakeholders. In contrast, our system utilizes cloud technologies, offering enhanced flexibility and scalability. Our incorporates microservices to integrate various measurement types, reducing interoperability issues. This architecture supports the easy integration of different standards and measurements, without being constrained by the inherent limitations of blockchain technology.

Oppermann et al. [46] introduce the “operation layer” at PTB (Physikalisch-Technische Bundesanstalt) to enhance domain workflows using a cloud-native, distributed microservices architecture. This layer streamlines processes, breaks down data silos, and automates the creation of DCCs by connecting laboratory workflows with administrative data. While their approach provides a broad solution for managing and improving workflows in the domain, including the DCC workflows, we propose a more focused tool for the calibration process providing details at the implementation level.

Pontarolli et al. [47] conducted a study in the field of Industrial Automation Systems titled "Microservice-Oriented Architecture for Industry 4.0." Their focus was on using microservices to enhance industrial applications. They implemented their approach using the Moleculer framework [48], an open-source microservices framework for Node.js. The study demonstrates the significant benefits of microservice architectures in integrating advanced technologies such as IoT and cloud computing within industrial settings. While Pontarolli's work addresses general industrial automation, our approach specifically targets the calibration industry. We integrate various equipment types and communication technologies to ensure efficient and accurate calibration processes. Additionally, our architecture utilizes a Continuous Integration/Continuous Delivery (CI/CD) pipeline enhanced with a Textual Variability Model (TVM) for dynamic product configuration to ensure the ability to adapt to changes while boosting the system's responsiveness.

The study conducted by Nummiluikki et al. [49] discussed the implementation of DCCs within an industrial setting. Their research, part of a Proof of Concept (PoC) project involving multiple partners, aimed to test the feasibility of DCCs in a fully digitalized calibration environment. The PoC focused on creating a digital environment for calibration data generation, transfer, and usage, including DCC validation and digital signatures. Although their approach effectively demonstrated a digitalized calibration process, our proposed architecture extends this concept by integrating uncertainty calculation services to work alongside the implemented DCC generation/authentication services via leveraging microservices to enhance scalability and flexibility.

The cloud-side implementation of AutoRFPower contains two main services: uncertainty calculation and DCC generation. There is related work making uncertainty calculations available on the internet, such as the NIST Uncertainty Machine (NUM) [15]. NUM is an online tool based on a client-server architecture that performs uncertainty calculations based on LoP and MCS. Unlike our approach, which integrates measurements from physical equipment and generates DCCs in an IIoT environment, NUM focuses on server-side calculations without any consideration of hardware-level measurements or producing DCCs. Another example is Metas.UncLib [50]. It is a stand-alone desktop application that can be used to solve complex problems related to metrology, so it's not comparable to our approach.

A DCC is the digital version of a calibration certificate. Since the calibration certificate is an important output of the process [27], it circulates among the stakeholders in the industry, including customers, calibration labs, and national metrology institutes. Therefore, digitizing and standardizing this document is crucial. This effort is pioneered by PTB by providing a structured format for DCCs [16], [51]. We aim to generate PTB DCC-compatible calibration certificates and currently adopting their standard as much as possible. On the other hand, our architecture is designed with extendibility in mind, so we can incorporate new standards if and when they emerge.

The study presented in [52] aims to establish a domain-oriented development environment by integrating feature and process models into a well-defined architectural framework. This approach utilizes variability management for product configuration, employing a feature model-driven methodology and microservices.

While MSDeveloper is crafted as a domain-oriented development environment by integrating feature and process models into a layered architecture, our approach is specifically designed to tackle the unique challenges in the calibration industry. By incorporating a TVM, our architecture dynamically manages the heterogeneity of calibration equipment and processes. This model enables real-time adjustments to calibration settings and hardware configurations. Integrated into a microservice-based cloud architecture, this approach not only enhances system flexibility and scalability but also efficiently manages data flows and the generation of digital calibration certificates (DCCs), providing a streamlined and adaptable solution for the metrology domain.

In this thesis, an enhanced version of the work presented in [28] is demonstrated in a detailed manner. Besides, a broader background and literature review, implementation details of the uncertainty calculations, the workflows of the calibration and the authentication processes, a case study demonstrating the applicability of the approach and the validation of the used uncertainty calculation techniques are presented for the first time in this study. Moreover, [28] was focusing on the software architecture aspect of the proposed system; however, in the present study, we are additionally focusing on the details of the calibration processes starting from gathering data from devices such as power sensors.

CHAPTER 3

AUTOMATIC RF POWER MEASUREMENT AND SUPPORTING SOFTWARE

The material presented in this chapter has been derived mainly from [1].

In this chapter, automatic RF power measurement is explained in detail, with a focus on the devices used and the algorithms implemented to achieve accurate and reliable measurements. The chapter provides an in-depth presentation of the software we have developed, which includes a thorough discussion of its architecture, specifications, and user interfaces. This software, known as AutoRFPower (ARP), was developed in collaboration with TÜBİTAK National Metrology Institute as part of a broader effort supported by The Scientific and Technological Research Council of Turkey (TÜBİTAK) under Grant No. 5200040, entitled "A New Method and Software Development for Automatic RF Power Measurement and RF Power Meter Calibration."

The ARP system represents a significant advancement in automating RF power measurements. Its development has been guided by the need for metrologically traceable measurement systems that minimize user errors and account for the effects of the devices used on the measurement results. These considerations are thoroughly discussed in Section 3.1, where we provide background information on RF power measurement and present the underlying algorithm and measurement setup used in the ARP system.

In Section 3.2, we delve into the architecture of the ARP software, illustrating how it is structured to operate efficiently within a local environment. The software's specifications and user interfaces are also demonstrated.

The chapter concludes with a discussion 3.3 comparing the ARP software with existing commercial RF power measurement software. This comparison highlights the unique capabilities of the ARP software, including its compatibility with a wide range of devices, its focus on uncertainty calculations based on the Guide to the Expression of Uncertainty in Measurement (LoP), and its ability to perform measurements without the need for special measurement devices. These features, along with the software's robust architecture and user-friendly design, position the ARP system as a powerful tool for both research and practical applications in the field of RF power measurement.

3.1 Automatic RF Power Measurement

For accurate and reliable RF power measurement, it is necessary to use metrologically traceable measurement systems that are free from user errors and that take into account the effects of the devices used on the measurement [53]. In order to minimize the errors caused by the user, in RF power measurement systems, it should be preferred that the measurements be performed automatically rather than manually. In this chapter, background information about RF power measurement and the software developed for automatic RF power measurement are provided. Then, automatic RF power measurement values with power sensors at different frequency values and the metrological advantages of sequential and multiple measurements for automatic power measurement systems are presented. The automatic RF power measurement algorithm and the measurement setup, the architecture of software and its specifications and requirements, and also the advantages and disadvantages of the AUTORF-POWER (ARP) compared to commercial products are presented.

3.1.1 Automatic RF Power Measurement Setup

A typical RF power measurement system consists of an RF signal generator, a power sensor and a power meter. A generic RF power measurement system is depicted in Figure 3.1. Different power sensors such as thermocouple, semiconductor diode and thermistor can be used as RF power sensors. It is necessary to select the power sensor in accordance with the measured power and frequency, and use a power meter compatible with the power sensor [54]. To measure RF power automatically, software is required to control the operations of the computer and the measurement system devices as indicated in Figure 3.1. GPIB (General Propose Interface Bus) adapter and protocol should be used for communication between the computer and the devices [55]. The realization of such a system is illustrated in Figure 3.2.

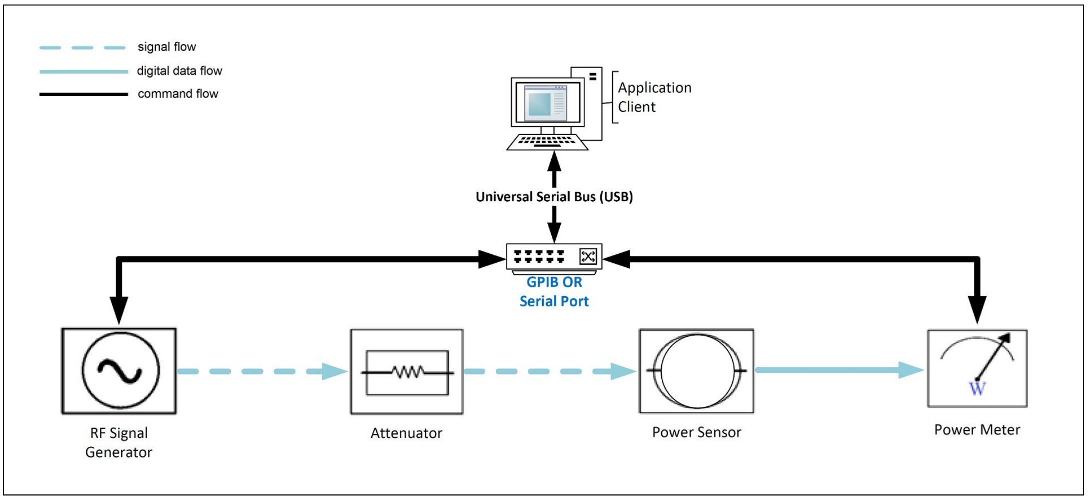

Figure 3.1: The measurement setup and its connections [3]

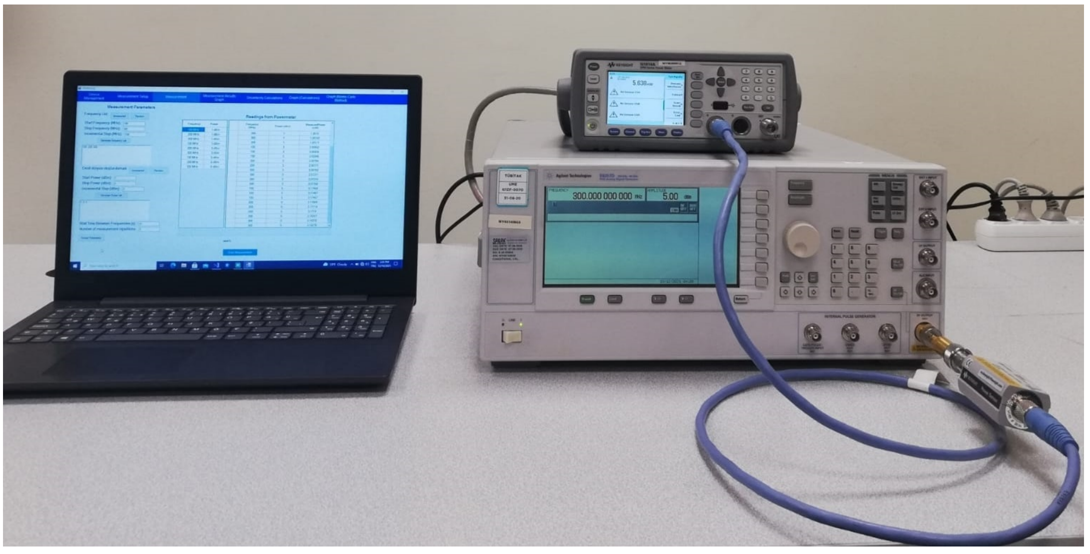

Figure 3.2: Realization of a measurement setup [1]

3.1.2 Automatic RF Power Measurement Algorithm

RF power is calculated with Equation (3.1)after completing the measurement. The inclusion of an attenuator in the measurement setup is optional. In case of using a test setup without an attenuator, the $S_{21\text{A}}$ term will be excluded from the equation.

$$P_{\text{out}} = \frac{P_{\text{read}}}{\text{CF}_{\text{std}}} \left| \frac{1}{S_{21A}} \right|^2 |1 - \Gamma_A \Gamma_{\text{out}}|^2 \quad (3.1)$$

where;

- P_{DUT}: Power of Device Under Test (DUT) RF signal generator,
- P_{READ}: Power Average of reading power from power meter,
- CF_{STD}: Calibration Factor of Standard (STD) power sensor,
- $S_{21\text{A}}$: Forward transmission coefficient of the attenuator (complex),
- Γ_{A}: Complex equivalent reflection coefficient of the input port of attenuator(s) in Figure 3.3 (reflection coefficient of the STD power sensor input (Γ_{STD}), if an attenuator is not placed on the setup)
- Γ_{DUT}: Complex reflection coefficient of DUT signal generator in Figure 3.3

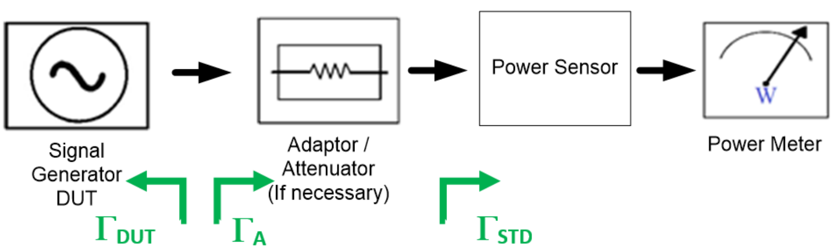

Figure 3.3: Reflection coefficients in power measurement using power sensor and attenuator [1]

3.2 ARP: Automatic RF Power Measurement Software

3.2.1 Architecture

Figure 3.4 depicts the implemented software's architecture designed to run in the local environment. The system is divided into layers, and the system assets are shown in corresponding layers. There are two classes of users. The operators are customers who are using the software for power measurement. The developers are responsible from introduction of new hardware, handling of compatibility issues between devices included in test setups as well as the maintenance of the system. The physical layer represents the hardware that are used in the power measurement setup(s), as illustrated in Figure 3.2.

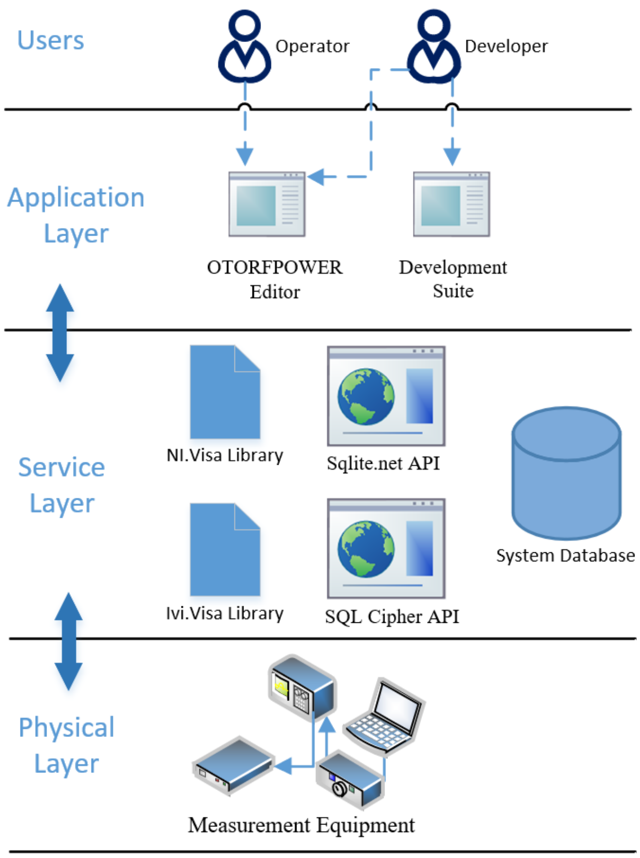

Figure 3.4: The ARP software architecture [1]

The application layer represents the editors, forms, GUI elements of the software and the tools used in the development. The ARP software was developed in C# [56] programming language using Microsoft Visual Studio 2019 (MVS) [57]. The user interface (UI) of the software has been designed and implemented using Windows Forms, which is a .NET UI Framework.

The service layer consists of elements that handle the interactions between different components of the system along with external elements used in the system such as APIs, software libraries and database. The ARP software incorporates an SQLite database in order to store device and measurement data. The database is generated using a sqlite-net API [58]. In addition, each database file generated by ARP is encrypted with the SQLitePCLRaw cipher [59] and password protected. Both APIs are shared as a reusable code and integrated to the MVS IDE via the NuGet package manager. A running instance of the ARP works with a relational database file that contains information on the devices registered to the system along with their properties and specifications. It's saved on a specific location in the file system and this file can be transferred to another computer to be utilized in another measurement or setup. The communication between the software and the devices in the physical layer is handled using the IEEE 488.2 protocol [60] with the use of Standard Commands for Programmable Instruments (SCPI) developed by the Ivi Foundation [61]. All messages within the test setup presented in Figure 1 are interchanged through a GPIB adapter. The Ivi.visa [62], and the NI.visa [63] libraries include the dictionaries for the SCPI commands that are used for passing information among the devices in the measurement setup using a Keysight and a National Instruments GPIB adapter, respectively.

The ARP and its components have been developed using the Model-View-Controller (MVC) pattern as presented in Figure 3.5. In our work, the View layer corresponds to the UI of the editor window. The objects in UI can be passed to the Controller to be modified in the database later. View layer also takes the inputs from the user to be implemented in the Model layer. The Controller layer acts as an operator and handles the interactions between Model and View layers. The Model layer corresponds to the software elements that provide the structure of the data along with a relational database, which includes both device and measurement data. Moreover, the corresponding assets, such as device parameters and settings for the algorithms used for the measurement, are employed in this layer.

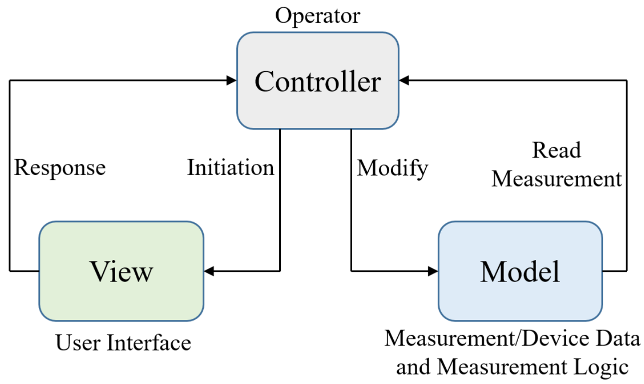

Figure 3.5: The MVC pattern used in the development [1]

3.2.2 Specifications and User Interface

The ARP editor offers three main windows to end-users, each with a different objective. The functions of each of these windows and their respective features are shown in Figure 3.6.

The device management is used to register new devices to the system with presets determined by the developers as well as edit or delete registered devices. When a device is registered to the system, generic information of the device, such as model, maker information, identifier set by the operator, serial number, certificate number/date, supported frequency/power range, GPIB address and connector type(s) are saved in the database for future use. Some of this information can be crucial to prevent possible operator errors. Also, there is some specific information based on the device type that is stored to be used in the uncertainty calculations and DCC generation.

The User Interface (UI) for the Signal Generator management form is given in Figure 3.7. Alongside the generic information, the operators fill the Standing Wave Ratio(SWR) and RC Incomplex fields according to the registered device's certification.

The Standing Wave Ratio (SWR) is a critical parameter in RF measurements, used to quantify the level of impedance mismatch in a transmission line. Impedance mismatches can result in reflected waves, leading to inaccuracies in power measurements.

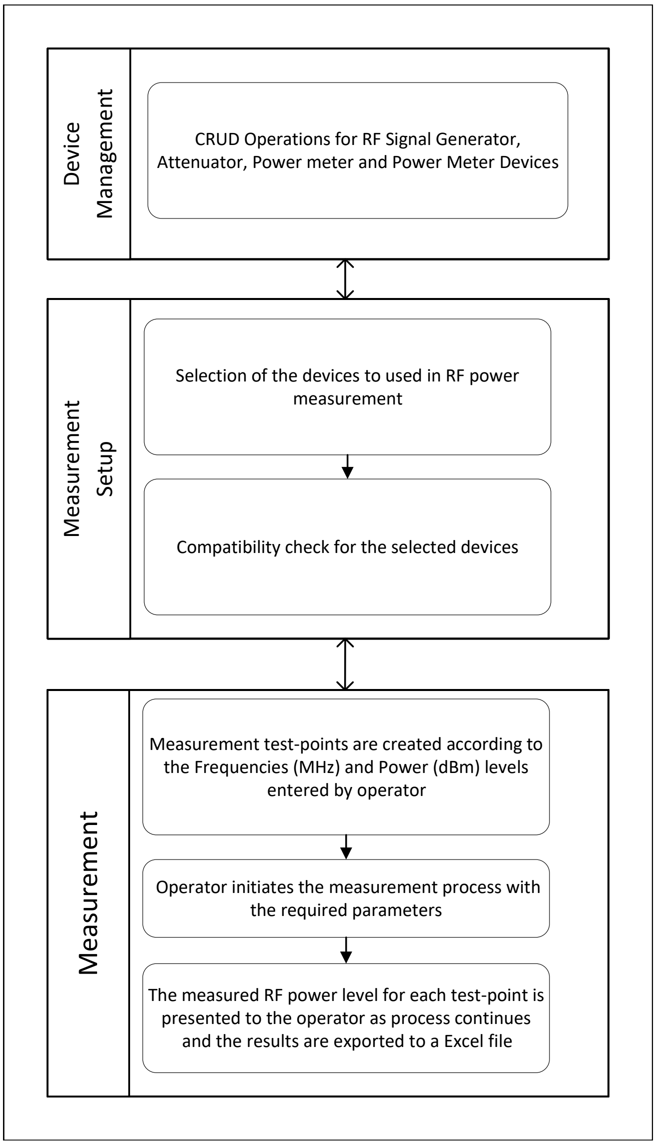

Figure 3.6: The ARP software process for automatic measurement [1]

The SWR field in the ARP software is specifically designed to calculate the uncertainty associated with these impedance mismatches, thereby improving the accuracy of RF power measurements.

In this context, the SWR field requires three key parameters: the Starting Frequency (MHz), the Finishing Frequency (MHz), and the SWR Value. The Starting Frequency denotes the lower bound of the frequency range over which the SWR is evaluated, ensuring that the uncertainty calculations cover the entire relevant frequency range of the RF signal under test. The Finishing Frequency defines the upper bound of this

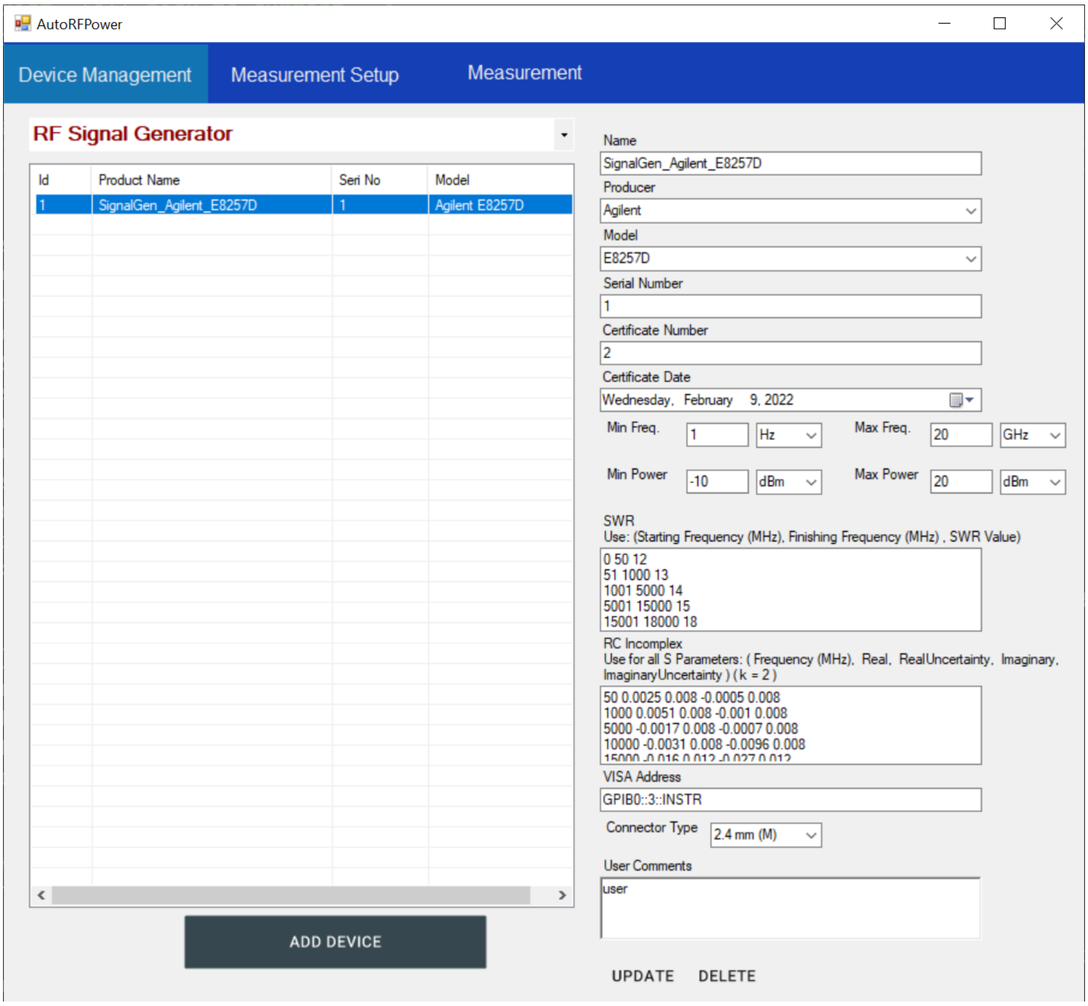

Figure 3.7: The UI for the Signal Generator management form

frequency range, ensuring comprehensive coverage.

The SWR Value represents the ratio of the maximum to minimum voltage (or current) in the standing wave pattern along the transmission line. A lower SWR indicates a better impedance match and reduces uncertainty in the power measurement.

The RC incomplex field is crucial for uncertainty calculations that consider both the real and imaginary components of impedance. As impedance changes with frequency, it's essential to precisely model both components for accurate RF power measurement. The information gathered here contributes to the overall uncertainty calculation, ensuring a comprehensive understanding of how impedance uncertainties impact the total measurement uncertainty. The RC incomplex field collects several parameters: Frequency (MHz), Real, Real Uncertainty, Imaginary, and Imaginary Uncertainty. The Frequency parameter specifies the frequency at which the impedance values are measured or calculated, recognizing the frequency-dependent nature of impedance. The Real component corresponds to the resistive part of the impedance at the specified frequency, while Real Uncertainty captures the associated uncertainty, reflecting potential deviations due to measurement limitations. Similarly, the Imaginary component represents the reactive part of the impedance, either inductive or capacitive, at the given frequency, with Imaginary Uncertainty indicating potential deviations. These parameters collectively contribute to a detailed uncertainty budget, essential for high-precision RF measurements. The uncertainties are reported with a coverage factor (k=2), corresponding to a confidence level of approximately 95%, providing a realistic estimation of the measurement's uncertainty range.

The user interface (UI) for the Power Sensor management form is given in Figure 3.8. Two critical fields contribute to the uncertainty calculations, namely the Calibration Factor and the Reflection Coefficient.

The Calibration Factor field plays a crucial role in adjusting the sensor's response to different frequencies. It is used to correct the raw measurements obtained from the power sensor, ensuring accurate power level results. This field comprises three attributes: Frequency, Calibration Factor, and Uncertainty. The Frequency attribute specifies the frequency at which the calibration factor is determined, recognizing that the sensor's response varies across frequencies. The Calibration Factor corrects the raw measurement at the specified frequency to obtain an accurate power reading. Additionally, the Uncertainty attribute associated with the Calibration Factor indicates the potential deviation from the true value, considering the limitations of the calibration process. This uncertainty significantly influences the accuracy of the power measurement within the overall uncertainty budget.

The Reflection Coefficient field is another critical element in the uncertainty calculations, addressing the impact of impedance mismatches between the power sensor and the device under test. This coefficient, which can be complex, describes the ratio of the reflected signal to the incident signal at a given frequency, influencing the accuracy of power measurements. It consists of five attributes: Frequency, Real,

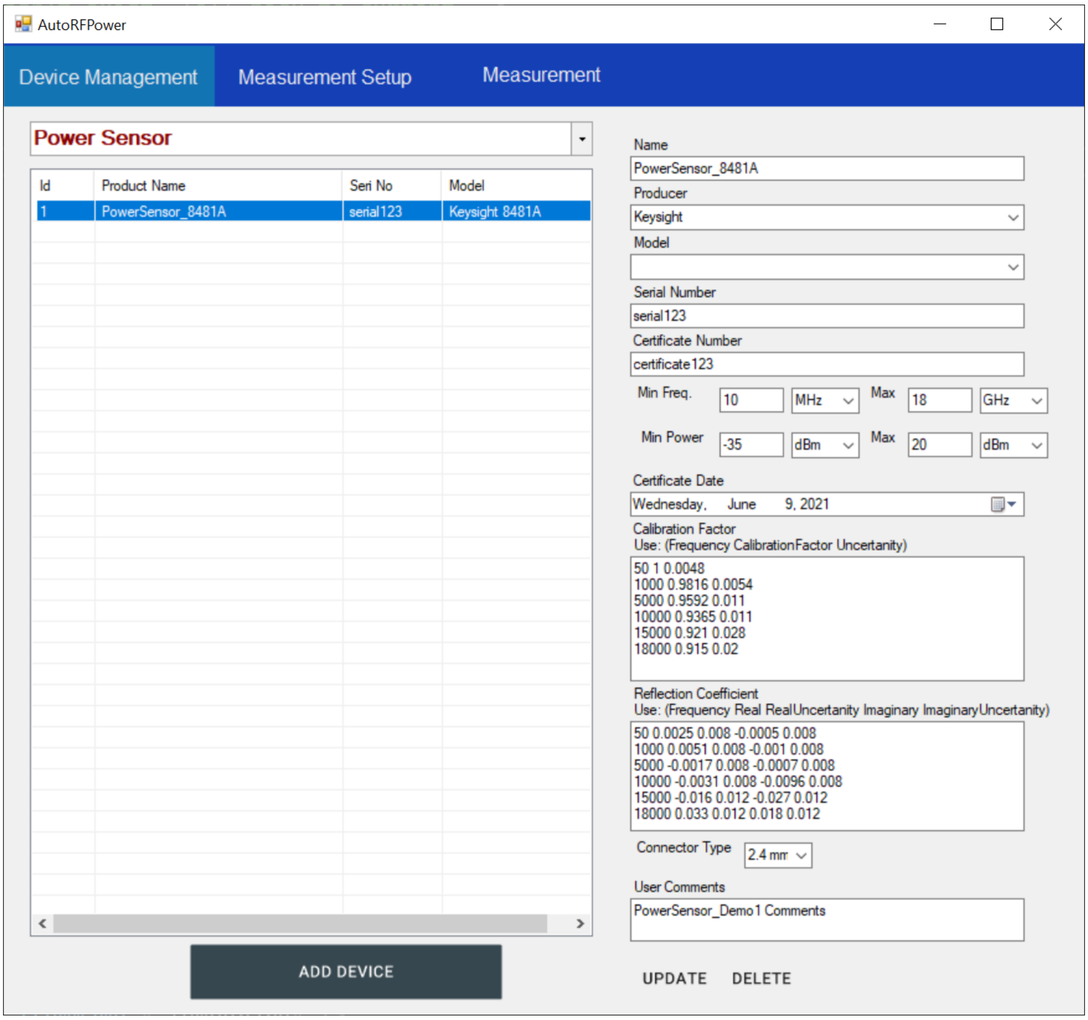

Figure 3.8: The UI for the Power Sensor management form

Real Uncertainty, Imaginary, and Imaginary Uncertainty. The Frequency attribute identifies the specific frequency at which the reflection coefficient is measured, ensuring frequency-specific calculations. The Real and Imaginary attributes represent the real and imaginary components of the reflection coefficient, respectively, defining the complex impedance mismatch. The Real Uncertainty and Imaginary Uncertainty attributes capture potential deviations in these components, reflecting the precision of the impedance measurement. These uncertainties significantly contribute to the overall uncertainty calculation by influencing the accuracy of the reflection coefficient and, consequently, the power measurement.

In the Power Meter UI, shown in Figure 3.9, several fields are integral to the process of uncertainty calculation, including Resolution, Accuracy, Ref Power RF Power Value, and Ref Power RF Power Uncertainty. These fields play a crucial role in ensuring the precision and reliability of RF power measurements.

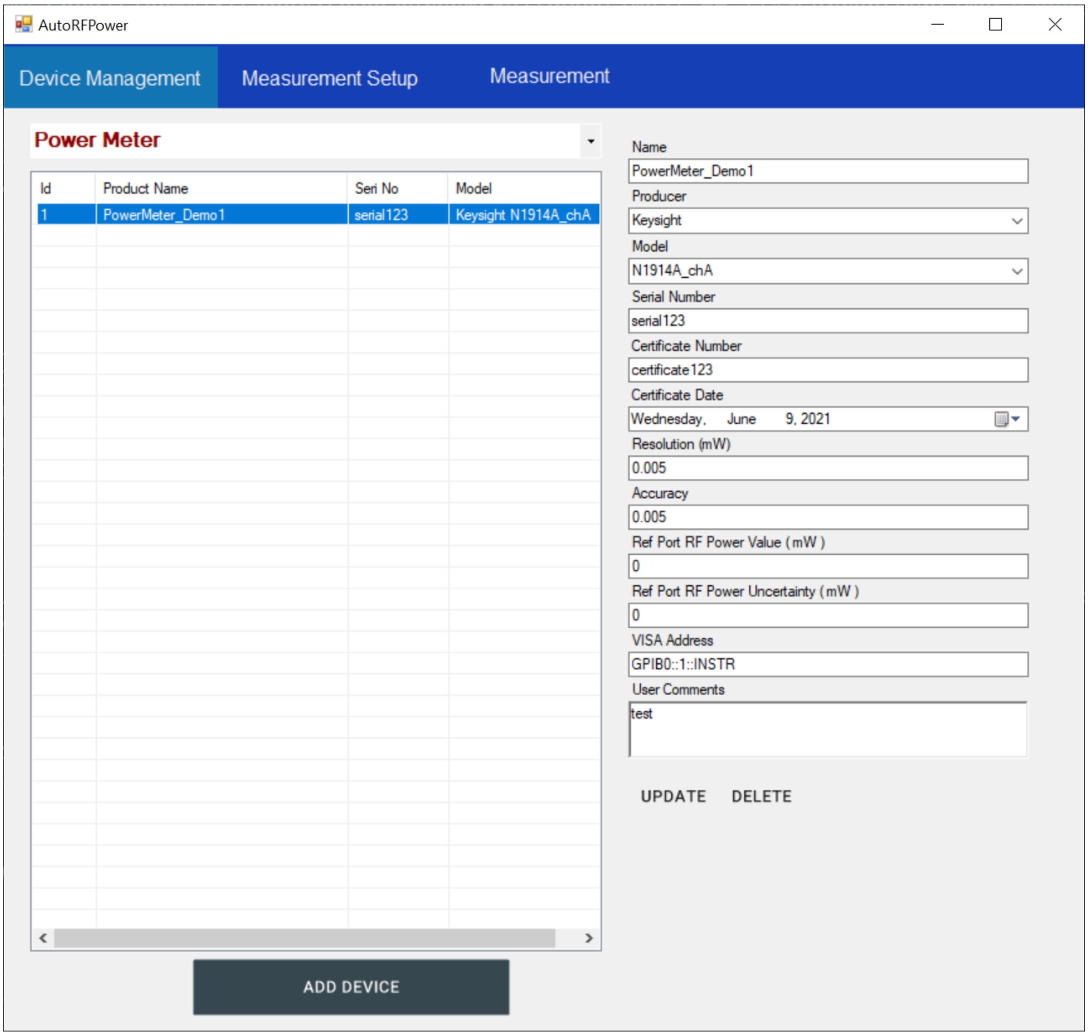

Figure 3.9: The UI for the Power Meter management form

The Resolution field, expressed in milliwatts (mW), represents the smallest measurable change in power that the power meter can detect. This parameter is crucial for understanding the granularity of the measurement, as it defines the meter's ability to distinguish between closely spaced power levels. A finer resolution indicates a higher capability of the meter to detect small variations in power, which is essential for applications requiring high precision. Moreover, the resolution directly impacts the uncertainty calculation, as a lower resolution might introduce a larger quantization error into the measurement process.

The Accuracy field represents the degree to which the measured RF power value corresponds to the true value. This parameter is typically provided as a percentage of the reading or as an absolute value in milliwatts. Accuracy is a key factor in the uncertainty budget, as it encompasses various sources of error, including systematic errors and the inherent limitations of the measurement system. Understanding the accuracy of the power meter is essential for evaluating the reliability of the measurements and for quantifying the overall uncertainty.

The RF Power Value (in mW) represents the reference power level measured by the power meter, serving as a standard for comparing other measurements. The accuracy and reliability of this reference power are crucial for calibrating and validating the measurement system. Additionally, the RF Power Uncertainty (in mW) quantifies the potential deviation from the true power level, considering factors such as measurement noise, environmental conditions, and the calibration process. This uncertainty is an essential component of the overall uncertainty budget, impacting the precision of subsequent measurements relying on this reference value.

The user interface (UI) for the Attenuator management form is given in Figure 3.10. The attenuator form includes all S parameters given below to be utilized during the uncertainty calculations (when an Attenuator is used in the measurement):

- The **S11 (Input Reflection Coefficient)** field represents the reflection coefficient at the input port of the attenuator. It is used to measure how much of the signal is reflected back towards the source due to impedance mismatches.
- The **S21 (Forward Transmission Coefficient)** field is crucial for understanding how much of the input signal is transmitted through the attenuator to the output. This parameter is particularly important for assessing the attenuation level and its impact on the signal's integrity.
- The **S12 (Reverse Transmission Coefficient)** field captures the reverse transmission coefficient, which measures the amount of signal that is transmitted in the reverse direction, from the output to the input. This parameter is essential for understanding the isolating properties of the attenuator.
- The **S22 (Output Reflection Coefficient)** field measures the reflection coeffi-

cient at the output port of the attenuator, indicating how much of the signal is reflected back from the output towards the attenuator.

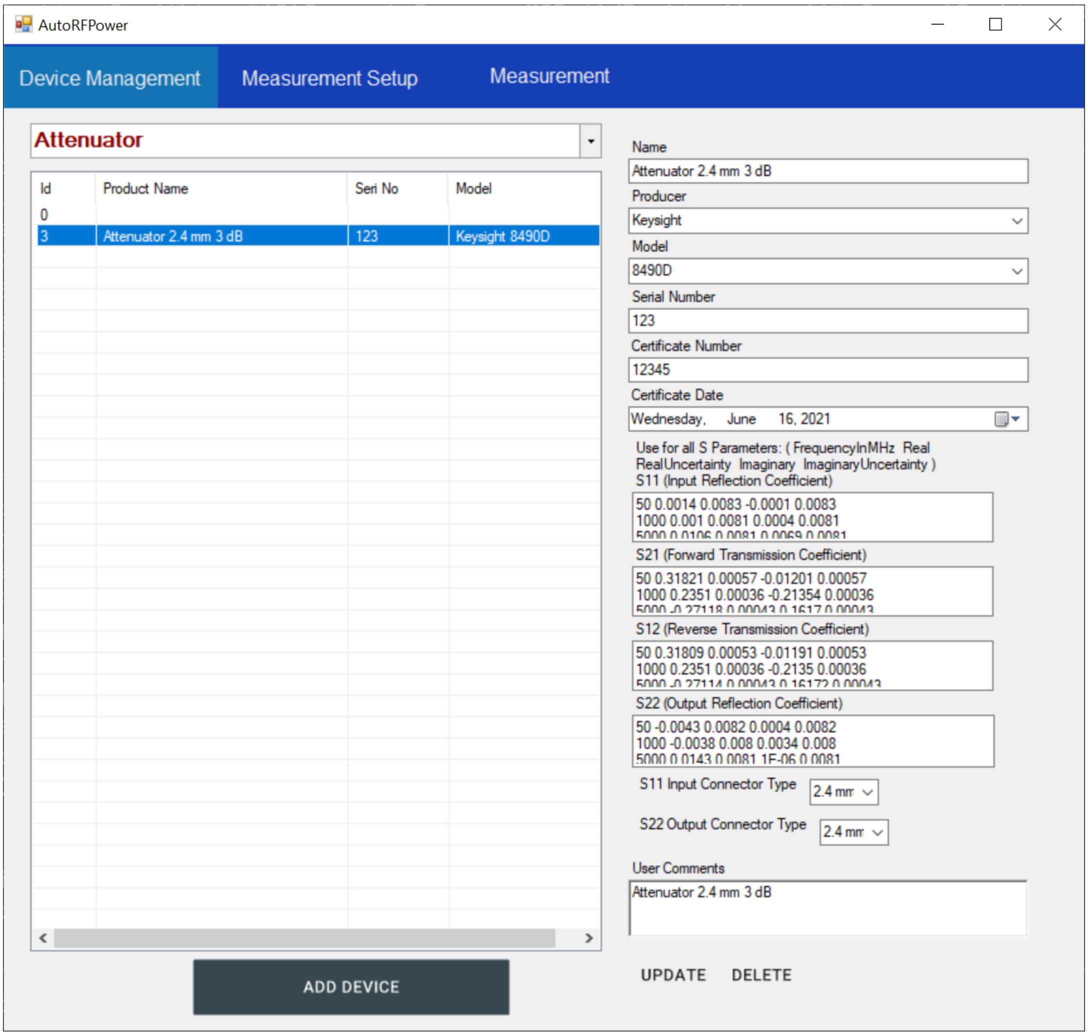

Figure 3.10: The UI for the Attenuator management form

When all devices that are going to be included in the automatic RF power measurement exist in the database with correct specifications, the measurement setup window presented in Figure 3.11 allows operators to select an RF signal generator, an attenuator (optional), a power sensor and a power meter to be used. The ARP checks the compatibility of the selected devices in order to prevent errors and failures. This step includes checking the connector(s) and power/frequency ranges of the devices in the test setup. If there are no problems with the current setup, the operator is allowed to proceed to the measurement window.

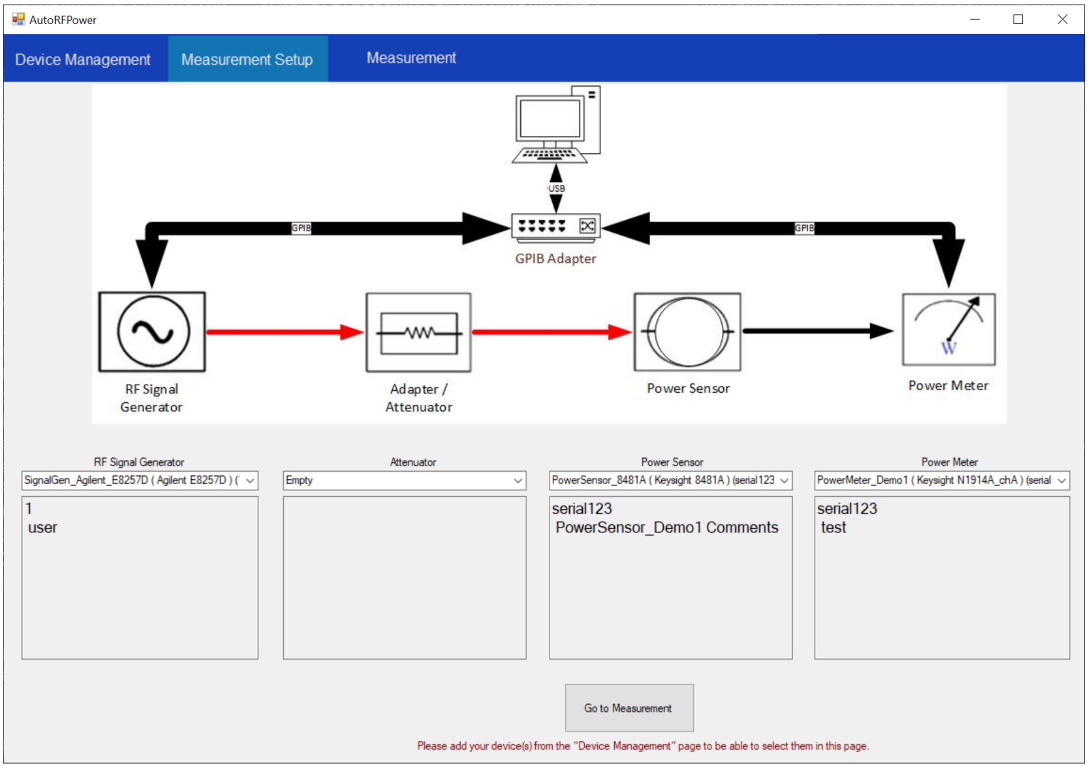

Figure 3.11: The UI for the Measurement Setup form

In the measurement form 3.12, the operator sets the test-points that will be used in the automatic measurement either manually or with the help of the buttons placed in the UI for automatic generation. When the selected frequencies and power levels are all entered, the operator provides measurement parameters such as wait time between each frequency/measurement and the number of repetitions for each test point. When all the required information is available, the operator starts the automatic RF power measurement, and the results are presented one by one for each measurement taken at a specific test point. In this way, the operator can observe and review the process. Finally, an Excel file with the results is created and saved on the computer.

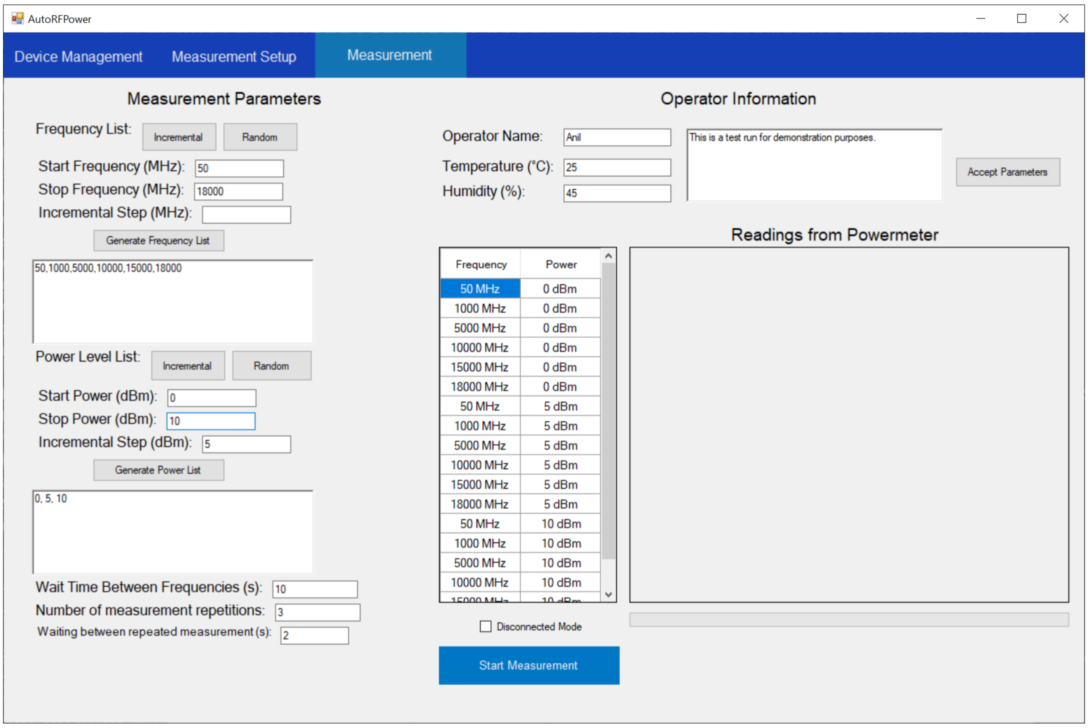

Figure 3.12: The UI for the Measurement form

3.3 Discussion

There are various commercial multi-functional software that has the capability of RF power measurement, some of the most popular ones and their suppliers are as follows:

- Virtual RF Power Meter Software (VPM3) from Bird [64]
- Metrology.Net (MDN) from Cal Lab Solutions [65]
- Benchvue PM software (BPM) from Keysight [66]
- Rohde & Schwarz NRP-Z Software (NRP) from Rohde & Schwarz [67]
- MET/CAL (MET) from Fluke [68]

The capabilities of the listed popular commercial RF power measurement software and the ARP are presented in Table 3.1. Where, Y means “Yes”, the software has the capability for that feature; N means “No”, the software does not have that capability;

Table 3.1: Comparison of Capabilities in RF Power Measurement Software (Adapted from [1])

	VPM3	MDN	BPM	NRP	MET	ARP
Automatic RF Power Measurement	Y	Y	Y	Y	Y	Y
Compatibility with various RF power sensors	N	Y	N	N	P	Y
Measurement capability in both dBm and mW	Y	N	Y	Y	Y	Y
Requirement for a special measurement device	N	N	N	N	Y	N
Uncertainty calculations based on LoP	N	N	N	N	N	Y

P means "Partially", the software has that capability partially. Some of the software products have been developed by the device manufacturers, and their usage can be restricted to the devices they produce. This imposes a severe limitation on users as they are forced to either work with the devices produced by a specific producer or to purchase software that is compatible with the devices they own. Further, some of them require some special measuring devices, which, apart from the software cost, also bring the device cost to the overall budget. The ARP software is compatible with devices produced by various leading manufacturers and does not require a specific measurement device to be used. Many commercial RF power measurement software products do not produce suitable data for uncertainty calculations, since they are designed for practical measurements rather than data collection by choosing a measurement size that will facilitate metrological calculations. On the other hand, ARP has been designed from the ground-up with uncertainty calculations in mind and uncertainty calculations based on GUM are already under development.

Automatic RF power measurement allows faster, more reliable and higher-resolution tests compared to manual measurement. Additionally, automation of the testing process helps minimize errors. The ARP software enables users to measure RF and microwave power safely, quickly and accurately, both in production and in calibration processes. In order to achieve the best user experience, this software is developed with an intuitive IU that divides the measurement process into steps with the use of various forms according to the operational requirements of each step. Transitions within forms are controlled, and additional validation mechanisms are put in place to minimize operator errors. The needs of commercial and defence sector users have been carefully considered. The capability of having an encrypted database file that could be transferred to another test setup along with the device and test information improves the usability, portability and security of the software. Presently, the software allows for device definitions to be updated; as a future extension, new devices can be added to the system. This feature is useful for enhancing the software in terms of extensibility and helps with reducing the cost in the long term.

CHAPTER 4

PROPOSED MICROSERVICE ARCHITECTURE WITH VARIABILITY MANAGEMENT

The material presented in this chapter has been derived mainly from [27], [28] and [3].

In this chapter, the microservice architecture that extends the Internet of Measurement Things (IoMT) framework with the incorporation of microservices and variability management is presented. The aim of this chapter is to provide a comprehensive overview of the architectural enhancements, the system workflow, and the statistical techniques utilized within the presented cloud-based system.

The chapter commences by exploring the extension of the IoMT architecture through the adoption of a microservice-oriented approach, emphasizing the advantages of reduced coupling and independent deployment of functionalities. The microservices are designed to oversee access control and ensure scalability through advanced variability management techniques. This section will elucidate how these microservices are structured within the cloud services layer and how variability management is utilized to tailor access to various microservices based on user qualifications and specific measurement requirements. While the current implementation is specifically tailored to RF power measurements, the architecture is inherently adaptable and can accommodate a broad spectrum of measurement types, such as temperature and pressure. This flexibility is achieved through systematic variability handling, which streamlines configuration, enhances scalability, and bolsters security by granting users access based on their qualifications and eligibility.

Subsequently, the chapter explores the implementation of a Continuous Integration/-

Continuous Deployment (CI/CD) pipeline, which is crucial for the efficient deployment and updating of the system. This CI/CD pipeline, implemented using Google Cloud services, automates the entire software deployment process, from initial build to final deployment. This capability is particularly important given the frequent updates and new features introduced to the application, where maintaining precision and reliability in metrology processes is critical. The integration of automated triggers within this pipeline further enhances variability management, ensuring that any changes in the Textual Variability Model (TVM) or other configurations automatically initiate a build process. This process not only facilitates seamless updates but also ensures that new configurations and updates are applied to the system without interruptions.

The chapter also explores the system workflow, providing a comprehensive overview of the entire process, from the initial connection of measurement devices to the generation of DCCs. This section details the interactions between the various components of the system, including the role of microservices in managing uncertainty calculations and DCC generation, ensuring that the entire process is both efficient and accurate.

Finally, the chapter concludes with an examination of the statistical techniques used in the system. This section will provide insights into the theoretical foundations and practical implementations of these techniques, highlighting how they contribute to the overall accuracy and reliability of the system. Through these detailed discussions, this chapter aims to provide a clear understanding of the proposed microservice architecture and its enhancements through variability management within the IoMT framework.

4.1 Enhancing the IoMT Architecture through Microservices with Variability Management

The IoMT architecture is extended by adopting a microservice perspective and related technologies on the cloud layer. Figure 4.1 represents the overall architecture detailing the entities residing in the layers.

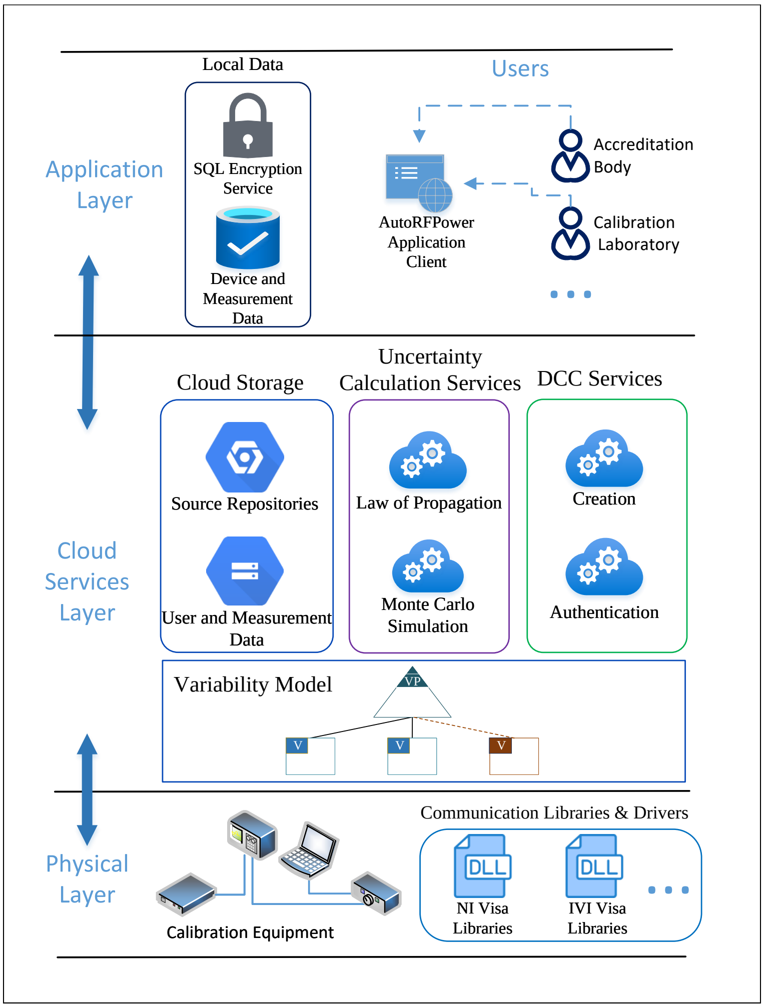

Figure 4.1: An abstract representation of the realized architecture and its components (Adapted from [3])

The application layer hosts the User Interface (UI) along with essential components that facilitate communication with the hardware in the lower layer. These components include device-specific drivers. Additionally, certificate information for the registered devices is collected from users and securely stored in an encrypted database. Communication with measurement devices is facilitated through these device drivers in the application layer, which interact with communication libraries located in the lower layers. The results of the measurements are then stored in the SQL database within the application layer.

The cloud services layer contains the uncertainty calculation services, the DCC services, and the cloud storage services. Both measurement and user data are hosted on Microsoft SQL Server in a container, ensuring high availability and secure data handling practices. Specifically, our architecture utilizes advanced encryption and access control mechanisms to secure data transmission and storage, while also facilitating uncertainty calculations and digital certificate generation. This layer not only supports high data throughput through the use of purpose-built microservice containers but also interacts seamlessly with lower layers, enabling the product configuration using variability models to be resolved and integrated across the system.

The physical layer contains the measurement setups, featuring a PC configured to control and process data generated by the registered devices, such as a Power Signal Generator, a Power Meter, a Power Sensor, and an optional Attenuator. All components in this layer work in coordination to enable precise measurement and calibration.

4.1.1 Utilization of a CI/CD Pipeline for Deployment

Figure 4.2 summarizes the overall architecture of the proposed system. Since we are using Google Cloud as a platform, we have adopted and adapted their recommended workflow for the development and deployment of containerized applications, utilizing a CI/CD pipeline [4]. This architecture enables a full-cycle development workflow that begins with collecting measurements from the hardware at the lowest level, transferring this data to the cloud environment, performing uncertainty calculations, and ultimately generating DCCs in compliance with standards set by regulatory bodies such as PTB.

There are three core Google Cloud services at the core of our CI/CD pipeline: Cloud Build, Artifact Registry, and Cloud Deploy. Cloud Build is a service that executes builds on Google Cloud infrastructure. It compiles source code, performs tests, and produces ready-to-deploy software packages. Artifact Registry is designed to store, manage and secure the container images and additional libraries. It facilitates consistent access control and integration with existing CI/CD tools by providing a centralized location for the custom-built software artifacts. Finally, Cloud Deploy automates the delivery of applications to specified Google Cloud environments. This service

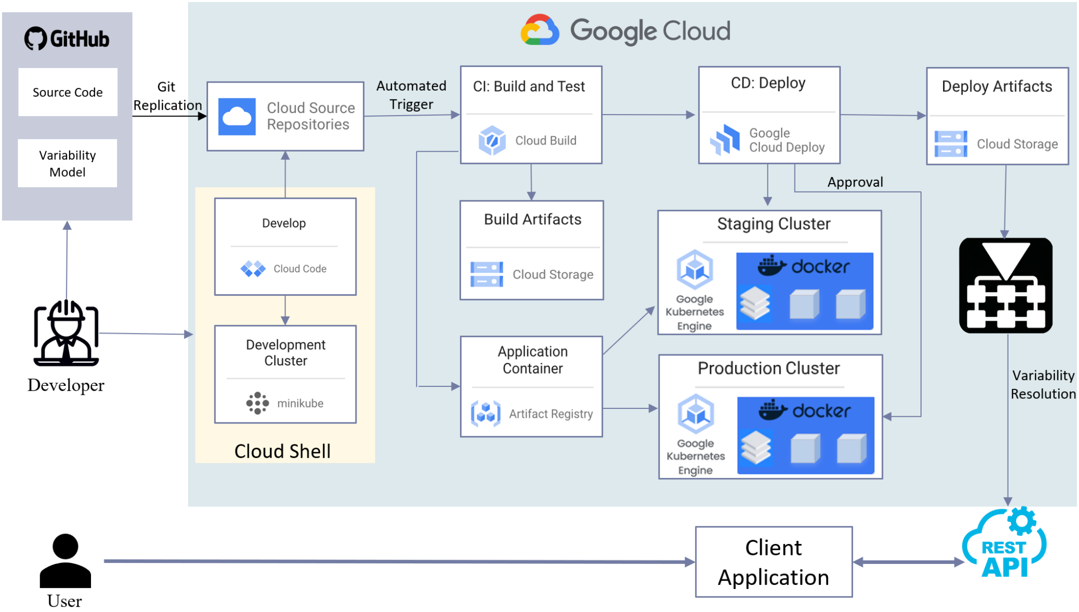

Figure 4.2: Overall architecture and the CI/CD pipeline (Adapted from [4]).

streamlines deployments, ensuring they are repeatable, predictable and secure across multiple stages of the production process. The continuous aspect of CI/CD is crucial for our application, allowing for seamless and frequent updates without disrupting the system. This capability is vital since the addition of new devices and features is frequent, and any disruption during the updates may result in downtime or loss of data integrity, which is unacceptable in calibration processes. Combined, these services automate the software deployment process from initial build to final deployment, improving the efficiency and reliability of the system's software development lifecycle.

In addition to the conventional workflow, an automated trigger has been integrated into the Cloud Build Service. YAML [69], a human-readable data serialization language, is used to handle the manipulation of configuration files. This trigger is executed whenever a Git replication occurs from a source Git repository to the Cloud Source Repository. In this way, every time when a change is made in the Google Source Repository or an update is committed via Git replication, a new build for the updated version is directly generated and staged under the deployment cluster ready to be deployed after approval without any interruption to the running instance.

4.1.2 Utilization of Variability Management for Product Configuration

Users of AutoRFPower are calibration laboratories that use different families of equipment with diverse communication protocols and libraries. This diversity is addressed by variations in code, hence the need for modeling and managing variability. Therefore, a customization mechanism is needed. Rather than doing the customization in an ad-hoc manner, we employed a systematic way of handling variability. To this end, an additional Textual Variability Model (TVM) has been developed and integrated into the source code. This TVM manages essential information such as device manufacturer/models for all device types, compatibility information among those devices and communication protocols/libraries, and access tokens for all microservices offered across the system. Each client/user has a configuration file that includes the variants/features assigned to them, regulating their permissions to access system components, various devices, and functions.

The UserModel in the TVM determines the access rights for users to specific system features. For example, For instance, setting attributes such as `MCS_Enabled = true`, `LoP_Enabled = true`, `RCI_Enabled = true`, `SWR_Enabled = true`, and `DCC_Enabled = true` allows a user to use the MCS methods, LoP for uncertainty calculations, perform measurements using the incomplex reflection coefficient values, perform measurements using the vectorial reflection coefficient with the SWR values, and DCC generation, respectively.

The DeviceModel in the TVM is responsible for managing user access to specific devices by controlling compatibility and access tokens. This ensures that only authorized personnel can operate or interact with particular measurement setups. The TVM effectively segregates user roles and device access, creating a secure and tailored environment that efficiently caters to different calibration scenarios. An illustration of the TVM in action is depicted in Figure 4.3, demonstrating the specification of user roles and device configurations within the model.

```
UserModel
    UserId = 1
    Username = "admin"
        MCS_Enabled = true
        LoP_Enabled = true
        RCI_Enabled = true
        SWR_Enabled = true
        DCC_Enabled = true

    UserId = 2
    Username = "operator"
        MCS_Enabled = true
        LoP_Enabled = true
        RCI_Enabled = true
        SWR_Enabled = true
        DCC_Enabled = true

    UserId = 3
    Username = "technician"
        MCS_Enabled = false
        LoP_Enabled = true
        RCI_Enabled = true
        SWR_Enabled = true
        DCC_Enabled = false

        .
        .
        .
```

```
DeviceModel
    DeviceId = 1
    DeviceType = "RF Signal Generator"
    ProducerModel = "Keysight E8257D"
    UserId = 2

    DeviceId = 2
    DeviceType = "Power Sensor"
    ProducerModel = "Keysight 8481A"
    UserId = 2

    DeviceId = 3
    DeviceType = "Power Meter"
    ProducerModel = "Keysight N1914A_chA"
    UserId = 2

    DeviceId = 4
    DeviceType = "Attenuator"
    ProducerModel = "Keysight 8490D"
    UserId = 2
.
.
.
    DeviceId = 9
    DeviceType = "RF Signal Generator"
    ProducerModel = "HP 83630A"
    UserId = 3

    DeviceId = 10
    DeviceType = "Power Sensor"
    ProducerModel = "R&S NRVD_chA"
    UserId = 3
```

Figure 4.3: Excerpt of the Textual Variability Model (TVM) showing user roles and device configurations.

The interaction between the Cloud Services layer and the Application layer is facilitated through the exchange of JSON-formatted files over HTTP requests using a RESTful API [70]. This approach provides a standardized and stateless communication protocol, simplifying the integration of various services and enhancing the scalability and maintainability of the system. Additionally, the utilization of RESTful principles allows our architecture to maintain efficient data exchange and real-time updates. For example, the transfer of configuration information from the client and the subsequent response from the cloud services layer, generated according to the TVM, enables dynamic resolution of existing variability in runtime. Furthermore, since the TVM is embedded directly into the source code, any changes made in this model automatically trigger a new build through the CI/CD pipeline, ensuring that updates are deployed without interrupting running services.

The automated triggers within the CI/CD pipeline significantly enhance variability management. These triggers ensure that changes in the TVM or other configuration aspects initiate a build process that deploys the updated services. This capability is crucial in a metrology environment where precision and reliability are paramount.

By combining variability management with automated triggers in the CI/CD pipeline, our approach enhances the system's responsiveness and agility. It enables the seamless integration of new services or modifications while maintaining high levels of accuracy and reliability in calibration processes. This automation reduces the risk of errors, ensures consistent service quality, and allows the system to efficiently process calibration data and generate DCCs with minimal manual intervention.

4.1.3 Uncertainy Calculation and DCC Generation Microservices

In the current implementation of our system, We have generated microservices for two different uncertainty calculation methods: LoP and MCS. Although the formulations of these calculations are provided by the International Bureau of Weights and Measures (BIPM) [23], there is no restriction on the tools and technologies that can be used for their implementation. Therefore, there are different programming languages and environments in their implementations creating heterogeneity. Given the fact that the calibration equipment requires specific libraries for communication, each technique can run in its own environment. Hence, having microservices for these implementations and containerization of them is a promising solution. For example, in our application, the LoP service is implemented using the C# programming language and its native libraries, while the MCS service is developed in Python 3, utilizing the open-source Pandas data analysis library. Moreover, the calibration equipment has its own environment encapsulated in the containers.

Figure 4.4 presents the dockerized components to host various microservices within the cloud environment which are orchestrated via Google Kubernetes Engine. Each microservice is encapsulated within its docker container, thus ensuring isolation while lowering dependencies, enhancing scalability and increasing deployment speed. Furthermore, the adoption of Docker containers to host microservices has many advantages such as; shortening delevopment time, improved fault tolerance, and provides

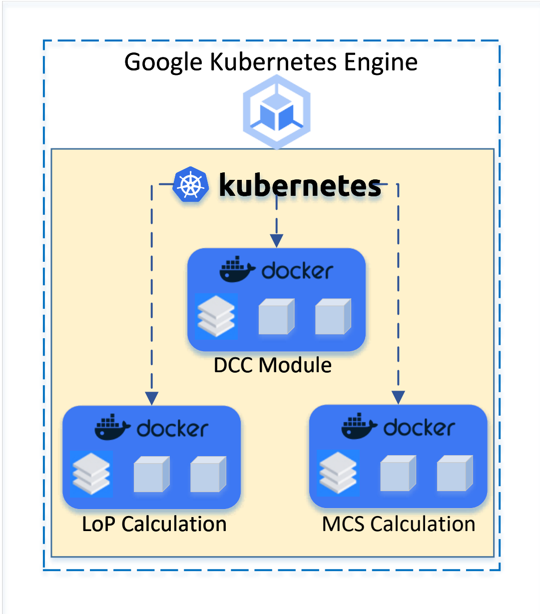

Figure 4.4: Dockerized containers in the cloud environment [3]

more a consistent environment during the development, testing and production [71]. Inclusion of the GKE for microservice orchestration ensures efficient resource management, automatic scaling and robust load balancing capabilities especially crucial for computationally heavy tasks such as MCSs. This configuration is not only useful for the optimization of operational efficiency, but also enhances the reliability and adaptability of the system accommodating agile responses to dynamic requests caused by rapidly advancing technology and industry's demands.

In the DCC microservice, we have employed the XML data format in compliance with the PTB's standardization of DCCs. Using XML facilitates well-structured and highly exchangeable data that supports seamless transitions when formatting DCCs based on the existing definitions and adapting to newly designed standards. After performing the uncertainty calculations either or both LoP and MCS microservices, the results are converted into the XML format and forwarded to the DCC microservice, with the utilization of REST API[70] and Flask framework [72]. Then, this XML file is parsed and converted into a human-readable PDF format in the cloud environment where users can access their certificates through an authentication service.

We have implemented a robust method for authenticating DCCs by leveraging UUIDs

(Universally Unique Identifier). These UUIDs are generated using a combination of the current timestamp and the Media Access Control (MAC) address of the machine, ensuring both temporal and spatial uniqueness. Each original document stored on the cloud storage is assigned a unique UUID, which is then used to generate a corresponding quick response (QR) code. When scanned, this QR code directly links to the original document on the cloud server, ensuring its authenticity. By embedding these QR codes in our DCCs, we provide a reliable means for verifying the integrity and originality of the documents. This method enhances security by significantly reducing the probability of duplication, as the identifiers are both time and hardware-specific. Implementing UUID-based QR codes thus ensures a high level of trust and reliability in the DCC authentication system.

4.2 System Workflow

This section provides a comprehensive overview of the RF power measurement implementation, covering the entire process from initial measurements taken in the physical layer (test setup) to uncertainty calculations in the cloud services, and concluding with the generation of DCCs in a cloud-based environment. This end-to-end approach utilizes an enhanced microservice architecture in support of IoMT to guarantee efficient and accurate calibration processes.

The process begins with the user (e.g. a technician or engineer) connecting the test setup, which consists of a Signal Generator, Power Sensor, Power Meter and an optional Attenuator to a computer running the ARP (client) application. This application can communicate with the test setup using both General Purpose Interface Bus (GPIB) adapters and serial communication ports via the Institute of Electrical and Electronics Engineers (IEEE) 488.2 communication protocol. A schematic of the test setup is shown in Figure 3.1. The necessary device drivers (such as NI Max, Keysight IO Suite, etc.) must be installed on the PC running the software.

The process model shown in Figure 4.5, drawn using BPMN (Business Process Model and Notation) 2.0, illustrates the user authentication and client configuration based on variability resolution. BPMN 2.0 is a graphical representation for specifying business processes in a business process model. It provides a standard way to visualize the steps in a process, which enhances clarity and communication among stakeholders.

When the devices are prepared for measurement, the operator launches the client application from the computer, which is connected to test devices using a GPIB adapter. The application starts with a login window and prompts the operator to enter their credentials. Then, the username and passport entered are forwarded to the Cloud Services API (CSAPI) and compared with the stored credentials. If the access is granted by the system, CSAPI returns a configuration file in JSON format. This file consists of information derived and transformed from a TVM, which is used for the configuration of the client application based on the features and devices that are allowed for that specific user. The access rights for users can be updated based on their qualifications by a system admin or an authorized person.

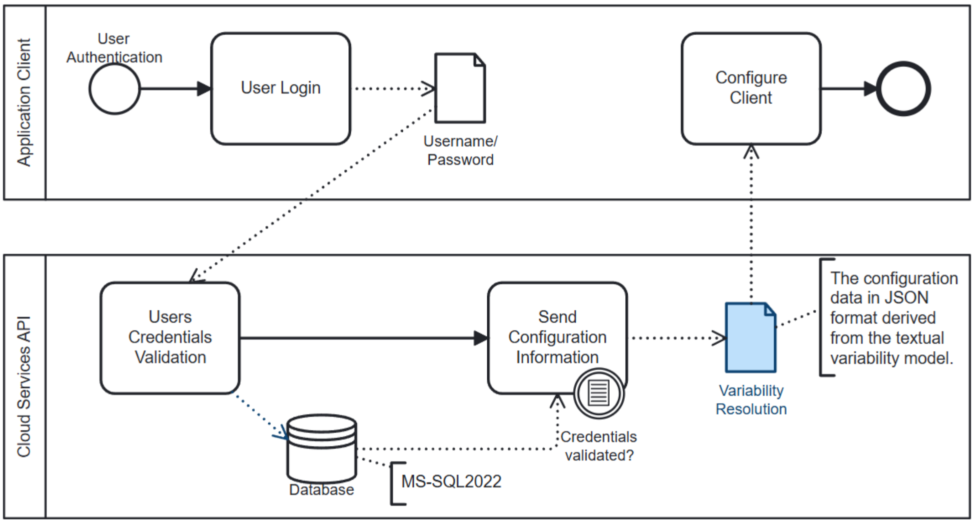

Figure 4.5: User authentication and client configuration based on variability resolution [3]

In a technology-intensive domain such as metrology, it is of utmost importance to ensure that only qualified personnel are authorized to operate specific equipment and carry out designated tasks. Inadequate qualifications can result in significant errors and potential misconduct, posing a risk to the integrity of the calibration process. Therefore, effective management of user access is essential. By restricting device operations and feature access based on user qualifications, the system guarantees that only trained and certified individuals can execute specific functions. This approach not only enhances the precision and reliability of the calibration process but also ensures adherence to industry standards and regulations. Upon finalizing the client configuration, the operator can initiate the measurement process. The process model illustrated in Figure 4.6 outlines the overall workflow of the system. It begins with automated RF power measurements on the client application and demonstrates the implemented microservices for uncertainty calculations and DCC operations, as well as the interactions among all these components. This model offers a comprehensive understanding of the system's end-to-end operations, promoting clarity in the sequence of actions and the roles of various components.

Automated RF power measurement commences with the operator selecting from registered devices stored in the local database. If any device from the measurement setup hasn't been registered, the operator registers that device to the system by entering specific information about the device, such as serial number, certificate information and any other required details for the uncertainty calculations. The process then continues with the determination of the measurement parameters. These parameters include the test points (power levels and frequencies to be measured), the number of measurements for each test point, and the waiting times between instances.

When the measurement is completed, all results, along with the setup & operator information, are combined, converted into a JSON file, and sent to the CSAPI with an HTTP request to be stored in a cloud database.

4.3 Statistical Techniques Used

Our system offers users the flexibility to conduct uncertainty calculations using either the LoP or MCS methods. With a simple click, users can trigger their chosen method, which then sends a request to the CSAPI to gather the essential data for the calculations. Once the data is transferred to the designated container, the calculations are executed, and the results are subsequently showcased in the application and stored in the cloud database within the cloud services container.

The LoP method, which is based on the principles outlined in the specifications provided in reference [23], utilizes analytical formulas to systematically evaluate the uncertainties associated with various measurement processes. This method provides a structured and standardized approach, making it widely utilized in various calibration practices. The implementation of this method is straightforward, as the core principles of GUM are consistent and universally applicable.

Grounded in the central limit theorem, the LoP calculates uncertainty by assuming that all input parameters contributing to the uncertainty calculation follow a normal distribution. If the input parameters originate from other distributions—such as rectangular, triangular, or U-shaped—they must be transformed into a normal distribution. Parameters that do not initially follow a normal distribution are treated as if they do after transformation. Specifically, when an uncertainty component is derived from rectangular, triangular, or U-shaped distributions, it should be divided by $\sqrt{3}$, $\sqrt{6}$, and $\sqrt{2}$, respectively, to achieve a normal distribution equivalent. This approach ensures that the uncertainty calculated using the LoP method is symmetrically distributed and approximates a balanced normal distribution. The combined uncertainty can then be determined using Equation (4.1):

$$u(k=1) = \sqrt{\sum_{i=1}^{n} (c_i^2 \cdot u_i^2)} \tag{4.1}$$

where, u(k=1) is the combined uncertainty with coverage factor one (68 % reliability), c_i represents the sensitivity coefficient of each uncertainty component in the model function (*f(.)*), and u_i denotes the uncertainty value of each component in the model function with normal distribution. The c_i can be computed using a partial derivative of the model function of the relevant uncertainty component $\left(\frac{\partial f(.)}{\partial i}\right)$. Figure 4.7 shows the visualization of the calculation flow of the LoP method.

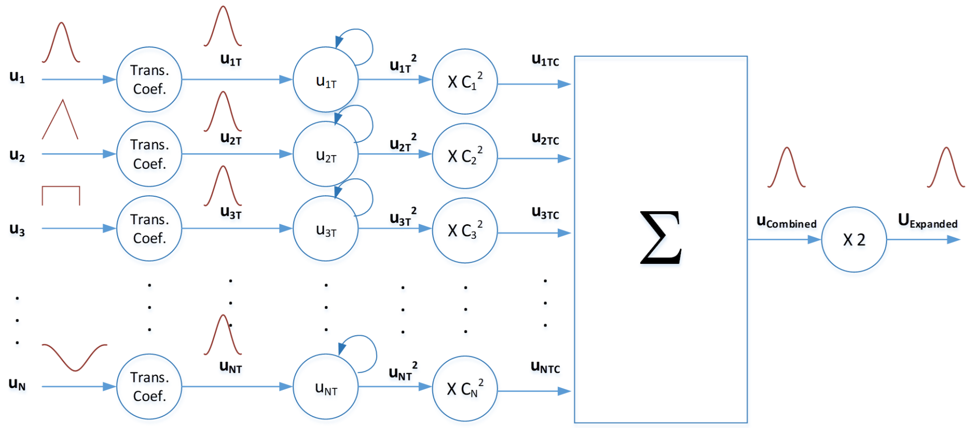

Figure 4.7: Law of Propagation of Uncertainty Process flow (Adapted from [2])

In the present study, the LoP method based on GUM was integrated into the cloud environment to handle uncertainty calculations for various calibration processes. The implementation followed the same principles, ensuring that all input parameters contributing to the uncertainty were accurately transformed into normal distributions.

MCS utilizes computational algorithms to model and propagate uncertainties through a large number of simulated trials, making it a stochastic method. By generating and analyzing numerous random samples, MCS provides a detailed statistical representation of measurement uncertainties. Its flexibility enables tailored application to specific scenarios and measurement complexities, making it a powerful tool in modern metrology. However, this flexibility also means that MCS implementation can vary significantly from case to case, depending on specific requirements and available computational resources. This adaptability is a key advantage of MCS, allowing for precise and customized uncertainty assessments across diverse calibration contexts.

In the present study, the MCS method was adapted to the cloud-based system, allowing for the simulation of uncertainty calculations using large datasets and extensive computational resources. The combined uncertainty was calculated using the equation (4.2):

$$u(k=1) = \sqrt{\sum_{i=1}^{n} u_{Ri}^2} \tag{4.2}$$

where, u(k=1) is the combined uncertainty with coverage factor one (68 % reliability), u_{Ri} represents the randomly generated uncertainty values of each component.

The MCS method is an analytical approach used to calculate measurement uncertainties. It is generally not the preferred method when the Law of Propagation of Uncertainty (LoP) can be applied, as performing a large number of repetitive measurements, such as 10^5 times or more, can be challenging [73]. To simulate real measurements, all input parameters for uncertainty calculation need to be randomized at least 10^5 times before performing the uncertainty calculation. In the MCS method, there is no symmetrical normal distribution transformation, and the effect of the input component's distribution can be observed in the combined uncertainty.The process flow is illustrated in Figure 4.8, providing a clear and structured visualization of the steps involved in evaluating uncertainty based on MCS in RF power measurements. This figure is essential for understanding the systematic flow of the algorithm and how each component contributes to the overall uncertainty analysis. The workflow begins with the Inputs (u_i), representing the initial data or parameters required for the simulation. These inputs are then transformed into Probability Density Functions (PDFs) $p(u_i)$, defining the underlying distributions that model the uncertainties.

Next, the process moves to the Parameterization of Distributions stage, where the Location and Scale parameters are calculated. The Location parameter typically indicates the central tendency of the distribution, such as the mean or expected value, while the Scale parameter reflects the dispersion or spread, such as the standard deviation. These parameters are crucial as they define the shape and characteristics of the probability distributions used in subsequent steps. In the Randomization stage, 10^5 random samples are generated for each distribution based on the previously defined parameters. This extensive sampling ensures a robust representation of the range of possible outcomes, allowing for a thorough analysis of uncertainty. The samples then undergo Statistical Analysis, where key metrics, including mean, standard deviation, and confidence intervals, are calculated to quantify the uncertainty in the system. Finally, in the Result Aggregation step, these statistical results are combined into a single, unified measure of uncertainty, referred to as u_{combined}.

Expanding on the detailed process flow outlined in Figure 4.8, we will delve deeper into the technical implementation of the MCS algorithm, especially when deployed as a container in a cloud environment. To gain a more granular understanding of how the MCS algorithm operates within this context, we present a Unified Modeling Language (UML) [74] sequence diagram in Figure 4.9. This diagram complements the process flow by illustrating the specific interactions and integrations within the MCS container, showcasing the algorithmic flow as it interfaces with cloud-based services.

The sequence diagram details the steps involved in quantifying uncertainties by simulating various potential outcomes based on input data. The MCS algorithm is implemented through several key stages: initializing variables, setting up necessary parameters, reading and processing input data files stored in cloud-based object storage, generating random variables to model uncertainties in the measurements, calculating confidence intervals for the simulated data, and saving the results back to the cloud storage for further analysis and reporting.

The sequence diagram for the MCS algorithm includes several key participants and interactions. The user initiates the simulation process, triggering the main process which orchestrates the entire simulation. The main process interacts with the object storage service, a cloud-based service used to store and retrieve data files. Pandas, a data manipulation library, is employed for reading and processing Excel files, while Matplotlib, a plotting library, is utilized for visualizing the results. Scipy.stats, a statistical library, generates random variables necessary for the simulation, and XlsxWriter, a library for writing output results to Excel files, stores the simulation results. Each of these components works seamlessly together to execute the MCS algorithm, ensuring efficient data handling, statistical computation, and result visualization and storage.

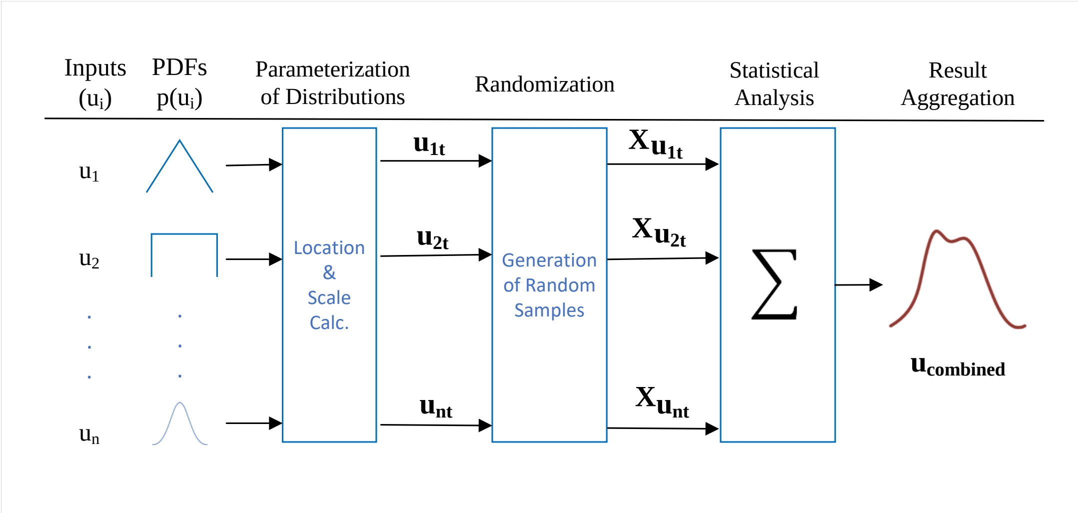

Figure 4.8: Monte Carlo Simulation Process for Uncertainty Calculation (Adapted from [2])

The process begins with the user initiating the main process, which requests a list of files from the object storage service. This entails querying the cloud storage for files that match a specific naming pattern recognized as an output of the client application. Once the target files are identified, the main process downloads the content of each file from the object storage service and reads it into a Pandas DataFrame, which serves as the primary data structure for managing and processing the input data. Additionally, the main process defines various helper functions, including `get_beta_distribution`, which calculates the statistical parameters necessary for the simulation.

The algorithm calculates the location and scale parameters for the beta distribution by iterating through the input data and stores these parameters for use in subsequent steps of the simulation. The `generalized_special_RVs` function is defined to generate random variables based on the input data and various statistical distributions essential for modelling the uncertainties in RF power measurements.

The process continues with the MCS loop iterating through each row of the input data. If a specific column value is zero, the row is skipped to avoid unnecessary computations. For other rows, the algorithm generates random variables using the defined function, calculates confidence intervals, and plots histograms using Matplotlib. These histograms offer a visual representation of the probability distribution of the simulated data. The calculated statistics, including mean and standard deviation, are then appended to respective lists for further analysis.

Upon completion of the simulation loop, the main process creates an Excel workbook using XlsxWriter, where it writes the results, including confidence intervals, mean values, and standard deviations. The completed Excel workbook is then uploaded back to the object storage service to ensure secure storage and accessibility for further analysis. The process concludes with the main process printing the total simulation time, marking the end of the MCS algorithm's execution. This series of interactions and computations effectively models the uncertainties in RF power measurements, providing valuable insights into the system's behaviour under various conditions.

Finally, the operators can generate DCCs based on the performed measurements and subsequent uncertainty calculations by sending a request to the DCC Microservice via the application UI. When triggered, DCC Microservice container retrieves the necessary data from the CSAPI. This data includes the user information based on the logged-in profile, all of the data on the devices selected in order to perform the measurement and uncertainty calculation results. The data is transformed into XML format inside the container to be shaped in compliance with the defined specifications/standards. Then the XML data is embedded into HTML tags to form a document which can be distributed/printed in ".pdf" format.

To ensure the created document can be authenticated by officials or third parties, a method called "UUID Generator" is called within the container. This method uses a combination of the exact timestamp at which the operation is initiated and the MAC address of the device to create a unique 32-character string that is hard to replicate. Then, it is appended to a prefix link that points to a running instance of the authentication service on cloud servers. A QR code containing this link is placed on each DCC, allowing each certificate to be authenticated using the cloud authentication service.

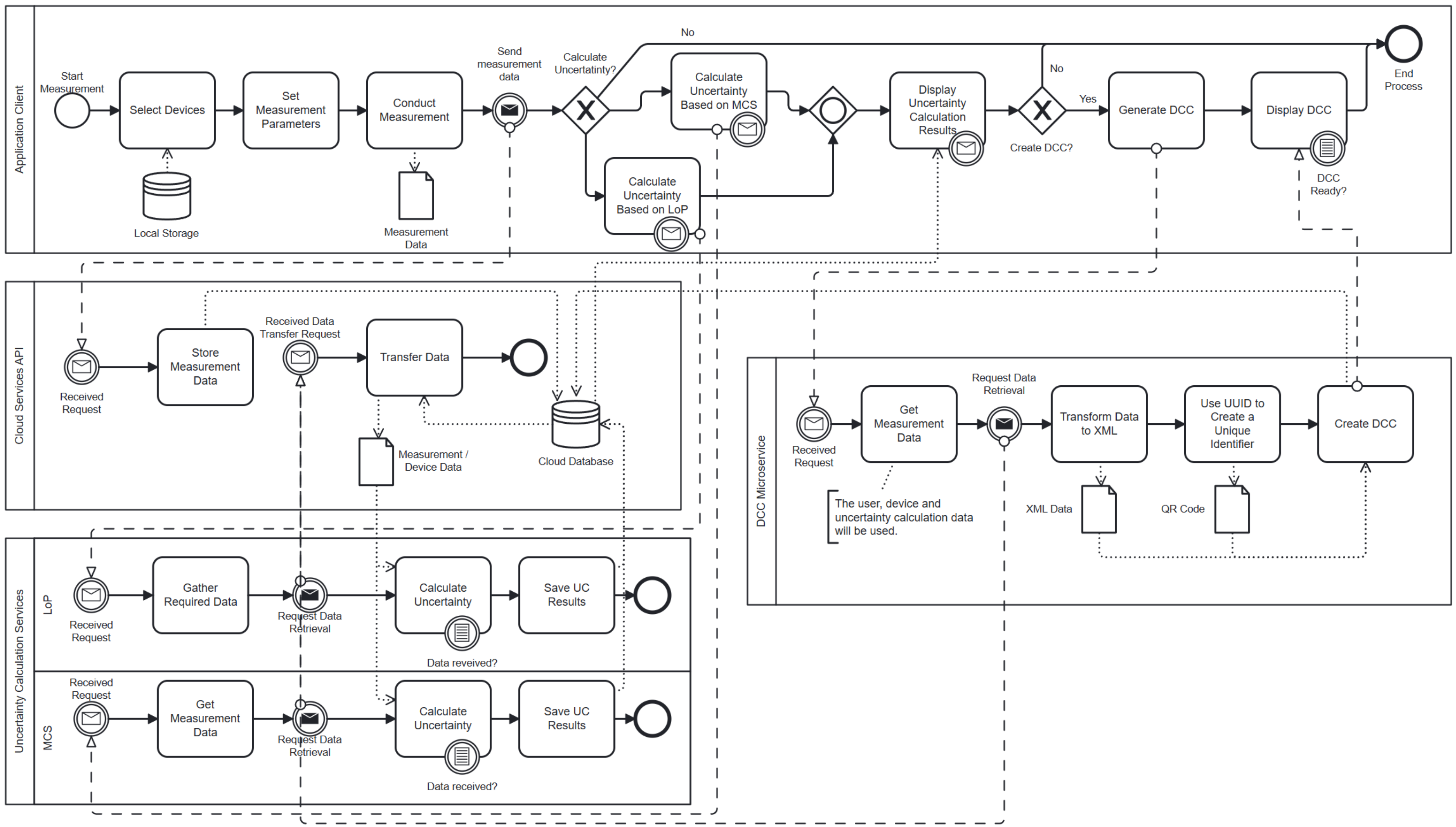

Figure 4.6: The process for the Uncertainty Calculation and DCC Generation [3]

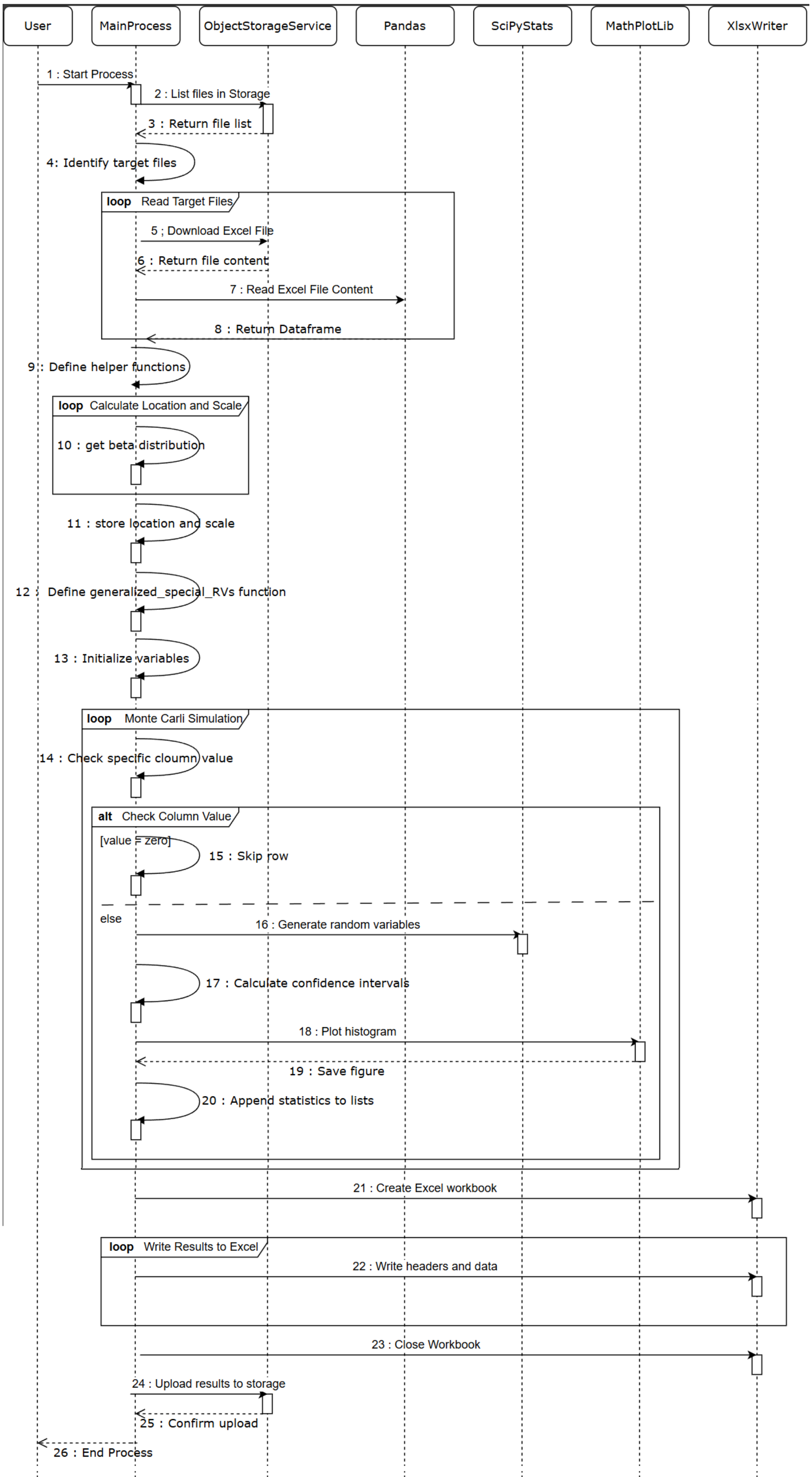

Figure 4.9: Sequence Diagram for the Monte Carlo Simulation [3]

CHAPTER 5

COMPREHENSIVE SYSTEM EVALUATION: CASE STUDY, UNCERTAINTY ANALYSIS, AND METHODOLOGICAL VALIDATION

The material presented in this chapter has been derived mainly from [2] and [3].

This chapter presents a comprehensive evaluation of our system, commencing with a detailed case study that illustrates its ability to integrate and manage measurement devices, conduct measurements, and perform uncertainty calculations within a cloud-based environment. Subsequently, we validate the accuracy and reliability of the uncertainty calculation methods by comparing them with results from a prior study [2] conducted in local settings. Additionally, the section includes a critical evaluation of the employed uncertainty calculation methods, namely LoP and MCS, highlighting their respective strengths and limitations. Finally, we discuss the broader implications of our findings, evaluate the system's constraints, and suggest future avenues for enhancing the system's capabilities, particularly in the realm of digital metrology and calibration.

5.1 Case Study: A Practical Application of the System

An Agilent E8257D signal generator, a Keysight 8481A power sensor, and a Keysight N1914A power meter were connected using a National Instruments (NI) GPIB adapter. The client application was started, and user credentials were entered to access the system. At this point, the client was configured in real-time according to the granted access rights based on the application of the TVM for our user. After configuration, only the devices available to us could be listed and selected. The devices in the measurement setup are registered to the system via the forms presented in Figures 3.7, 3.8, and 3.8, respectively.

Next, we opened the NI Max application to verify the communication addresses assigned by the operating system for the signal generator and power meter. Since these addresses are not permanent and may change based on the virtual environment of the computer (e.g., after each restart), we compared them and updated as necessary. Then, we moved on to the Measurement Setup interface and chose the devices to be used for the measurements. Once selected, we were prompted to enter the measurement parameters in the Measurement Form. We designated 24 test points for the measurements, including test frequencies of 50, 1000, 5000, 10000, 15000, and 18000 MHz, and power levels of 0, 5, 10, and 15 dBm. We also set the waiting times between measurements. Afterwards, we added remarks for the test and documented the environmental conditions: temperature in degree of Celsius and humidity as a percentage. The measurement was then initiated and executed as per the configuration. Upon completion, the results were displayed on the user interface and saved to the cloud database.

After completing the measurement, we initiated the uncertainty calculations using a designated button. The MCS calculations were carried out in the cloud environment and took approximately 20 minutes to process data from the 24 chosen test points. Upon completion, we received a link to an Excel file containing the results. Subsequently, we proceeded with the generation of the DCC. An excerpt from the generated DCC is illustrated in Figure 5.1. We then used a camera to scan the QR code, which directed us to a web page displaying the original results of the MCS calculations.

5.2 Validation of the Uncertainty Calculations

While it is essential to perform accurate and successive measurements, it is equally important to ensure that the uncertainty of these measurements is calculated correctly. This calculation should adhere to the principles outlined in the GUM document. According to GUM, the evaluation of measurement uncertainties can be conducted using the LoP [23] and MCS method [24], both of which are fundamental approaches in achieving reliable uncertainty quantification [75], [76].

In order to ensure the reliability and accuracy of the uncertainty calculation methods integrated into our microservice-based cloud architecture, we cross-validated them with the LoP and MCS methods from our prior research [2]. These methods were originally validated through a comprehensive comparison of uncertainty calculations performed using the ARP software and the Oracle Crystal Ball (OCB) [77] simulation application in a local environment. The study tested the methods across various RF power levels and frequencies, demonstrating their effectiveness in practical applications. This section outlines the validation process for both methods after migrating them to the cloud environment as microservices. Despite the initial validation being conducted locally, the methods are expected to function equivalently in the cloud environment, as they are dockerized versions of the same codes.

The technical specifications for the power meter, power sensor, RF signal source, and attenuator used in this measurement setup are provided in Table 5.1. Measurements were conducted at frequencies of 50 MHz, 1000 MHz, and 5000 MHz, as well as at 10000 MHz, 15000 MHz, and 18000 MHz, and at power levels of 0 dBm and 5 dBm. The ARP software was used to take 10 measurements from each test point, and the operator manually recorded the data for software validation.

Table 5.1: The used devices and their supported range. (Adapted from [2])

Device	**Model**	**Measurement Range**
RF Signal Generator	E8257D	250 kHz – 40 GHz
Attenuator (10 dB)	8491B	10 MHz – 18 GHz
Power Sensor	E4413A	50 MHz – 26.5 GHz
Power Meter	N1914A	-70 dBm to +44 dBm

In order to calculate the combined uncertainties, the equations (4.1), (4.2) were used for the LoP and MCS methods, respectively. During the uncertainty calculations process, the impedance mismatch (M) is assumed to be 1, since the magnitude of Γ_{DUT} is known.

Four different settings were created for the evaluation process:

- **Setting-1**: Uncertainty was calculated according to the GUM LoP method using manual measurement data and the MS Excel application.

- **Setting-2**: ARP software performed the uncertainty calculations using the LoP method (ARP-LoP).
- **Setting-3**: ARP software performed the uncertainty calculations using the MCS method (ARP-MCS).
- **Setting-4**: The Oracle Crystal Ball application was used to calculate uncertainty using the MCS method (OCB-MCS).

Uncertainty components and their statistical distributions are presented as:

- u_{PREAD}: Uncertainty of repeated power measurements, assumed to follow a Gaussian distribution.
- u_{PMacc}: Uncertainty of the power meter's accuracy, assumed to follow a rectangular distribution.
- u_{PMres}: Uncertainty of the power meter's resolution, assumed to follow a rectangular distribution.
- $u_{\text{CF\_STD}}$: Uncertainty of the calibration factor (CF) of the standard power sensor (STD PS), assumed to follow a Gaussian distribution.
- u_{S21A}: Uncertainty of the forward transmission coefficient of the attenuator, assumed to follow a Gaussian distribution.
- $u_{|\Gamma_A|}$: Uncertainty of the magnitude of Γ_A, assumed to follow a Gaussian distribution.
- $u_{|\Gamma_{\text{DUT}}|}$: Uncertainty of the magnitude of Γ_{DUT}, assumed to follow a Gaussian distribution.
- u_{Θ_A}: Uncertainty of the phase of Γ_A, assumed to follow a Gaussian distribution.
- $u_{\Theta_{\text{DUT}}}$: Uncertainty of the phase of Γ_{DUT}, assumed to follow a Gaussian distribution.
- u_M: Uncertainty due to the impedance mismatch of the connector where the magnitude of Γ_{DUT} is known, assumed to follow a U-shaped distribution.

To validate the LoP uncertainty calculation of ARP, the manually calculated LoP uncertainty using MS Excel was compared with the ARP LoP uncertainty calculation in Setting-1. The results showed a good agreement with differences at the level of 10^{-4}. This small difference is within an acceptable range, allowing this setting to be used as a reference for further comparisons.

In the first step of the study, the uncertainties calculated in Setting-2 and Setting-3 were compared. The Setting-1 was used as the reference value for this comparison.

In the second step, the uncertainties were calculated on the same measurement data, according to both Setting-3 and Setting-4. The calculated uncertainty values are presented in Table 5.2. Note that the coverage factor is taken as two during the calculation of combined uncertainties (u(k=2), 95% reliability). The differences between the calculated uncertainties and the reference uncertainty are provided in Table 5.3. In this section, only the differences between the calculated uncertainties were used as the evaluation method for comparison results.

Table 5.2: Comparison of calculated power and uncertainties at different test points. (Adapted from [2])

Frequency (MHz)	**Power** (dBm)	**Setting-1** (mW)	**Setting-2** (mW)	**Setting-3** (mW)	**Setting-4** (mW)
50	0	0.98580	0.01206	0.01209	0.01209
1000	0	1.00024	0.01412	0.01411	0.01413
5000	0	0.92623	0.03606	0.03619	0.03610
10000	0	0.90027	0.02156	0.02159	0.02159
15000	0	0.87236	0.06486	0.06500	0.06488
18000	0	0.82781	0.09185	0.09188	0.09193
50	5	3.15143	0.03851	0.03845	0.03856
1000	5	3.17756	0.04481	0.04473	0.04470
5000	5	2.94086	0.11449	0.11445	0.11458
10000	5	2.85686	0.06840	0.06853	0.06844
15000	5	2.78701	0.20720	0.20712	0.20749
18000	5	2.65898	0.29503	0.29472	0.29510

Table 5.3: Calculated uncertainty differences relative to the reference value (Setting-1: LoP based on GUM) (Adapted from [2])

Frequency (MHz)	**Power (dBm)**	**Setting-3 (ARP MCS method)**	**Setting-4 (OCB MCS method)**
50	0	0.00003	0.00003
1000	0	-0.00001	0.00001
5000	0	0.00013	0.00004
10000	0	0.00003	0.00003
15000	0	0.00014	0.00002
18000	0	0.00003	0.00008
50	5	-0.00006	0.00005
1000	5	-0.00008	-0.00011
5000	5	-0.00004	0.00009
10000	5	0.00013	0.00004
15000	5	-0.00008	0.00029
18000	5	0.00003	0.00003

The uncertainties calculated using the MCS method within the ARP software's uncertainty calculation module were plotted for each frequency. Selected plots are shown in Figure 5.2. The plots indicate that the uncertainties obtained from the MCS method do not follow a homogeneous normal distribution. This non-homogeneous distribution is evident from the triangles drawn in the figures; although the triangles are symmetrical, their peaks do not align with the peaks of the histograms. To achieve a homogeneous normal distribution, more than 10^5 power measurements should be conducted, rather than relying on randomized power values generated from just 10 measurements.

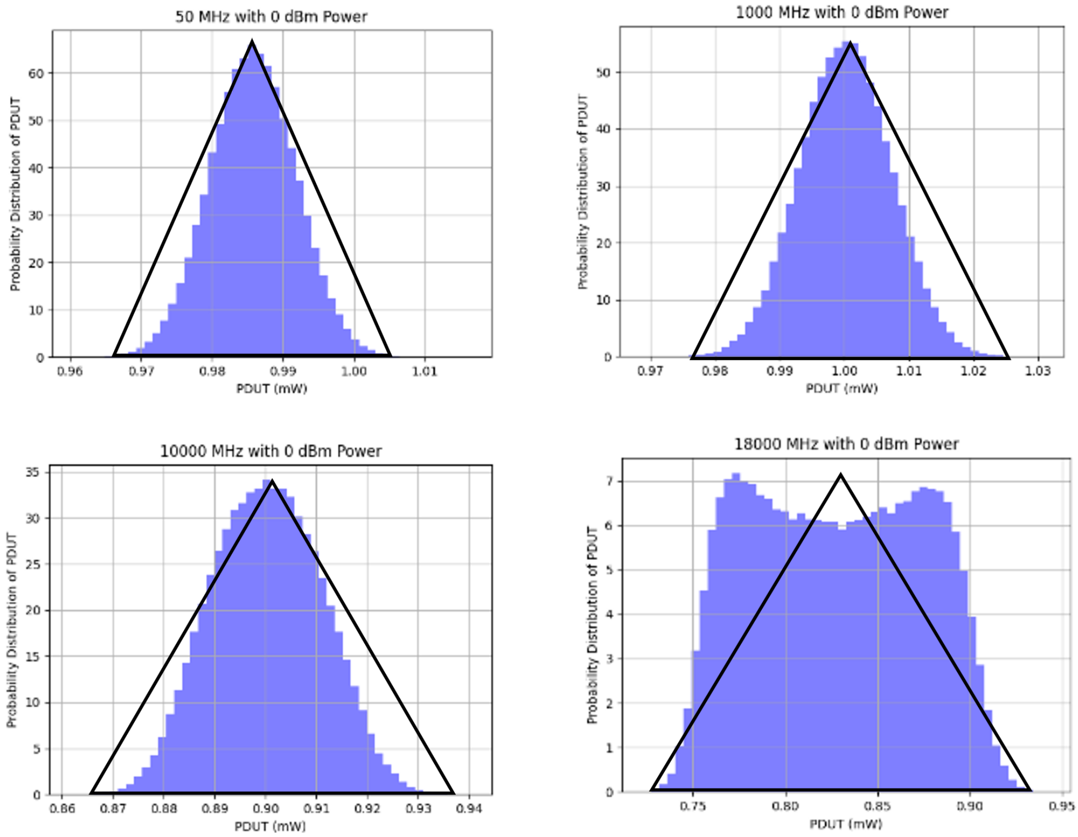

Figure 5.2: The calculate MCS Uncertainties at 0 dBm and 50 MHz, 1000 MHz, 10000 MHz, 18000 MHz test points (Adapted from [2])

The LoP method [23] calculates the combined uncertainty by assuming that all input parameters have normal distributions. This method is implemented in the ARP software, which was validated by comparing manually calculated uncertainties using MS Excel with those generated by the software. The results showed a high level of agreement, with differences at the 10^{-4} level, confirming the method's accuracy.

The MCS method provides an alternative approach by simulating the real measurement process multiple times, generating input parameters with their actual statistical distributions without assuming normal distribution. In our previous study, this method was validated using both the ARP software and the Oracle Crystal Ball application. The results showed that the MCS method produced a non-symmetrical normal distribution of uncertainties, reflecting a more realistic representation of the measurement process.

To validate the cloud-based implementation, we conducted uncertainty calculations using the same frequency, power levels, and settings as in the case study presented in Section 5.1. The results were compared with the manually calculated uncertainties from the evaluation study conducted with the utilization of ARP software [2]. This comparison revealed that the cloud-based system consistently yielded results, with the LoP method maintaining its high accuracy and the MCS method accurately representing the non-symmetrical distribution of uncertainties.

The uncertainties determined by both methods were well within acceptable limits, highlighting the effectiveness of the cloud-based system in executing precise and dependable uncertainty calculations. The incorporation of these methods into the microservice architecture guarantees that the system can proficiently handle intricate calibration processes, offering resilient and adaptable solutions for the calibration industry.

5.3 Critical Evaluation of Algorithms and Methodologies

In this research, we utilized two different uncertainty calculation methods: the LoP and MCS, as outlined by the BIPM [23]. Although LoP is the most commonly used method, it may not be suitable for non-linear models or when input variables deviate significantly from a normal distribution. In such cases, MCS is preferred because it can accurately account for complex interactions and does not require the assumption of linearity or normally distributed inputs, which are often necessary for LoP. Although MCS is more resource-intensive, requiring at least 10^5 or more repetitive measurements to simulate real-world conditions, it is particularly valuable in laboratories with the capability to perform measurements at very high frequencies. Therefore, both methods are employed to leverage their respective strengths in uncertainty analysis.

In the realm of uncertainty calculations, it's important to evaluate the adequacy of various methodologies, from traditional statistical methods like the MCS to newer approaches such as predictive machine learning models. While predictive machine learning models have shown promise in various fields, their application in RF power measurements presents significant challenges. RF power measurements can be performed with devices from multiple manufacturers, each with unique characteristics that may influence the measurement results. Additionally, the performance of these devices can degrade over time due to wear, further complicating the modelling process. Environmental factors such as temperature and humidity can also have substantial effects on measurement accuracy. Creating a reliable predictive model for uncertainty calculation requires accounting for these variations through the incorporation of a comprehensive and diverse dataset that encompasses the full range of possible conditions and device behaviours. However, the availability of such data is often limited, as it is considered classified by many institutions, which are reluctant to share it. Obtaining the necessary dataset is highly challenging, if not impractical, given the number of devices and environmental variables involved. In contrast, the MCS method, which does not rely on historical data or predictive algorithms, provides a more robust approach in this context. MCS can accurately model uncertainty by simulating real-world conditions without the need for extensive training data, making it a more reliable choice for uncertainty calculation in RF power measurements. Nevertheless, machine learning models might offer potential in controlled environments or for specific applications where sufficient data and stable conditions are available.

5.4 Results, Implications, Limits, and Future Perspectives

Digitalization efforts in metrology, although still fairly preliminary, are emerging as a critically important endeavour. Metrology as a field is defined by its utmost heterogeneity, consisting of a wide array of complex devices produced by various manufacturers, often lacking interoperability due to proprietary standards, technical challenges and business-related decisions. Additionally, the nature of metrology involves handling sensitive data that demands high computational power and precision during processing, all the while maintaining stringent security measures. Given these unique requirements, the adoption of cloud computing along with microservices presents a particularly apt solution with significant advantages in terms of system flexibility, extendibility, and reduced dependencies through the use of dockerized microservice environments. These characteristics ensure that our system can efficiently integrate diverse components and adapt seamlessly to ongoing technological advancements.

Our cloud-based application incorporates advanced variability handling mechanisms to ensure flexibility and adaptability from a software engineering perspective. Although presently tailored to RF power measurements, the system is inherently extensible and can accommodate a broad spectrum of measurement types, including temperature, pressure, and more. Through systematic variability handling, the application can efficiently manage multiple measurement processes, streamlining configuration and enhancing scalability. Additionally, it offers precise access control, granting users permissions based on their qualifications and eligibility, thus bolstering security.

Our solution emphasizes the benefits of microservice architecture in digital metrology and calibration applications, including scalability and maintainability [78], [79]. We aim to drive digital transformation in the metrology and calibration industry by using advanced microservice-based tools and technologies, as traditional monolithic architectures are costly and becoming less effective due to inherent diversity and scalability needs.

Nevertheless, this research may face some threats to its validity that need to be considered. Primarily, the initial cost of cloud technologies may pose a barrier to widespread adoption. However, the long-term benefits such as scalability and operational efficiency are expected to outweigh these concerns. Furthermore, some stakeholders accustomed to handling sensitive data, e.g. in the defence industry, may have reservations about storing their data on the cloud. Even if the cloud-based system has robust security measures, it is still a third-party entity.

Fortunately, our approach is not dependent on any specific Platform as a Service (PaaS) providers. The use of dockerized microservices allows for deployment across various PaaS providers and can be adapted to Infrastructure as Code (IAC), where servers, storage, and networking are managed by developers.This level of flexibility ensures that our system can be tailored to meet the specific security and operational needs of different stakeholders, thereby alleviating some of the concerns associated with cloud adoption.

Digital Calibration Certificate

Administrative Data

Software Name: CSE-DCC Tool

Release: v1.0.0

Coredata

Country Code (ISO3166_1): SE

Used Language(s) (ISO639_1): en

Mandatory Language(s) (ISO3166_1): en

Unique Identifier: 56752da6-5f60-419d-92f2-25a924f2bddb

Date of Reception: 2024-04-24

Date of Calibration: 2024-04-24

Place of Calibration: CSE Calibration Lab

Date: 2024-04-24

Scan the QR code to verify this certificate:

Approved by [Approver's name]

Devices under Test

[redacted] [redacted] [redacted]

Calibration Laboratory

Personnel: [redacted]

e-mail: [redacted]

Street Number: [redacted]

Calibration Laboratory Code: [redacted]

Person(s) authorizing the report

Customer: Customer's name

e-mail: firsname.lastname@customer-company.com

Street Number: 8

Measurement Results

Freq (MHz)	Power (dBm)	PDUT_Corr_GUM (mW)	UPDUT_k2_GUM (mW)	Low Endpoint for 95% Confidence Interval	High Endpoint for 95% Confidence Interval	Mean of PDUT_MCS (mW)	Standard Deviation of PDUT_MCS (mW)	2 * Standard Deviation of PDUT_MCS (mW)	Total Simulation Time Elapsed = 157.763578176494 sec
50	0	9.86	0.10	-98.34	-96.34	-97.31	0.50	1.01	NaN
1000	0	10.51	0.10	-113.43	-111.43	-112.44	0.51	1.03	NaN
5000	0	10.35	0.17	-113.15	-110.06	-111.58	0.78	1.57	NaN
10000	0	10.55	0.27	-120.61	-117.16	-118.83	0.87	1.75	NaN
15000	0	10.80	0.39	-130.89	-123.03	-126.73	2.01	4.02	NaN
18000	0	10.68	0.32	-127.58	-121.93	-124.66	1.44	2.89	NaN
50	5	31.41	0.33	-996.03	-976.39	-986.48	5.10	10.21	NaN
1000	5	33.48	0.31	-1152.42	-1132.04	-1142.15	5.25	10.49	NaN
5000	5	32.92	0.54	-1145.88	-1115.56	-1130.49	7.90	15.80	NaN

Figure 5.1: An excerpt of the generated DCC [3]

CHAPTER 6

CONCLUSIONS

This thesis has explored the potential of leveraging microservice-based cloud architectures to address the complexities inherent in the metrology and calibration industry. Through the development and deployment of an enhanced IoMT (Internet of Measurement Things) architecture, this work has demonstrated how microservices can be effectively used to manage the digital transformation challenges in this field. The integration of microservices, along with advanced uncertainty quantification techniques, not only streamlines the calibration process but also enhances the scalability, adaptability, and reliability of the system.

The deployment and validation of the ARP application within this framework have demonstrated the practicality and advantages of our approach. The integration of advanced uncertainty calculation methodologies, such as the LoP and MCS, within a scalable cloud infrastructure underscores the robustness and adaptability of the proposed system. The empirical validation through the case study on RF power measurement highlights the system's capability to handle complex calibration tasks.

The microservice-based design ensures that each component can be independently developed, deployed, and scaled. This modularity is crucial for integrating diverse measurement types and adapting to technological advancements, thereby future-proofing the system against evolving industry requirements.

The digital transformation in metrology goes beyond simply adopting new technologies; it involves fundamentally reshaping how calibration and measurement processes are managed. The proposed architecture aligns with these trends by offering a flexible, cloud-based platform capable of handling the growing complexity of calibration tasks. Through the use of containerized environments and orchestration via platforms like Google Kubernetes Engine (GKE), the system can efficiently scale to accommodate the increasing volumes of calibration data and processes. This scalability is particularly crucial given the growing complexity and volume of calibration tasks associated with digital transformation trends.

The use of robust authentication mechanisms, including UUID-based QR codes for DCCs, ensures the integrity and authenticity of calibration certificates. This is a critical advancement in ensuring trust and reliability in digital metrology practices.

The cloud-based infrastructure enables efficient data processing and storage, facilitating real-time updates and integration with various measurement devices. This not only enhances the accuracy and reliability of calibration results but also reduces the likelihood of errors due to manual intervention.

The findings of this study hold promise, but it's important to acknowledge certain limitations. One notable constraint when using a cloud-based microservice architecture in the calibration industry is the potential for privacy and data security issues. Calibration data, often sensitive and proprietary, may contain crucial information that organizations are hesitant to store or share on cloud servers due to concerns about unauthorized access, data breaches, or regulatory non-compliance. Addressing these concerns will necessitate the development of more robust encryption methods, access control mechanisms, and possibly exploring hybrid cloud solutions that allow sensitive data to remain on-premises while still benefiting from cloud-based processing capabilities.

Furthermore, the implementation and maintenance of a microservice-based architecture requires expertise in cloud technologies, posing a significant barrier for some organizations, particularly those with limited technical resources. The complexity of designing, deploying, and managing such an architecture demands a deep understanding of cloud infrastructure, containerization, orchestration tools, and microservice principles. Additionally, the financial costs associated with cloud-based architectures can be significant, encompassing not only the direct costs of cloud services but also ongoing expenses related to scaling, maintenance, and security.

For organizations with constrained budgets, the high cost of cloud services, coupled with the need for specialized skills, may deter the adoption of this architecture. To address these challenges, efforts should be directed towards developing more user-friendly interfaces and deployment tools that simplify the management of microservices and cloud resources. Moreover, offering flexible pricing models or hybrid solutions that allow critical processes to be managed on-premises while leveraging cloud resources for specific tasks could help lower the entry barrier and make this architecture more accessible and cost-effective for a broader range of organizations.

While this work primarily focuses on RF power measurement, there is potential for future research to explore its application in a broader range of calibration processes, including temperature and pressure measurements. Expanding the system's capabilities to accommodate different types of measurements would further validate its flexibility and scalability. Ongoing project preparations aim to develop a digital infrastructure for temperature measurement and dissemination, including a workflow management task that seeks to integrate with various metrology institutes across Europe. Additionally, the generation and management of DCCs could play a significant role in ensuring the accuracy and security of certificates while contributing to standardization across the industry.

The integration of machine learning algorithms within the microservice architecture could provide predictive analytics capabilities, enabling the system to anticipate calibration needs and optimize measurement processes in real-time. This would further enhance the efficiency and accuracy of the calibration process, particularly in complex and dynamic environments.

As the industry moves towards greater digitalization, there is a growing need for standardized approaches to calibration data management. Future work should focus on contributing to the development of industry-wide standards for DCCs and uncertainty quantification, ensuring that the proposed architecture is interoperable with existing and emerging systems.

In conclusion, this thesis has laid a solid foundation for the future of digital metrology through the application of microservices supported by variability management for dynamically configured solutions. By addressing the current challenges in calibration processes and aligning with the digital transformation trends in the industry, this work not only advances the state of the art in metrology but also opens up new pathways for innovation and improvement. The proposed architecture represents a significant step towards more automated, reliable, and scalable calibration systems, setting the stage for ongoing advancements in the field.

REFERENCES

[1] A. Cetinkaya, A. K. Dogan, E. Danaci, and H. Oguztuzun, "Autorfpower: Automatic rf power measurement software for metrological applications," in *2021 2nd International Informatics and Software Engineering Conference (IISEC)*, pp. 1–4, IEEE, 2021.

[2] E. Danacı, A. Kartal Doğan, E. C. Çiçek, A. Çetinkaya, M. Ç. Kaya, M. S. H. Oğuztüzün, and G. Tünay, "Uncertainty evaluation using law of propagation and monte carlo simulation methods with the autorfpower measurement software," *Konya Journal of Engineering Sciences*, vol. 12, no. 3, p. 596–607, 2024.

[3] A. Cetinkaya, M. C. Kaya, E. Danaci, and H. Oguztuzun, "Uncertainty calculation as a service: Integrating cloud-based microservices for enhanced calibration and dcc generation," *Sensors*, vol. 24, no. 17, 2024.

[4] Google, "Develop and deploy containerized apps using a ci/cd pipeline," 2024. Accessed on April 25, 2024.

[5] S. Newman, *Building microservices.* " O'Reilly Media, Inc.", 2021.

[6] A. Bucchiarone, N. Dragoni, S. Dustdar, P. Lago, M. Mazzara, V. Rivera, and A. Sadovykh, "Microservices," *Science and Engineering. Springer*, 2020.

[7] H. Aljawawdeh, S. Abuezhayeh, E. Qaddoumi, and L. Maghrabi, *Smooth Change: Moving from Monolithic to Microservices with Flexible Methods*, pp. 559–572. Cham: Springer Nature Switzerland, 2023.

[8] L. Chen and M. A. Babar, "Variability management in software product lines: An investigation of contemporary industrial challenges," in *Software Product Lines: Going Beyond* (J. Bosch and J. Lee, eds.), (Berlin, Heidelberg), pp. 166–180, Springer Berlin Heidelberg, 2010.

[9] W. K. Assunção, J. Krüger, and W. D. Mendonça, "Variability management meets microservices: six challenges of re-engineering microservice-based web-

shops," in *Proceedings of the 24th ACM Conference on Systems and Software Product Line: Volume A-Volume A*, pp. 1–6, 2020.

[10] M. Kuster, "A measurement information infrastructure's benefits for industrial metrology and iot," in *2020 IEEE international workshop on metrology for industry 4.0 & IoT*, pp. 479–484, IEEE, 2020.

[11] A. Varshney, N. Garg, K. Nagla, T. Nair, S. Jaiswal, S. Yadav, and D. Aswal, "Challenges in sensors technology for industry 4.0 for futuristic metrological applications," *Mapan*, vol. 36, no. 2, pp. 215–226, 2021.

[12] M. S. Nikoo, M. C. Kaya, M. L. Schwartz, and H. Oguztuzun, "An mii-aware soa editor for the industrial internet of things," in *2019 II Workshop on Metrology for Industry 4.0 and IoT (MetroInd4. 0&IoT)*, pp. 213–218, IEEE, 2019.

[13] S. Hackel, S. Schönhals, L. Doering, T. Engel, and R. Baumfalk, "The digital calibration certificate (dcc) for an end-to-end digital quality infrastructure for industry 4.0," *Sci*, vol. 5, no. 1, 2023.

[14] M. S. Gadelrab and R. A. Abouhogail, "Towards a new generation of digital calibration certificate: Analysis and survey," *Measurement*, vol. 181, p. 109611, 2021.

[15] T. Lafarge and A. Possolo, "The NIST uncertainty machine," *NCSLI Measure*, vol. 10, no. 3, pp. 20–27, 2015.

[16] S. Hackel, F. Härtig, J. Hornig, and T. Wiedenhöfer, "The digital calibration certificate," *PTB-Mitteilungen*, vol. 127, no. 4, pp. 75–81, 2017.

[17] H. Lasi, P. Fettke, H.-G. Kemper, T. Feld, and M. Hoffmann, "Industry 4.0," *Business & Information Systems Engineering*, vol. 6, pp. 239–242, Aug 2014.

[18] M. Saeedi Nikoo, M. C. Kaya, M. L. Schwartz, and H. Oguztuzun, *Internet of Measurement Things: Toward an Architectural Framework for the Calibration Industry*, pp. 81–102. Cham: Springer International Publishing, 2019.

[19] M. J. Kuster, "Toward a Metrology Information Infrastructure," in *Metrologist NCSLI Worldwide News*, 2013.

[20] "Metrology.NET Calibration Automation Platform." https://www.metrology.net/. Accessed: 2019-09-23.

[21] Kubernetes, "Kubernetes documentation," 2024. Accessed on Aug 5, 2024.

[22] G. Cloud, "Google kubernetes engine (GKE)," 2024. Accessed on April 25, 2024.

[23] J. Jcgm *et al.*, "Evaluation of measurement data—guide to the expression of uncertainty in measurement," *Int. Organ. Stand. Geneva ISBN*, vol. 50, p. 134, 2008.

[24] I. Bipm, I. IFCC, and I. Iupac, "Evaluation of measurement data-supplement 1 to the'guide to the expression of uncertainty in measurement," *Propagation of distributions using a Monte Carlo method*, 2008.

[25] M. Saeedi Nikoo, M. C. Kaya, M. L. Schwartz, and H. Oguztuzun, *Internet of Measurement Things: Toward an Architectural Framework for the Calibration Industry*, pp. 81–102. Cham: Springer International Publishing, 2019.

[26] M. C. Kaya, M. Saeedi Nikoo, M. L. Schwartz, and H. Oguztuzun, "Internet of measurement things architecture: Proof of concept with scope of accreditation," *Sensors*, vol. 20, no. 2, 2020.

[27] A. Cetinkaya, M. C. Kaya, E. Danaci, and H. Oguztuzun, "Uncertainty calculation-as-a-service: An IIoT application for automated RF power sensor calibration," in *Proceedings of the IMEKO TC6 International Conference on Metrology and Digital Transformation*, (Berlin, Germany), IMEKO, 2022.

[28] A. Cetinkaya, M. C. Kaya, T. B. Bzuneh, and H. Oguztuzun, "Uncertainty calculation-as-a-service: Microservice-based metrology applications," in *Software Architecture*, (Cham), pp. 183–190, Springer Nature Switzerland, 2024.

[29] I. S. O. Vim, "International vocabulary of Metrology –Basic and general concepts and associated terms (VIM)," *International Organization*, 2012.

[30] P. R. G. Couto, J. C. Damasceno, S. d. Oliveira, and W. Chan, "Monte carlo simulations applied to uncertainty in measurement," *Theory and applications of Monte Carlo simulations*, vol. 2, pp. 27–51, 2013.

[31] S. Dustdar and M. P. Papazoglou, "Services and service composition–an introduction (services und service komposition–eine einführung)," *IT-Information Technology*, vol. 50, no. 2, pp. 86–92, 2008.

[32] Z. Xiao, I. Wijegunaratne, and X. Qiang, "Reflections on soa and microservices," in *2016 4th International Conference on Enterprise Systems (ES)*, pp. 60–67, 2016.

[33] S. Suloglu, M. C. Kaya, A. Cetinkaya, A. Karamanlioglu, and A. H. Dogru, *Cloud-Enabled Domain-Based Software Development*, pp. 109–130. Cham: Springer International Publishing, 2020.

[34] R. Capilla, J. Bosch, K.-C. Kang, *et al.*, "Systems and software variability management," *Concepts Tools and Experiences*, vol. 10, p. 2517766, 2013.

[35] K. Pohl and A. Metzger, "Variability management in software product line engineering," in *Proceedings of the 28th international conference on Software engineering*, pp. 1049–1050, 2006.

[36] M. C. Kaya, A. Cetinkaya, and A. H. Dogru, "Off-the-shelf connectors for interdisciplinary components," *Journal of Integrated Design and Process Science*, vol. 22, no. 3, pp. 35–53, 2018.

[37] A. Dogru, "Toward a component-oriented methodology to build-by-integration," *Development of component-based information systems*, pp. 49–69, 2005.

[38] M. Ç. Kaya, "Modeling variability in component oriented software engineering," Master's thesis, MSc Thesis, Middle East Techical University, 2015.

[39] A. Cetinkaya, M. C. Kaya, and A. H. Dogru, "Component integration through connector supported process models," in *Proceedings of the SDPS 22nd international conference on emerging trends and technologies in convergence solutions*, (Birmingham, AL), pp. 31–37, 2017.

[40] M. C. Kaya, A. Cetinkaya, and A. H. Dogru, "Composition capability of component-oriented development," in *5th International Symposium on Innovative Technologies in Engineering and Science (ISITES)*, (Baku, Azerbaijan), pp. 1299, 1307, 2017.

[41] A. Çetinkaya, "Variable connectors in component oriented development," Master's thesis, MSc Thesis, Middle East Technical University, 2017.

[42] A. Cetinkaya, M. C. Kaya, and A. H. Dogru, "Enhancing xcoseml with connector variability for component-oriented development," in *Proceedings of the SDPS 21st international conference on emerging trends and technologies in designing healthcare systems*, (Orlando, FL), pp. 120–125, 2016.

[43] A. Çetinkaya, A. Karamanlıoğlu, M. Ç. Kaya, and A. H. Doğru, "Eşgüdümlü bileşenler ile birleştirme yaklaşımı," *presentation in Ulusal Yazılım Mühendisliği Sempozyumu, Antlaya, Turkey*, 2017.

[44] M. C. Kaya, M. Saeedi Nikoo, S. Suloglu, B. Tekinerdogan, and A. H. Dogru, *Managing Heterogeneous Communication Challenges in the Internet of Things Using Connector Variability*, pp. 127–149. Cham: Springer International Publishing, 2017.

[45] C. Zet, G. Dumitriu, C. Fosalau, and G. C. Sarbu, "Automated calibration and dcc generation system with storage in private permissioned blockchain network," *Acta IMEKO*, vol. 12, no. 1, pp. 1–7, 2023.

[46] A. Oppermann, S. Eickelberg, and M. Meiborg, "Digital transformation: Towards process automation in a cloud native architecture," *Acta IMEKO*, vol. 12, no. 1, pp. 1–6, 2023.

[47] R. P. Pontarolli, J. A. Bigheti, L. B. R. de Sá, and E. P. Godoy, "Microservice-oriented architecture for industry 4.0," *Eng*, vol. 4, no. 2, pp. 1179–1197, 2023.

[48] MoleculerJS, "What is moleculer?," 2024. Accessed on July 29, 2024.

[49] J. Nummiluikki, S. Saxholm, A. Kärkkäinen, and S. Koskinen, "Digital calibration certificate in an industrial application," *Acta IMEKO*, vol. 12, no. 1, pp. 1–6, 2023.

[50] M. Zeier, J. Hoffmann, and M. Wollensack, "Metas. unclib: A measurement uncertainty calculator for advanced problems." Software tool. Available as a measurement uncertainty calculator.

[51] S. Hackel, S. Schönhals, L. Doering, T. Engel, and R. Baumfalk, “The digital calibration certificate (dcc) for an end-to-end digital quality infrastructure for industry 4.0,” *Sci*, vol. 5, no. 1, p. 11, 2023.

[52] B. Kuruoglu Dolu, A. Cetinkaya, M. C. Kaya, S. Nazlioglu, and A. H. Dogru, “Msdeveloper: A variability-guided methodology for microservice-based development,” *Applied Sciences*, vol. 12, no. 22, 2022.

[53] A. Fantom, *Radio frequency & microwave power measurement*. No. 7, IET, 1990.

[54] D. M. Pozar, *Microwave engineering: theory and techniques*. John wiley & sons, 2021.

[55] A. Yüğrük, E. Danaci, A. K. Doğan, and A. O. Salman, “The effects of sequential and multiple measurement on rf power,” in *2021 29th Signal Processing and Communications Applications Conference (SIU)*, pp. 1–4, IEEE, 2021.

[56] Microsoft, “C# docs - get started, tutorials, reference,” 2024. Accessed on Aug 15, 2024.

[57] Microsoft, “Visual studio 2019 version 16.11 release notes,” 2024. Accessed on Aug 15, 2024.

[58] F. A. Krueger, “sqlite-net,” 2021. Accessed on Aug 15, 2024.

[59] SourceGear, “Nuget gallery | sqlitepclraw.bundle_e_sqlcipher 2.1.9,” 2021. Accessed on Aug 15, 2024.

[60] IEEE, “Ieee standard for programmable instrumentation,” 2004. Accessed: 2024-08-20.

[61] I. Foundation, “The scpi standard,” 2004. Accessed on Aug 15, 2024.

[62] I. Foundation, “Specification downloads ivi foundation,” 2004. Accessed on Aug 15, 2024.

[63] N. Instruments, “Ni-visa .net library - ni,” 2004. Accessed on Aug 15, 2024.

[64] Bird, “Vpm3 virtual rf power meter software,” 2024. Accessed on Aug 24, 2024.

[65] C. L. Solutions, "Metrology software, redefined," 2024. Accessed on Aug 24, 2024.

[66] K. Technologies, "Benchvue power meter/sensor control and analysis," 2024. Accessed on Aug 24, 2024.

[67] Rohde&Schwarz, "Software for r&s nrp-zx1," 2024. Accessed on Aug 24, 2024.

[68] FlukeCorp, "Met/cal calibration management software," 2022. Accessed on Aug 24, 2024.

[69] yaml.org, "The official yaml web site," 2024. Accessed on Aug 4, 2024.

[70] RedHat, "What is a rest api?," 2020. Accessed on April 25, 2024.

[71] W. M. C. J. T. Kithulwatta, W. U. Wickramaarachchi, K. P. N. Jayasena, B. T. G. S. Kumara, and R. M. K. T. Rathnayaka, "Adoption of docker containers as an infrastructure for deploying software applications: A review," in *Advances on Smart and Soft Computing* (F. Saeed, T. Al-Hadhrami, E. Mohammed, and M. Al-Sarem, eds.), (Singapore), pp. 247–259, Springer Singapore, 2022.

[72] Flask, "Flask framework," 2024. Accessed on April 25, 2024.

[73] G. M. Mahmoud and R. S. Hegazy, "Comparison of gum and monte carlo methods for the uncertainty estimation in hardness measurements," *International Journal of Metrology and Quality Engineering*, vol. 8, p. 14, 2017.

[74] T. O. M. Group, "About the unified modeling language specification version 2.5.1," 2017. Accessed on Aug 5, 2024.

[75] P. R. G. Couto, J. C. Damasceno, S. d. Oliveira, and W. Chan, "Monte carlo simulations applied to uncertainty in measurement," *Theory and applications of Monte Carlo simulations*, vol. 2, pp. 27–51, 2013.

[76] D. CF, "Uncertainty, calibration and probability," *Bristol, UK, Adam Hilger*, 1991.

[77] Oracle, "Crystal ball: Spreadsheet-based application for predictive modeling, forecasting, simulation, and optimization," 2024. Accessed on Aug 25, 2024.

[78] S. Baškarada, V. Nguyen, and A. Koronios, "Architecting microservices: Practical opportunities and challenges," *Journal of Computer Information Systems*, 2020.

[79] L. De Lauretis, "From monolithic architecture to microservices architecture," in *2019 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*, pp. 93–96, 2019.

CURRICULUM VITAE

PERSONAL INFORMATION

Surname, Name: Çetinkaya, Anıl
Nationality: Turkish (TC)

EDUCATION

Degree	Institution	Year of Graduation
M.S.	Middle East Technical University	2017
B.S.	Eastern Mediterranean University	2011

PROFESSIONAL EXPERIENCE

Year	Place	Enrollment
2013-2024	Middle East Technical University	Research and Teaching Assistant

RESEARCH PROJECTS

- TÜBİTAK: TEYDEB 1505 Project,
  No: 5200040
  Title: A New Method and Software Development for Automatic RF Power Measurement and RF Power Meter Calibration
  In collaboration with: TÜBİTAK UME & Spark Calibration
  Role: Researcher
  Advisor: Prof. Dr. Halit Oğuztüzün

- European Partnership on Metrology (EURAMET) Project,
  No: 22RPT04
  Title: Development of RF and microwave metrology capability II (RFMicrowave2)
  Participants: CMI, TÜBİTAK, METU, SIQ, RISE, NSAI, IMBiH, Trescal
  Role: Researcher
  Advisor: Prof. Dr. Halit Oğuztüzün

PUBLICATIONS

Journal Articles

Science Citation Index Expanded

[1] Cetinkaya, A., Kaya, M. C., Danaci, E. and Oguztuzun, H., 2024. Uncertainty Calculation as a Service: Integrating Cloud-Based Microservices for Enhanced Calibration and DCC Generation. *Sensors*, 24(17), 5651. https://doi.org/10.3390/s24175651

[2] Danaci, E., Bayrak, Y., Cetinkaya, A., Arslan, M., Sakarya, H., Dogan, A. K., and Tunay, G., 2023. Analysis of Vector-Network-Analyzer-Based Power Sensor Calibration Method Application. Instruments, 7(3), 21., DOI: 10.3390/instruments7030021

[3] Dolu, B. K., Cetinkaya, A., Kaya, M. C., Nazlioglu, S., and Dogru, A. H., 2022. MSDeveloper: A Variability-Guided Methodology for Microservice-Based Development. Applied Sciences, 12(22), 11439, DOI: 10.3390/app122211439.

Other Indexes

[4] Kaya, M.C., Cetinkaya, A., and Dogru, A.H., 2018. ”Off-the-Shelf Connectors for Interdisciplinary Components,” *Journal of Integrated Design and Process Science*, DOI: 10.3233/JID190003.

National - Reviewed

[5] Danaci, E., Dogan, A. K., Cicek, E.C.,Cetinkaya, A., Kaya, M. C., Oguztuzun, H., and Tunay, G, 2024. Uncertainty Evaluation Using Law of Propagation and Monte Carlo Simulation Methods with the AutoRFPower Measurement Software. *Konya Engineering Sciences (KONJES) Journal*, vol. 12, no. 3, pp. 596–607, 2024, doi: 10.36306/konjes.1364464.

Book Chapters

[6] Suloglu, S., Kaya, M.C., A. Cetinkaya, Karamanlioglu, A., and Dogru, A.H.,2020. “Cloud enabled Domain Based Software Development,” *Software Engineering in the Era of Cloud Computing*, 109-130 Springer, Cham.

International Conference Publications

[7] Cetinkaya, A., Kaya, K. M., Bzuneh, T. B., and Oguztuzun, H., 2024. "Uncertainty Calculation-as-a-Service: Microservice-Based Metrology Applications." *18th European Conference on Software Architecture (ECSA)*, (pp. 183–190). Springer, Cham.

[8] Cetinkaya, A., Kaya, K. M., Danaci, E., and OĞUZTÜZÜN, M., 2022. Uncertainty Calculation-As-A-Service: An IIoT Application for Automated RF Power Sensor Calibration. In IMEKO TC6 International Conference on Metrology and Digital Transformation.

[9] Kızılbey, O., Arslan, M., Bayrak, Y., Cetinkaya, A., Yüğrük, A., and Danaci, E., 2022. A Novel Software for Automatic Calibration Factor Measurement of RF Power Sensors. In 2022 7th International Conference on Computer Science and Engineering (UBMK) (pp. 371-373). IEEE.

[10] Cetinkaya, A., Doğan, A. K., Danaci, E., and Oğuztüzün, H., 2021. AUTORF-POWER: Automatic RF Power Measurement Software for Metrological Applications. In 2021 2nd International Informatics and Software Engineering Conference (IISEC) (pp. 1-4). IEEE.

[11] Cetinkaya, A., Suloglu, S.,Kaya, M. C., Karamanlioglu, A., Tokdemir, G., and Dogru, A. H., 2019 “An Experimental Study on Decomposition: Process First

or Structure First?," *9th International Symposium on Business Modeling and Software Design* (pp. 279–289). Springer, Cham.

[12] Cetinkaya, A.,Kaya, M. C., Dogru, A. H., 2017. "Component Integration Through Connector Supported Process Models," *Emerging Trends and Technologies in Convergence Solutions, Society for Design and Process Science* (pp. 31–37). Birmingham, ALABAMA, USA

[13] Kaya, M. C., Cetinkaya, A., and Dogru, A. H., 2017. Composition capability of component-oriented development. In 5th International Symposium on Innovative Technologies in Engineering and Science (ISITES), Baku-Azerbaijan.

[14] Cetinkaya, A., Kaya, M. C., and Dogru, A. H., 2016. Enhancing XCOSEML with connector variability for component oriented development. In Proceedings of SDPS 21st international conference on emerging trends and technologies in designing healthcare systems, Orlando, FL, USA (pp. 120-125).

National Conference Publications

[15] Cetinkaya, A., Karamanlioglu, A., Kaya, M. C., and Dogru, A. H., 2017. Integration Approach through Coordinated Components, In *Proceedings of the 11th Turkish National Software Engineering Symposium* (pp. 619–631). Alanya, Turkey.

Master's Thesis

[16] Cetinkaya, A., 2017. Variable Connectors in Component Oriented Development, MSc Thesis, Middle East Technical University.