A Modular Network Digital Twin for Radio Coverage Prediction: From Theory to Practice

Ayat Zaki-Hindi, Jean-Sébastien Sottet, Sumit Kumar, Ion Turcanu, Sébastien Faye
Luxembourg Institute of Science and Technology (LIST), Luxembourg
{ayat.zaki-hindi, jean-sebastien.sottet, sumit.kumar, ion.turcanu, sebastien.faye }@list.lu

***Abstract*—Network Digital Twins (NDTs) offer a structured framework for modeling, predicting, and optimizing wireless networks. This paper presents a modular NDT implementation based on the GreyCat platform, integrating graph-based data models and external functional algorithms for indoor radio coverage prediction. For the first time, we implement an NDT system aligned with ITU-T Recommendation Y.3090, covering both basic and functional model instantiation from modular and interoperable abstract structures. We generated a practical dataset using a software-defined radio (SDR)-based OpenAirInterface5G setup, with a gNB and commercial UE deployed in a controlled environment. This real-world dataset was used to benchmark Gaussian Process Regression (GPR) and Convolutional Neural Network (CNN) models for predicting RSRP-based radio coverage. Our results show that CNN outperforms GPR in under-sampled conditions, and we demonstrate how the modular architecture supports flexible model integration and benchmarking. This work represents a significant step toward practical, data-driven NDT deployments for wireless systems.**

I. Introduction

The emergence of Network Digital Twins (NDTs) for next-generation networks promises to transform the way wireless infrastructures are designed, monitored, and optimized. According to the ITU-T Y.3090 recommendation [1], an NDT comprises two key modeling components: basic models, which represent the current state of the network including topology, configuration, and environment; and functional models, which capture expected behaviors and enable simulation, prediction, and control. This layered modeling approach ensures the NDT can both describe and act upon the network it mirrors.

In this work, we explore how an NDT can be used to infer radio coverage from sparse measurement data, enabling predictive capabilities that support advanced planning, troubleshooting, and optimization tasks. More specifically, we extend a previously defined data structure and set of basic models that enable the graph-based representation and storage of network measurements and configuration data in a unified format [2]. These models are implemented using the GreyCat platform and are further enhanced with functional models that allow predictive NDT capabilities.

The objective of this study is to demonstrate how a general-purpose NDT, built on standardized and harmonized data structures for next-generation networks, can be instantiated and adapted for specific applications and scenarios. We showcase the flexibility of the proposed NDT architecture in selecting basic models that reflect the characteristics of the real network, alongside functional models tailored to the targeted application. This modular and adaptable design enables the creation of multiple NDT instances, supports the testing of diverse solutions, and facilitates model calibration. Ultimately, the insights gained from these instances can be used to recommend or implement changes in the real network, completing what is referred to as the lifecycle of the NDT instance [3].

This approach is motivated by the lack of practical implementations that demonstrate how NDTs can be used in realistic settings. While previous work outlined the architectural principles for structured, AI-integrated NDTs [4], our contribution lies in demonstrating the feasibility of such models through a concrete deployment and measurement-based evaluation used for predictive radio analytics.

The contributions of this paper are twofold:

1) We design the first implementation of an NDT aligned with ITU-T Rec. Y.3090 [1], including both basic and functional model instantiations for radio coverage prediction.
2) We demonstrate a practical data-driven NDT system using real-world measurements collected via a software-defined radio (SDR)-based OpenAirInterface5G gNB and commercial UE, operationalized within a graph-based modeling platform (GreyCat).

The rest of the paper is structured as follows. Section II reviews related work. Section III presents our system design. Section IV describes the evaluation methodology and results. Section V concludes the paper.

II. Related work

This section reviews prior work relevant to our approach and identifies key gaps in the literature that we aim to address.

A. NDTs for next-generation networks

Standards bodies and industry consortia are rapidly embracing NDTs. In 3GPP, SA5 (Service and System Aspects Working Group 5) has launched Rel-19 studies on NDT management and orchestration in TS 28.561 [3]. ETSI's Zero-touch network and Service Management (ZSM) working group has likewise published technical reports such as GS ZSM-018 [5] on integrating Digital Twins (DTs) into zero-touch network management. At the global level, ITU-R's 6G study report (IMT-2030) [6] identifies DTs as a key 6G use case, and ITU-T Rec. Y.3090 [1] specifies requirements for 6G NDTs.

The NDT architecture, as presented in [1], is structured into three distinct layers: the application layer, the DT layer, and the

IEEE Global Communications Conference (GLOBECOM) 2025

physical network layer. This layered design ensures modularity and interoperability, where the application layer hosts services and interfaces for users and external systems, the DT layer provides the digital representation and analytical capabilities of the network, and the physical network layer encompasses the real-world infrastructure and data sources. Building on this reference architecture, the European-funded project 6G-TWIN proposed a functional architecture that further specifies the responsibilities and internal components of each layer [4]. While the full architecture is beyond the scope of this paper, we focus here on the DT layer, and more precisely on the persistent representation of the network, which includes the Unified Data Repository (UDR), basic models, and functional models, addressing in part the management aspects of the NDT

In our previous work [2], we detailed the design of the UDR and the basic models, which are built following a structural approach grounded in telecommunications standards such as 3GPP. However, that work did not address the development of functional models or the instantiation process for both model types. This paper addresses that gap by focusing on how real data is extracted from the network and mapped into the basic model using the GreyCat tool, enabling a live, structured digital representation that serves as the foundation for higher-level analytics and intelligent functions within the NDT.

B. Tools for NDT implementation

Graph-based modeling plays a central role in enabling the structural and functional representation of NDTs. Existing DT platforms such as Azure Digital Twin[1], Eclipse Basyx[2], FIWARE [7], and Eclipse Ditto [8] offer modeling capabilities suitable for industrial IoT and smart systems. However, they often lack native support for dynamic graph structures or seamless integration with evolving semantic models. Among these, Azure DT supports graph-based representations and schema definitions via Digital Twins Definition Language (DTDL)[3], but its proprietary nature and limited openness pose integration constraints. Eclipse Basyx implements the Asset Administration Shell (AAS) standard [9], offering hierarchical modeling but limited flexibility for advanced graph topologies. FIWARE, based on Next Generation Service Interfaces – Linked Data (NGSI-LD) [7], promotes interoperability and contextual data modeling, yet lacks built-in schema enforcement and explicit graph abstraction.

GreyCat [4] is a framework specifically designed for building large-scale DTs, offering a unique combination of imperative object-oriented programming, persistent indexing, and scalable memory management to support efficient model development [10]. It relies on a temporal many-world graph database structure, enabling advanced features such as "what-if" scenario simulation through database forking and time segmentation with polynomial regression for compact data representation. GreyCat also supports embedded Machine Learning (ML) capabilities, including Gaussian Mixture Models, for on-the-fly micro-learning. These features align closely with the needs of our work, enabling dynamic, time-aware graph representations, real-time data processing, and seamless integration with AI/ML-driven functional models.

[1]https://azure.microsoft.com/en-us/products/digital-twins
[2]https://projects.eclipse.org/projects/dt.basyx
[3]https://azure.github.io/opendigitaltwins-dtdl/DTDL/v3/DTDL.v3.html
[4]https://greycat.ai/

C. Functional models and radio coverage prediction

Functional models in NDTs are designed to perform a wide variety of tasks, such as prediction, optimization, and "what-if" analysis. These tasks are often domain-specific such as radio resource allocation in the Radio Access Network (RAN), or network slicing in the core network. Complex applications like teleoperated driving can be enabled by combining multiple functional models in multiple domains, such as coverage prediction followed by optimal radio resource allocation in RAN, and optimal slicing in the core. As an initial step toward enabling NDT-based applications, this work focuses on radio coverage prediction as an example of a functional model.

Since exhaustive field measurements are impractical, modern approaches rely on model-driven, data-driven, or hybrid techniques to estimate full coverage maps from sparse observations [11]. Historically, radio coverage prediction has been grounded in physics-based models such as ray tracing and ray launching [12], achieving high modeling accuracy. However, they are computationally intensive and require detailed knowledge of the environment, making them less feasible for large-scale or dynamic deployments. On the other hand, empirical models such as the 3GPP or COST pathloss models are computationally efficient and widely applicable, but often fall short in heterogeneous environments where propagation conditions deviate from idealized assumptions [13].

In response to these limitations, recent research has increasingly turned to data-driven approaches that learn signal propagation characteristics directly from measurements [11]. These methods aim to reconstruct full coverage maps from sparse observations by leveraging statistical inference or ML. Notable examples include Kriging or Gaussian Process Regression (GPR), which interpolate spatial measurements using covariance kernels to model spatial correlations. Deep learning-based techniques have also gained traction: Convolutional Neural Networks (CNNs), such as RadioUNet [14], treat coverage estimation as an image-to-image translation problem and have demonstrated near–ray-tracing accuracy with significantly reduced computational cost. For our implementation, we choose two data-driven models with different complexities and sensitivity to data variations, namely, GPR and CNN.

III. System design

This section presents the NDT instantiation and lifecycle in the ITU-T-based architecture, showing the data models structured within GreyCat. This is followed by the methodology used to assess the functional integrity and capability of NDTs.

A. Data Models Instantiation

The ITU-T recommendations define basic and functional models in broad terms, without providing implementation

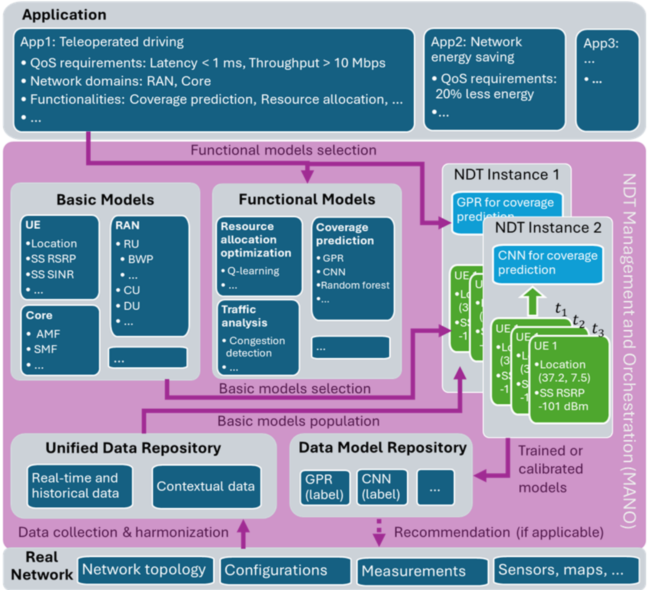


Figure 1: NDT architecture and instantiation process

details. In our work [2], we propose interpreting these models as structured libraries containing abstract data structures. These libraries are developed in a standardized and harmonized way to enable the creation of a universal, modular, and interoperable NDT. This approach is illustrated in Figure 1.

Basic models are graph-based representations of the network, capturing entities and their relationships. They encode configurations, measurements, and topological information in a format that is both flexible and extensible. Functional models build on top of this foundation by representing behaviors, optimization logic, and predictive capabilities. They often incorporate data-driven methods such as AI and machine learning to support intelligent decision-making. The UDR serves as the centralized storage component, aggregating both historical and real-time data from the physical network, sensors, and external contextual sources. When historical data is used to populate a basic model, the result is a time-indexed graph that represents the evolution of the network state over time.

The process of instantiating an NDT involves populating the basic model templates with data collected from the physical network and selecting the appropriate functional models based on the application and scenario. The lifecycle of an instantiated NDT is governed by a dedicated management component, referred to here as the NDT Management and Orchestration (MANO). The NDT MANO is responsible for creating, configuring, and maintaining NDT instances. While it can be fully automated—enabling the realization of the ZSM vision, we adopt a manual approach in this work. This allows us to focus on the implementation aspects of the NDT itself. Although manual configuration is suitable for limited-scale scenarios, it is not scalable, and automation through MANO must be addressed in future work.

To create an NDT instance, the MANO selects a set of functional models aligned with the target application. Concurrently, the basic models are populated with relevant data from the UDR, ensuring consistency with the chosen functional models. Calibration or training of the functional models on specific basic models yields parameterized versions tailored to the scenario. These trained models are stored in the NDT's data model repository, where they are labeled and managed by the MANO. This allows for the comparison of different algorithms or configurations and supports simulation-based evaluation before applying the models to the live network (only applicable for applications involving network control).

This concludes the description of the NDT instantiation process and lifecycle. In the following section, we demonstrate the implementation of these data models using GreyCat, which enables the creation of basic model libraries and natively supports various types of functional models.

B. Implementation in GreyCat

GreyCat serves as an implementation platform that notably stores knowledge and data about the network in its graph structure. It implements the data schema shown in Figure 2, supporting the data collected from the network elements: notably attributes and measurements from the New Radio Cell (NRCell): the Centralised Unit (NRCellCU) and its related Distributed Units (NRCellDU). Each NRCellDU is related to a carrier BandWith Part (BWP), and Time or Frequency Division Duplex (TDD / FDD) values. Most importantly it stores, connected User Equipment (UE) location and Synchronization Signal (SS) values.[5] It focuses in this study on key radio quality indicators, specifically: Reference Signal Received Power (RSRP), Reference Signal Received Quality (RSRQ), and Signal-to-Interference-plus-Noise Ratio (SINR).

Furthermore, GreyCat offers facilities to connect with data sources (e.g., MQTT connectors) and importer of data logs as well. One key aspect of Greycat is its capacity to handle data sharding, which enables rapid access to data at specific moments in time or space. This is achieved through a node pointer that consumes less memory and facilitates swift querying by efficiently navigating through the elements of the pointed objects. For instance, when dealing with SS values, we create a time node pointer. This pointer enables timely access to the desired value of the signal. It also offers support for prediction algorithms to extend coverage to unmeasured locations and visualization of the resulting coverage maps. It finally enables "what-if" scenarios to simulate network changes.

Although GreyCat natively supports the implementation of functional models, our current system design implements the considered GPR and CNN algorithms externally in Python, interfacing them with data stored in GreyCat. This architectural choice does not affect the validity or performance of the NDT evaluation, but it offered a more practical development path at this stage. In the following, we present the evaluation methodology for NDTs and position our specific implementation on the capabilities scale.

[5]The full GreyCat implementation is available in this repository: https://github.com/6GTWIN

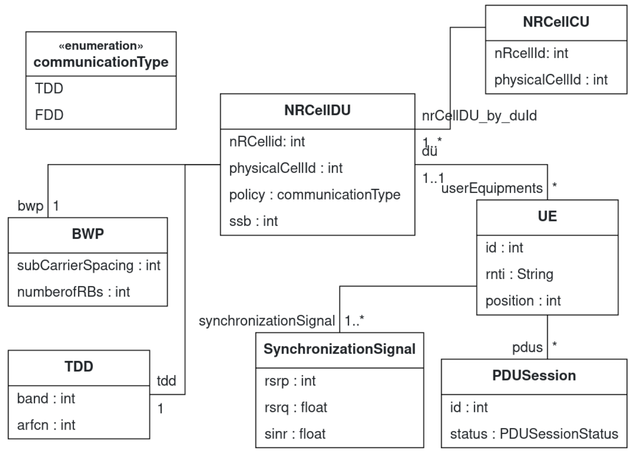


Figure 2: Excerpt of the basic model supporting our setup

C. NDT evaluation methodology

The assessment of our NDT is guided by the ITU-T classification of NDT capability levels [15], which defines a scale from Level 1 with limited to basic simulation tasks, to Level 5, where the NDT autonomously manages and reconfigures the physical network. However, this classification concerns the NDT MANO, which we have deliberately excluded from the scope of this work in order to focus on the implementation of NDT modeling. In terms of basic model integrity, our implementation currently corresponds to Level 3 [15]. This level enables the modeling of certain network topologies such as those commonly found in radio access or campus networks, and introduces limited logical relationships between network components. Given the limited scope and scale of the physical network considered in our use case, this level of modeling is sufficient to support our targeted functional models. The basic models encapsulate physical and provisioning attributes of network devices, as well as selective performance indicators. Regarding functional model integrity, the implementation reaches Level 1, which supports partial deployment of low-risk models, such as coverage estimation or site planning. At this stage, the functional models operate in an isolated environment without interacting with or modifying the physical network. This design aligns with the objectives of our current work, which focuses on model evaluation and prediction rather than network control or automation.

Other important metrics include the NDT accuracy and responsiveness based on data granularity and model quality. These metrics depend mainly on the available data and the data pipeline used for collection and harmonization, which will be investigated in future works with more complete NDT implementations. In our case, we aim to illustrate the modular approach and interoperable approach of the ITU-T architecture and the benefit of GreyCat in NDT modelling.

Details of the experimental setup, including data collection, partitioning strategy, and evaluation metrics, are provided in the following section.

IV. Performance evaluation

In this section, we present a comprehensive evaluation of the proposed NDT-based radio coverage prediction system. We begin by detailing the measurement campaign conducted in a controlled indoor environment, followed by a description of the data collection setup and methodology. Subsequently, we discuss the implementation of the prediction algorithms and analyze their performance based on standard accuracy metrics, comparing the predicted radio maps to ground-truth measurements.

A. Data collection scenario and setup

The measurement campaign was conductedon a laboratory floor of approximately 46 meters by 12 meters. A floor plan is shown in Figure 3.

The spatial coordinates corresponding to each measurement point were defined manually on the floor plan. These coordinates were not available natively from the network interfaces and were thus provided as external data inputs to the NDT. This approach is necessary because current network standards and taxonomies do not inherently provide location information for radio measurements.

A Software Defined Radio (SDR) based gNB node was deployed using OAIBOX which uses open-source OpenAir-Interface (OAI) framework [16]. The OAIBOX was chosen for its open-source flexibility, enabling full control of the 5G gNB stack using OAI. It allows precise logging and real-time customization, essential for measurement-driven experiments. A summary of the gNB configurations and components used are listed in Table-I.

Table I: Configuration Parameters and system components used for data collection

Parameters	**Value**	**Component**	**Specifications**
Duplexing Mode	TDD	gNB HW	USRP B210
Frame Format	DDDDDFUUUU	gNB SW	OAIBOX
Frequency	3809.28 MHz	UE	iPhone-14 Pro
Antenna Gain	3 dB	Antenna	Dipole
Bandwidth	40 MHz	MIMO Mode	1x1

B. Data collection Methodology

To collect the measurement logs, the UE was connected to the gNB. Upon a successful connection, the UE reports the key measurement parameters to the gNB. The primary focus for this study was on key radio quality indicators, specifically: RSRP, RSRQ, and SINR. Given that our transmission band is licensed for our dedicated use, interference levels are minimal. As a result, RSRQ and SINR metrics are less critical in the current context. Therefore, we focus on RSRP as the primary metric for coverage mapping, as it directly quantifies the received signal strength.

At each predefined location on the floorplan (see Figure 3), the UE remained stationary for approximately two minutes, during which measurement logs were continuously collected at the gNB. This report is logged during the data collection campaign for offline processing. A representation of such logs is shown in Figure 4.

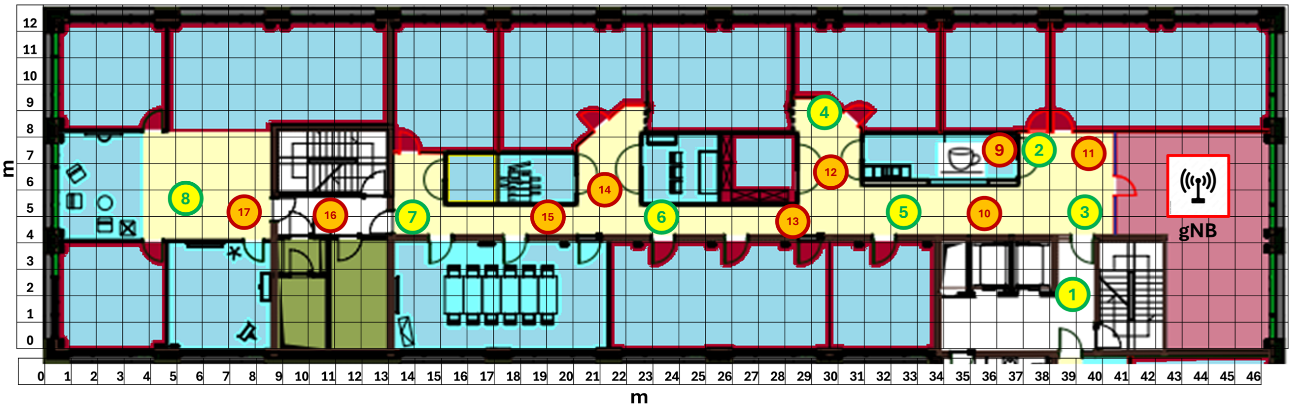
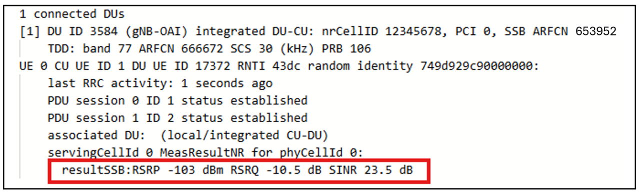


Figure 3: Floor plan of the indoor measurement scenario at LIST premises, showing the position of the gNB, the coverage area, and the georeferenced measurement points used for model training and evaluation. Axes are in meters (m)



```
1 connected DUs
[1] DU ID 3584 (gNB-OAI) integrated DU-CU: nrCellID 12345678, PCI 0, SSB ARFCN 653952
    TDD: band 77 ARFCN 666672 SCS 30 (kHz) PRB 106
UE 0 CU UE ID 1 DU UE ID 17372 RNTI 43dc random identity 749d929c90000000:
    last RRC activity: 1 seconds ago
    PDU session 0 ID 1 status established
    PDU session 1 ID 2 status established
    associated DU:  (local/integrated CU-DU)
    servingCellId 0 MeasResultNR for phyCellId 0:
      resultSSB:RSRP -103 dBm RSRQ -10.5 dB SINR 23.5 dB
```

Figure 4: RSRP, RSRQ and SINR logs reported by UE and displayed at gNB logs using OAI

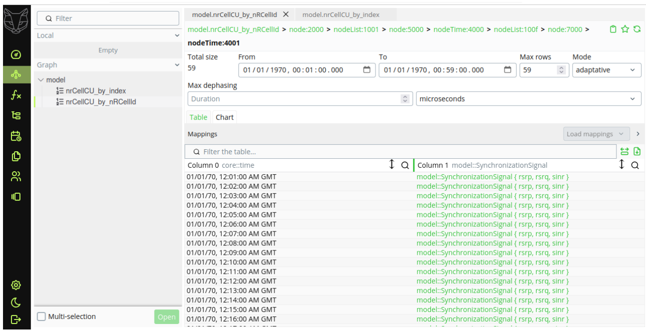


Figure 5: Greycat internal view, collection of synchronization signal for a UE at a given location.

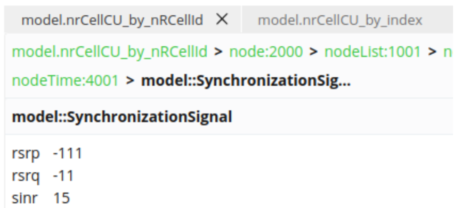


Figure 6: Greycat internal view of the SS at a given time

Once collected, the logs serve as input to the GreyCat NDT and are transformed to comply with the structural models illustrated in Figure 2. Focusing on the SS element which includes the RSRP values, measurements are represented as a collection of timestamped objects, each linked to a specific UE and location, as shown in Figure 5. At each timestamp, we can visualize the measured values, as illustrated in Figure 6. This temporal and spatial structuring enables efficient querying, manipulation, and prediction of data behavior across both time and location within the NDT.

C. Algorithms implementation

In the following, we provide a brief overview of the algorithms implementation in our set-up.

1) GPR implementation:

- **Input:** (x, y) coords and corresponding mean RSRP.
- **Kernel:** Radial Basis Function (RBF) kernel.
- **Training:** Hyperparameters (length scale).
- **Prediction:** Model queried over a grid covering the area.
- **Computation Cost:** $\mathcal{O}(N^3)$, manageable with small N.

2) CNN implementation:

- **Input:** 2D grid of mean RSRP values at given points; missing values masked.
- **Architecture:** Simple CNN with two conv. layers.
- **Loss Function:** Mean Squared Error (MSE) computed only on known points.
- **Training:** Trained directly on available measurements without data split.
- **Prediction:** Full map inferred in a single forward pass after training.

D. Numerical evaluation

The predicted RSRP maps for the chosen methods are illustrated in Figure 7 and Figure 8, which visualize the interpolated signal strength over the indoor area using the trained models. These visualizations highlight the differences in prediction smoothness, spatial sensitivity, and generalization behavior between the two methods. We note that the first eight measured locations, highlighted in yellow in Figure 3, were used for training only. The remaining points are used for testing.

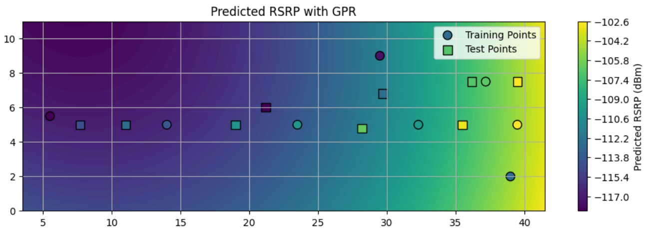


Figure 7: Predicted RSRP map using GPR

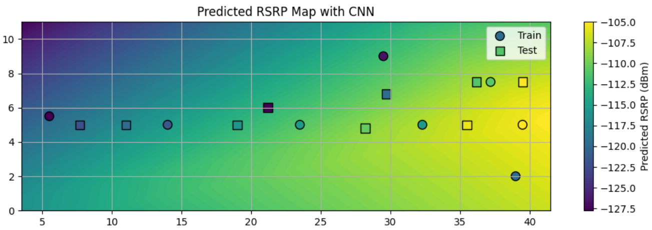


Figure 8: Predicted RSRP map using CNN

Table II presents the quantitative results of the prediction performance, computing the absolute prediction error between the mean of true RSRP measurements at each location against the predicted mean value from GPR and CNN models.

Both models are generally capable of capturing the spatial variation of the RSRP values in line-of-sight (LOS) areas. However, they exhibit reduced accuracy in non-LOS (NLOS) conditions, particularly in locations 11 and 14, where walls obstruct direct signal paths, highlighting the need for environment-aware models or additional context data to improve prediction fidelity in such conditions. Additionally, points 10, 13, and 15 show higher prediction errors due to the lack of nearby training data. Interestingly, CNN demonstrates greater resilience in such sparse-data scenarios, compared to GPR. The NDT enables model benchmarking and selection prior to deployment. By using the same structured data models from the GreyCat repository and associating them dynamically with different functional models, the NDT supports on-demand experimentation. This modular approach allows network planners to evaluate model behavior in specific environmental or data conditions—such as NLOS or sparse sampling, thereby improving flexibility for testing, optimization, and deployment decisions.

Table II: NDT accuracy using GPR and CNN

Location index	**True RSRP (dBm)**	**GPR predicted (dBm)**	**GPR |error| (dB)**	**CNN predicted (dBm)**	**CNN |error| (dB)**
9	-105.81	-107.31	1.50	-107.98	2.17
10	-101.43	-107.67	6.23	-106.66	5.23
11	-100.89	-104.93	4.04	-106.97	6.08
12	-112.79	-111.29	1.50	-110.04	2.75
13	-105.25	-111.82	6.57	-108.96	3.71
14	-119.35	-114.95	4.40	-112.92	6.43
15	-109.72	-115.41	5.69	-112.94	3.23
16	-112.89	-116.80	3.91	-116.69	3.80
17	-114.84	-116.88	2.04	-118.39	3.55

V. Conclusion

This paper presented a practical implementation of a NDT for indoor radio environments, focusing on the integration of structured data models and predictive algorithms using the GreyCat platform. The analysis of two algorithms was conducted entirely within the NDT, which underscores a key design advantage: the ability to benchmark and select algorithms using the same underlying structured data and models before committing to a deployment choice. Future work will focus on implementing more general scenarios with larger datasets, and on studying the automation aspects of NDT MANO in relation to NDT instantiation.

Acknowledgment

The 6G-TWIN project has received funding from the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union's Horizon Europe research and innovation program under Grant Agreement No 101136314.

References

[1] ITU-T, "Digital twin network - Requirements and architecture," Recommendation Y.3090, Feb. 2022.
[2] A. Zaki-Hindi, J.-S. Sottet, I. Turcanu, and S. Faye, "Network Digital Twins for 6G: Defining Data Taxonomy and Data Models," in *IEEE Conference on Standards for Communications and Networking (CSCN)*, IEEE, 2024, pp. 124–128.
[3] 3GPP, "TS 28.561 V0.2.0; Group Services and System Aspects; Management and orchestration; Management aspects of Network Digital Twins (Release 19)," Technical Specification 28.561, Apr. 2025.
[4] S. Faye, P. A. Soto Arenas, G. Volpe, and A. Zaki-Hindi et al., *A Functional Framework for Network Digital Twins*, Apr. 2025. [Online]. Available: https://doi.org/10.5281/zenodo.15304760.
[5] ETSI, "GS ZSM 018 V1.1.1; Zero-touch network and Service Management (ZSM); Network Digital Twin for enhanced zero-touch network and service management," Group Specification, Dec. 2024.
[6] ITU-R, "M.2160-0; M Series: Mobile, radiodetermination, amateur and related satellite services Framework and overall objectives of the future development of IMT for 2030 and beyond," Recommendation, Nov. 2023.
[7] U. Ahle and J. J. Hierro, "FIWARE for Data Spaces.," *Designing Data Spaces*, vol. 395, 2022.
[8] J. Robles, C. Martín, and M. Díaz, "OpenTwins: An open-source framework for the development of next-gen compositional digital twins," *Computers in Industry*, vol. 152, p. 104 007, 2023.
[9] R. Heidel, *Industrie 4.0: The reference architecture model RAMI 4.0 and the Industrie 4.0 component*. DIN Media GmbH, 2019.
[10] F. Fouquet, T. Hartmann, C. Cecchinel, and B. Combemale, "GreyCat: A Framework to Develop Digital Twins at Large Scale," in *Proceedings of the ACM/IEEE 27th International Conference on Model Driven Engineering Languages and Systems*, 2024.
[11] B. Feng, M. Zheng, W. Liang, and L. Zhang, "A Recent Survey on Radio Map Estimation Methods for Wireless Networks," *Electronics*, vol. 14, no. 8, p. 1564, 2025.
[12] Z. Yun and M. F. Iskander, "Ray tracing for radio propagation modeling: Principles and applications," *IEEE access*, 2015.
[13] S. Sun et al., "Investigation of prediction accuracy, sensitivity, and parameter stability of large-scale propagation path loss models for 5G wireless communications," *IEEE transactions on vehicular technology*, vol. 65, no. 5, pp. 2843–2860, 2016.
[14] R. Levie, Ç. Yapar, G. Kutyniok, and G. Caire, "RadioUNet: Fast radio map estimation with convolutional neural networks," *IEEE Transactions on Wireless Communications*, vol. 20, 2021.
[15] ITU-T, "Digital twin network - Capability levels and evaluation methods," Recommendation Y.3091, Dec. 2023.
[16] F. Kaltenberger et al., "OpenAirInterface: Democratizing innovation in the 5G Era," *Computer Networks*, vol. 176, p. 107 284, 2020.