Robust Learning and Reasoning
for Complex Event Forecasting

Project Acronym:	EVENFLOW
Grant Agreement number:	101070430 (HORIZON-CL4-2021-HUMAN-01-01 – Research and Innovation Action)
Project Full Title:	Robust Learning and Reasoning for Complex Event Forecasting

DELIVERABLE

D6.4 – Final Version of the Integrated EVENFLOW Prototype (PU version)

Dissemination level:	PU - Public, fully open
Type of deliverable:	R - Document, report
Contractual date of delivery:	31 December 2025
Deliverable leader:	INTRA
Status - version, date:	Final – v1.0, 2025-12-29
Keywords:	Integration, architecture, AI, neuro-symbolic, scalability, verification

Funded by the European Union

This document is part of a project that is funded by the European Union under the Horizon Europe agreement No 101070430. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the Commission. The document is the property of the EVENFLOW project and should not be distributed or reproduced without prior approval. Find us at www.evenflow-project.eu.

Executive Summary

This deliverable is the public version of deliverable D6.2, and both are exactly the same since no sensitive information was identified.

This deliverable presents the final architecture and integration outcomes of the EVENFLOW project, focusing on the realization, deployment, and validation of an integrated neuro-symbolic framework for robust learning and reasoning over complex event streams. EVENFLOW addresses the challenge of forecasting and explaining critical events in evolving, high-volume time-series data by combining neural representation learning with symbolic reasoning, formal verification, and scalable data management technologies.

Work Package 6 consolidates the architectural vision initially proposed earlier in the project into a concrete, deployable system that integrates heterogeneous tools developed across the consortium. The final EVENFLOW architecture adopts a modular, layered design that supports end-to-end data flows from raw perception streams to interpretable, verifiable forecasts. It enables interoperability between neural, symbolic, and verification components while accommodating the scalability and robustness requirements of real-world applications.

At the core of the architecture is a distributed data management backbone based on Apache Kafka and Kubernetes, deployed at NCSR facilities. This infrastructure supports high-throughput ingestion, buffering, and dissemination of streaming data across use cases and tools, while enabling secure access control and orchestration through standard cloud-native technologies. On top of this backbone, EVENFLOW integrates a diverse set of analytics and learning components, ranging from deep neural networks and neuro-symbolic automata to synopsis-based training optimization, scalable complex event recognition, and formal verification frameworks.

A key architectural contribution of EVENFLOW lies in the seamless integration of neuro-symbolic learning mechanisms. Several tools demonstrate how neural perception models can be coupled with symbolic structures—such as automata, logical rules, and event patterns—under probabilistic and differentiable semantics. In particular, the project shows that symbolic components, including automata guard conditions and temporal constraints, can be learned jointly with neural parameters using gradient-based optimization. This enables end-to-end training pipelines that retain interpretability while maintaining competitive predictive performance.

The integrated architecture is validated through a set of representative scenarios spanning multiple domains. These include scalable neuro-symbolic learning over streaming data, deadlock forecasting in robotic motion planning for Industry 4.0, leakage detection and sensor optimization in water infrastructure monitoring, synthetic trajectory generation and biomarker discovery in personalized medicine, and formal robustness verification of safety-critical perception models. Each scenario demonstrates a distinct integration path across the architectural layers, highlighting the interoperability of EVENFLOW components and their applicability to heterogeneous data modalities and operational settings.

Beyond forecasting accuracy, EVENFLOW places strong emphasis on robustness, explainability, and trustworthiness. Verification tools developed within the project enable certified robustness analysis of neural and neuro-symbolic models under bounded input perturbations, while rule-learning and symbolic reasoning components provide interpretable representations of learned behaviours. These capabilities address key regulatory and ethical requirements for AI systems operating in safety-critical and high-impact domains.

In summary, this deliverable documents the final EVENFLOW integrated scenarios, its architectural realization, and its validation across multiple scenarios. It demonstrates that neuro-symbolic approaches can be effectively engineered, deployed, and scaled in realistic settings, offering a viable path toward robust, interpretable, and verifiable AI systems for complex event forecasting.

Deliverable leader:	Ioannis Christou (INTRA)
Contributors:	Alesio Lomuscio (ICL), Nikos Katzouris (NCSR), Nikos Giatrakos (ARC), Davide Cirillo (BSC)
Reviewers:	Davide Cirillo (BSC), Nikos Giatrakos (ARC)
Approved by:	Athanasios Poulakidas, Vasiliki- Eleni Provopoulou (INTRA)

Document History			
Version	**Date**	**Contributor(s)**	**Description**
0.1	2025-12-24	INTRA	Initial version, based on D6.2 (the SEN version)
1.0	2025-12-29	INTRA	Version to be submitted after final QA

Table of Contents

Table of Figures

List of Tables

Definitions, Acronyms and Abbreviations

Acronym/ Abbreviation	Title
AI	Artificial Intelligence
CEF	Complex Event Forecasting
GDPR	General Data Protection Regulation
LSTM	Long short-term memory
ML	Machine Learning
RNN	Recurrent Neural Network
XAI	Explainable AI

Term	Definition
FAIR data	FAIR data are data which meet principles of findability, accessibility, interoperability, and reusability (FAIR)

1 Introduction

1.1 Project Information

EVENFLOW develops hybrid learning techniques for complex event forecasting, which combine deep learning with logic-based learning and reasoning into neuro-symbolic forecasting models. This approach combines neural representation learning techniques that construct event-driven features from streams of perception-level data with powerful symbolic learning and reasoning tools, which utilize such features to synthesize high-level, interpretable patterns for forecasting critical events.

To deal with the brittleness of neural predictors and the high volume/velocity of temporal data flows, the EVENFLOW techniques rely on novel, formal verification techniques for machine learning, in addition to a suite of scalability algorithms for training based on data synopsis, federated training and incremental model construction. The learnt forecasters will be interpretable and scalable, allowing for explainable and robust insights, delivered in a timely fashion and enabling proactive decision making.

EVENFLOW is evaluated on three use cases related to (1) oncological forecasting in healthcare, (2) safe and efficient behaviour of autonomous transportation robots in smart factories and (3) reliable life cycle assessment of critical infrastructure.

Table 1: The EVENFLOW consortium.

Number[1]	Name	Country	Short name
1 (CO)	NETCOMPANY-INTRASOFT	Belgium	**INTRA**
1.1 (AE)	NETCOMPANY-INTRASOFT SA	Luxemburg	**INTRA-LU**
2	NATIONAL CENTER FOR SCIENTIFIC RESEARCH "DEMOKRITOS"	Greece	**NCSR**
3	ATHINA-EREVNITIKO KENTRO KAINOTOMIAS STIS TECHNOLOGIES TIS PLIROFORIAS, TON EPIKOINONION KAI TIS GNOSIS	Greece	**ARC**
4	BARCELONA SUPERCOMPUTING CENTER-CENTRO NACIONAL DE SUPERCOMPUTACION	Spain	**BSC**
5	DEUTSCHES FORSCHUNGSZENTRUM FUR KUNSTLICHE INTELLIGENZ GMBH	Germany	**DFKI**
6	EKSO SRL	Italy	**EKSO**
7 (AP)	IMPERIAL COLLEGE OF SCIENCE TECHNOLOGY AND MEDICINE	United Kingdom	**ICL**

[1] CO: Coordinator. AE: Affiliated Entity. AP: Associated Partner.

1.2 Document Scope

The scope of this document includes the requirements fulfilment matrix, presentation of the updated final EVENFLOW architecture, and a detailed description of the integration scenarios established in the project to demonstrate the project results.

D6.4 is the public version of D6.2 and identical to it, as it was confirmed by the consortium that D6.2 contains no sensitive information.

1.3 Document Structure

This document is comprised of the following chapters:

Chapter 1 presents an introduction to the project and the document.

Chapter 2 presents the updated final EVENFLOW architecture.

Chapter 3 presents the EVENFLOW integrated scenarios.

Chapter 4 presents the conclusions.

2 Final EVENFLOW Architecture

2.1 Requirements for Integrated Systems

The EVENFLOW requirements were collected in an Excel file, [EVENFLOW integrated system requirements.xlsx](). The final architecture detailed in the next section, and the sub-systems and architectural components implemented, together, fulfil all the 12 major high-level requirements (R001-R012) set forth in the above-mentioned Excel spreadsheet.

2.2 Final EVENFLOW Architecture

The final architecture, based on the original logical view of the EVENFLOW architecture proposed in D6.1 "Architecture Design and Integrated System Specification (SEN version)," is shown in Figure 1 below.

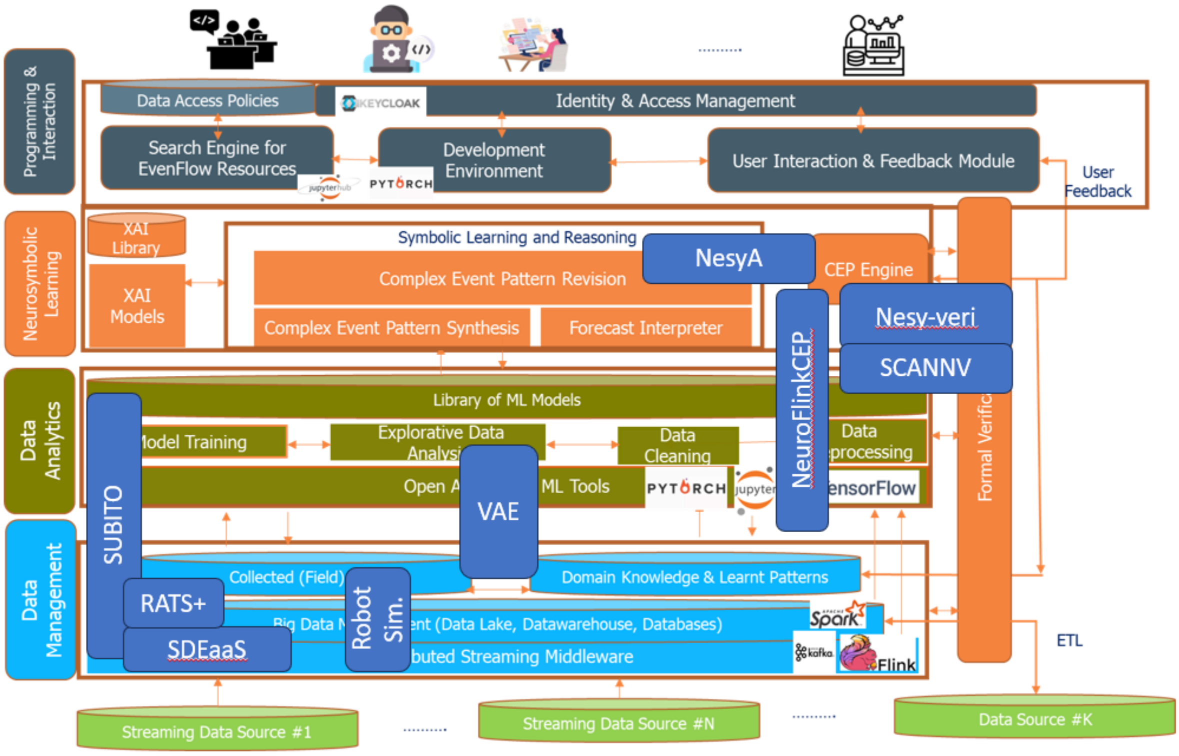

Figure 1: Final EVENFLOW Architecture Logical View.

In Figure 1 we show the extra components that have been developed in the context of EVENFLOW since the initial specification of the EVENFLOW architecture. These components are as follows:

1. **Robot Motion Simulator & Data Generator:** tool that simulates the motion of robots in a controlled environment and generates image datasets of what the cameras attached to the robots see as they move in their environment. Also outputs deadlocks when recognized.
2. **SuBiTO**: is an intelligent framework designed to optimize the trade-offs between training time and accuracy in real-time machine learning applications over Big Streaming Data. It tackles the challenges faced by Neural Networks (NNs) deployed in

high-speed, high-volume environments by continuously adjusting model parameters to ensure efficiency with minimal downtime.

3. **RATS+**: a set of scripts for the optimal scheduling of tumor simulations under various treatment options, suitable for use with scalable frameworks for parallel/distributed clusters/HPC.
4. **SDEaaS**: Synopses Data Engine as a Service combines the virtues of parallel processing and stream summarization towards delivering interactive analytics at extreme scale. It is built on top of Apache Flink/Dask and implements a novel synopsis-as-a-service (SDEaaS) paradigm.
5. **NeuroFlinkCEP**: a framework that integrates neural and symbolic Complex Event Recognition (CER) over a state-of-the-art Big Data platform, also optimizing neurosymbolic CER upon operating over IoT settings. NeuroFlinkCEP receives expressed patterns as extended regular expressions and automatically transforms them to FlinkCEP jobs per device.
6. **NesyA**: Neuro-symbolic Automata. Symbolic automata that combine the power of automata for temporal reasoning with that of propositional logic for static reasoning are a suitable formalism for expressing knowledge in temporal domains. Symbolic automata can be integrated with neural-based perception, under probabilistic semantics towards an end-to-end differentiable model. NeSyA (Neuro Symbolic Automata) is shown to either scale or perform more accurately than previous NeSy systems in a real-world event recognition task.
7. **VAE**: a tool for creating pseudo-temporal trajectories of the progression of tumor in a patient.
8. **Nesy-veri**: Neuro-symbolic Verification. We compile the symbolic part into a computational graph, stack it on top of the neural network, port the combined thing into ONNX, and use of-the-shelf verifiers to propagate bounds through the neural and symbolic parts in one go.
9. **SCANNV**: Scalable Neural Network Verification (SCANNV) approach, a set of techniques that can reduce the execution time of parallel neural network (NN) verification by optimising (i) how an input property is split into subproblems that can be verified in parallel and (ii) how these subproblems are scheduled for execution.

2.3 Final EVENFLOW Architecture Implementation and Deployment View

The implementation of the EVENFLOW architecture is based on the following Deployment View, showing how a number of data management services, based on the Kafka messaging architecture, has been installed at NCSR premises:

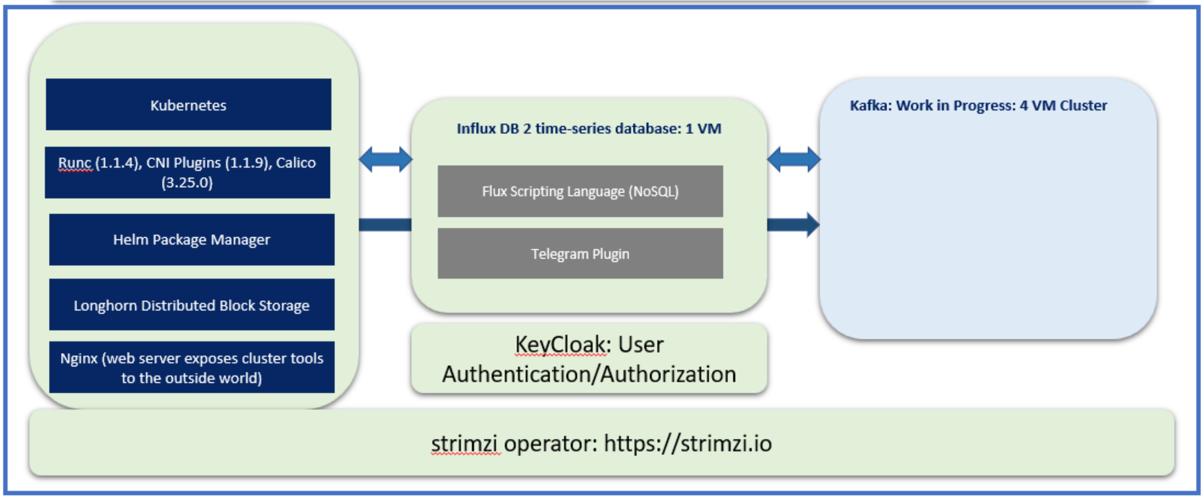

Figure 2: EVENFLOW Data Management components Final Deployment View.

The installation of the components shown in Figure 2 was done with the help of the Strimzi operator: Strimzi provides a way to run an Apache Kafka cluster on Kubernetes in various deployment configurations, making it particularly easy to handle KeyCloak authentication & authorization services, as it includes the tool in its bundle. The following software is up and running at the NCSR servers:

1. Kubernetes cluster
2. Containerd
3. Runc
4. Helm
5. Longhorn
6. Nginx
7. Zookeeper
8. Kafka broker services

The above components were deployed at INTRA and NCSR machines by M18 of the project, as part of the initial work reported in D6.1 “Architecture Design and Integrated System Specification (SEN version)”.

The Strimzi operator was installed at NCSR machines after M18 in order to facilitate the deployment of both the Kubernetes modules as well as the KeyCloak authentication/authorization service, together with the appropriate user credentials configurations.

9. KeyCloak

The connectors are the set of command-line scripts developed by INTRA for use by any partner who wishes to use them (e.g. DFKI) in order to create/modify/delete topics in the Kafka cluster running on NSCR machines, and read/write into these topics. The set of such scripts along with the appropriate credentials were given to interested partners (as modified appropriately Strimzi operator scripts). The development process followed an incremental approach whereby initial standard Kafka shell scripts were distributed for use with the Kafka cluster installed at INTRA-supplied servers (see D6.1 “Architecture Design and Integrated System Specification (SEN version)”), and later improved to work with KeyCloak authentication/authorization functions provided in the new Strimzi-based installation of the Kafka cluster installed at NCSR machines. Tests were made through the DFKI ROS simulation runs to ensure the read/write of messages to the EVENFLOW Kafka message bus.

3 Final EVENFLOW Integrated Scenarios

In this chapter, we present the final integrated scenarios that we use in EVENFLOW to demonstrate the usefulness and applicability of our developed tools, and their interoperability. For each scenario, we first present a schematic that corresponds to a *DFD level 0 diagram*, or equivalently a *Use-Case Context Diagram*) that shows the main tools involved in the scenario, and their data flows. We then proceed to present specific details about the scenario.

The first integrated scenario involves the cooperation of the NeSy tools (developed by NSRC) with the scalability toolkits developed by ARC. The data used in this demonstration scenario are variants of the NIST/MNIST open-access dataset of digitized hand-written digits. For more information, see D5.1 “Interim Version of Verification and Scalability Techniques” and D4.2 “Final Version of Online Neuro-Symbolic Learning & Reasoning Techniques.”

The second integrated scenario involves using the NeSy toolkit (NesyA, Wayeb -see D3.3 “Final Use Case Evaluation”) to forecast robot path movement and robot “deadlock detection” using the DFKI image dataset (see D3.3 “Final Use Case Evaluation”).

Data are stored in the Kafka installation deployed in the NCSR servers for the EVENFLOW data architecture needs.

The third scenario revolves around water infrastructure Use-Case (again, see D3.3 “Final Use Case Evaluation”) A number of tools have been developed and tested against the datasets provided by EKSO. Specifically, INTRA has developed several predictive ML tools based on Neural Networks (combination of RNN and CNN) as well as based on rule-learning algorithms that have produced a number of important results for both the 2nd and 3rd datasets provided by EKSO. NCSR has also worked with CNN-based forecasting tools to develop NN-based tools for detecting leakages in the 2nd dataset provided by EKSO. The relevant datasets are available on the Kafka message bus deployed at the NCSR servers.

The fourth scenario is about the bio-informatics Use-Case (BSC) and is concerned with the creation of synthetic trajectories that “simulate” the transition of early-stage cancer patients to late-stages (see D3.3 “Final Use Case Evaluation”) Besides the synthetic generation of path trajectories, a number of tools were developed to identify transcriptomic biomarkers indicative of the progress of a tumor.

The fifth scenario showcases the verification capabilities of the EVENFLOW tools, with the help of the image dataset developed in the Industry 4.0 Use Case. See D5.2 “Final Version of Verification and Scalability Techniques.”

Other minor scenarios are described in Section 3.6.

The particular ways in which data flows between various tools in each scenario is detailed in Table 2 below.

Table 2: Data Flows Between Tools.

Scenario Name / #	Tool Name	Tool Input	Tool Output
EKSO / 3.3	QARMA	Time-series data stored in RDBMS	Rules stored in RDBMS
EKSO / 3.3	QARMA-classifier	1. Time-series data stored in CSV file 2. Rules stored in RDBMS	Leakage Detection Decision (Yes/No)
EKSO / 3.3	LSTM	1. Time-series data stored in CSV file	Leakage Detection Decision (Tap#)
BSC/KIRC and MB	Synthetic Data Generation Pipeline	Gene expression data of real patients affected by renal carcinoma (KIRC) or medulloblastoma (MB)	Synthetic gene expression data. Either trajectories (KIRC) or static augmentation (MB)
BSC/KIRC	Static and dynamic classifier	1. Gene expression data of real patients and synthetic patients. 2. Metadata on the cancer stage of real patients.	Classification of synthetic timepoints in cancer trajectories.
BSC/KIRC	Dynamic Gene Set Enrichment Analysis	1. Synthetic gene expression trajectories. 2. Reactome pathways	Enrichment score of each interpolated time point and pathway.
ARC	SDEaaS	Training Data Streams	Summaries (mainly stratified samples) of training data streams
ARC	SuBiTO	Summaries of Training Data Streams, Unlabelled Data Streams	Up-to-date trained neural/neurosymbolic Model, Prediction Streams
ARC	NeuroFlinkCEP	Neural Model, Logical Workflow, Device Registry,	Recognized Complex Event Streams

Scenario Name / #	Tool Name	Tool Input	Tool Output
		Statistics, Perceptual Streams	
ICL	Nesy-veri	Trained Neural Network, Dataloader, Epsilon noise perturbations, verification method (IBP / CROWN+IBP)	Robust accuracy of the verified models, lower and upper bounds of the neural network outputs under adversarial noise perturbations

3.1 Scalable NeSy

3.1.1 General Description

This particular scenario works only with the MNIST dataset (http://yann.lecun.com/) and involves using both the scalability tools from ARC (in particular, the SuBiTO toolkit) as well as the Neuro-symbolic toolkits (NesyA) developed by NCSR. The general data flow is shown in the figure below.

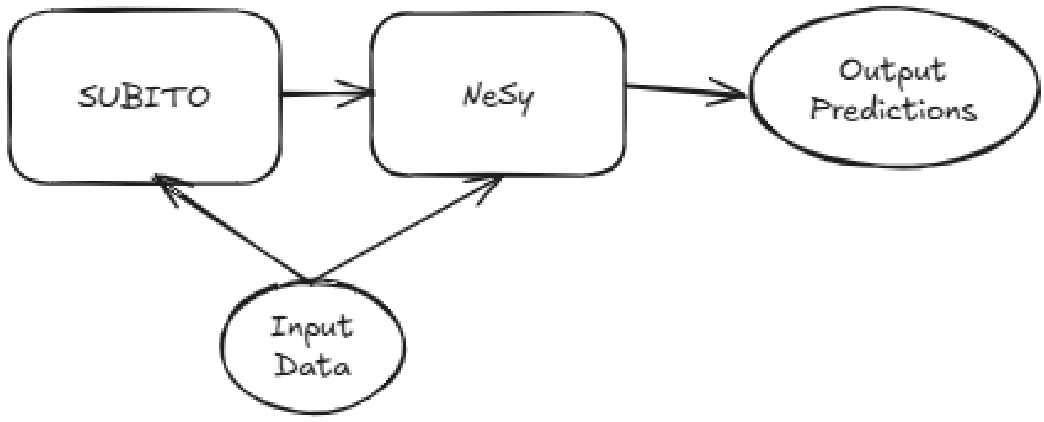

Figure 3: Data flow in integrated scenario #1.

For more information, see D5.1 “Interim Version of Verification and Scalability Techniques” and D4.2 “Final Version of Online Neuro-Symbolic Learning & Reasoning Techniques.”

The partners involved include ARC, NCSR and INTRA. The scenario will demonstrate the use of scalable synopses developed in the SuBiTO toolkit feeding the NeSy tools developed by NCSR (NesyA – neurosymbolic automata) to recognize specific digit sequences as specified by the rules of the automata at the end of the NesyA tool. INTRA coordinates the development of this scenario.

There are no related Use-Cases with this scenario, but it is important for demonstrating the collaboration of two of the major tools developed in EVENFLOW.

3.1.2 Scenario Details

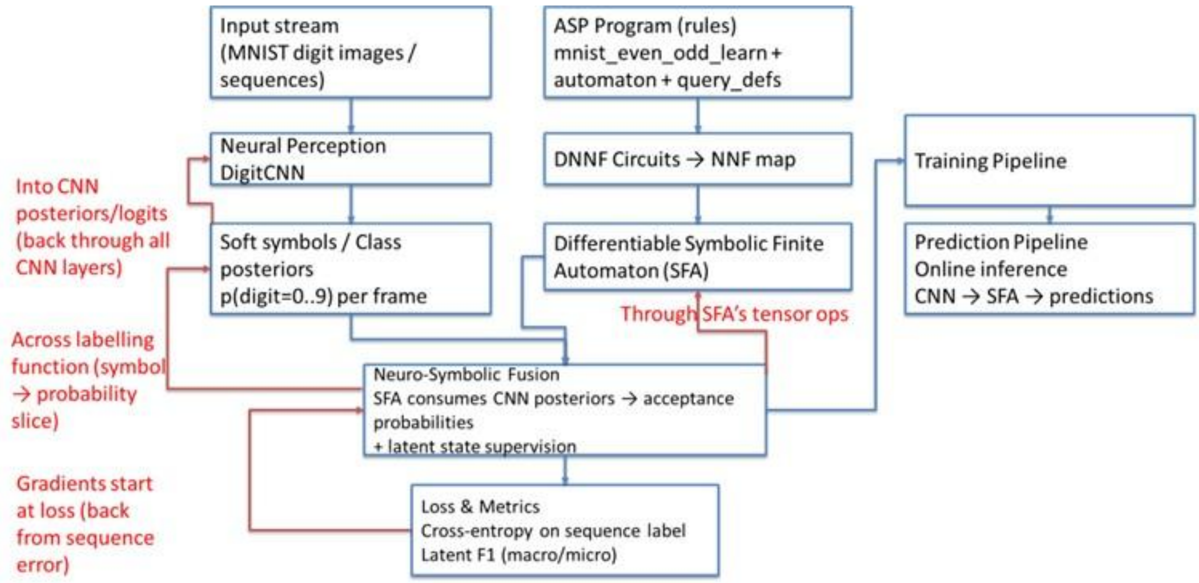

Figure 4: Overview of the NeSy-SuBiTO Scenario.

An overview of the NeSy-SuBiTO scenario is presented in Figure 4. In this scenario, an automaton is used as a “teacher" for a Convolutional Neural Net (CNN), which learns to predict MNIST digits from MNIST image sequences that satisfy or not the pattern specified by the automaton. Therefore, training is performed in an indirect fashion, without explicit labels at the image level, but with “downstream" labels, at the sequence level only. SuBiTO's optimizer is used to optimally adjust, at training time, the amount of additional training required, given the hybrid neuro-symbolic (CNN+automaton) model’s current predictive performance and the characteristics of the training data. Further details are provided in Section 3.2 of D5.2 “Final Version of Verification and Scalability Techniques.”

3.2 DFKI Scenario

3.2.1 General Description

This scenario demonstrates how neuro-symbolic tools (NesyA) can forecast and predict specific conditions in robot motion planning and execution, most importantly deadlock (which is the condition that occurs when a robot detects an obstacle -most likely another robot- in its immediate trajectory of motion.) For more information see D3.3 “Final Use Case Evaluation”. This workflow is aligned with the Industry 4.0 Use Case.

The data flow of this integrated scenario #2 is shown in the figure below:

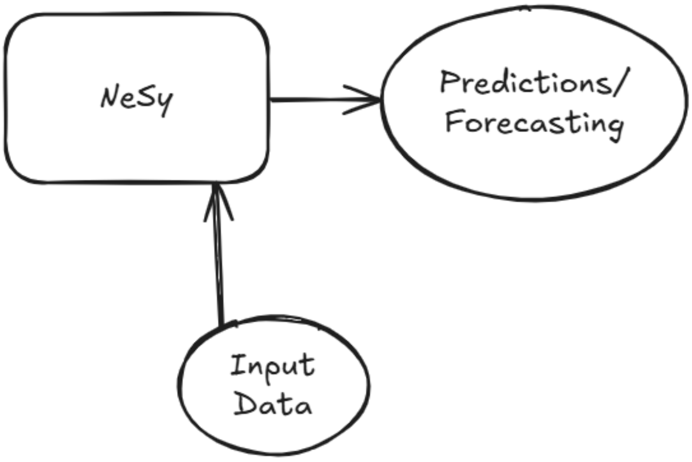

Figure 5: Data flow in integrated scenario #2.

The partners involved are DFKI (data providers), INTRA (message bus integrator), and of course NCSR (neuro-symbolic-based forecaster development).

Data reside in the Kafka message bus deployed for this purpose at NCSR servers, and are image data generated by the DFKI robotic environment simulator (Robot Motion Simulator and Data Generator in Section 2.2).

These data feed the NesyA toolset which outputs (after training) an appropriate forecaster that learns to forecast the imminent creation of a deadlock involving moving robots.

3.2.2 Scenario Details

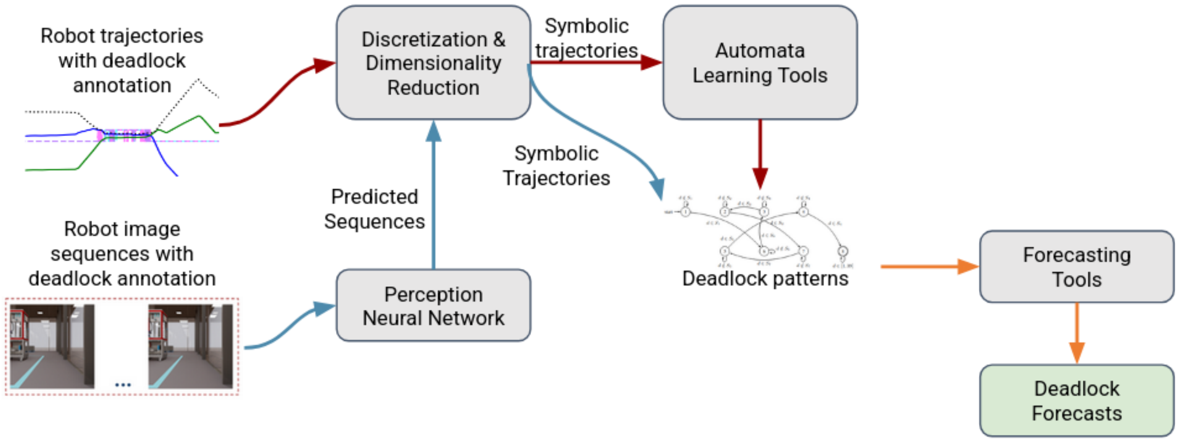

Figure 6: Overview of the DFKI deadlock forecasting scenario.

The DFKI forecasting scenario combines image and time series robot data, perception neural networks with symbolic deadlock patterns, as well as event forecasting tools, to deliver timely deadlock forecasts that can be used for re-planning and intimately reducing operational downtimes. An architectural overview of the approach and the involved components is presented in Figure 6, while further details and evaluation results are provided in Deliverable D3.3.

3.3 EKSO Scenario

3.3.1 General Description

This scenario shows how rule-learning systems as well as specialized neural architectures can learn to detect leakages in water pipes infrastructures, and even optimize the placement of (cheap) sensors in such pipes. For more information see D3.3 "Final Use Case Evaluation".

The data flow in this scenario is as shown in the figure below. The time-series data reside in the Kafka message bus deployed at NCSR servers.

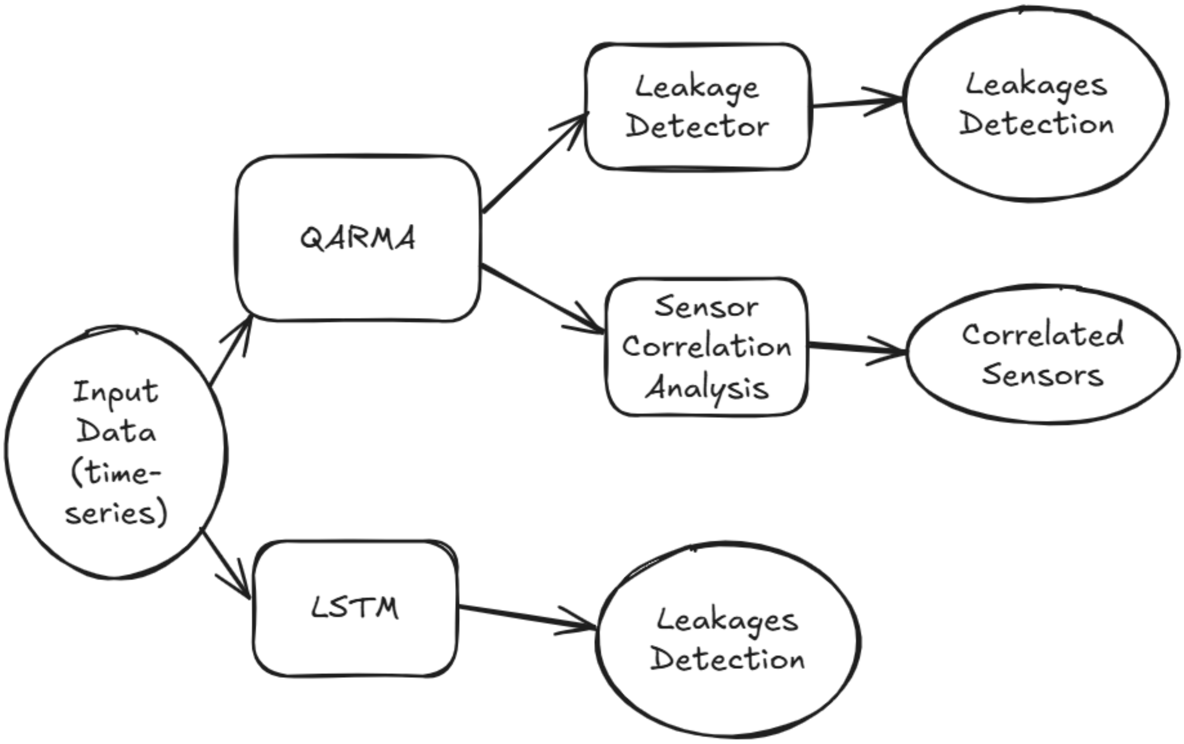

Figure 7: Dataflow in integrated scenario #3.

Also, use LSTM-approach and automata-based to detect patterns and leakages on 1st EKSO dataset (NCSR).

The partners involved are: EKSO (data provider, 3 distinct real-world datasets, not simulations), INTRA (data analyses and rule-learning system provider), and NCSR (neural-based approaches)

The scenario implements the Infrastructure Maintenance Use Case of the EVENFLOW project.

3.3.2 Scenario Details

The rule-based approach detects correlations between sensors via the QARMA rule-extraction algorithm family (INTRA) on the 3rd EKSO dataset. It also learns rules that apply when using only the most distant (from the leakage source) sensor, showcasing that even with a single sensor placed approximately 100m from the leakage source, we can detect leakage with accuracy above 95%! This result indicates that instead of placing sensors every 10m as done in the scenario developed by EKSO in Burgas municipality, it is sufficient to place sensors 100m apart (a 10x improvement in sensor placement requirements) and still be able to achieve a 95% accuracy with data corresponding to 10sec. By applying further aggregation heuristics over a 10 min. period, the detection accuracy can be shown to rise to 100%.

For the 3rd and final EKSO dataset, a total of approximately 20.000 rules were derived, all of them showing how every sensor value correlates with the reading value of every other sensor in the dataset, justifying to a high degree the fact that even the most distant sensor is sufficient to detect leakages (see figure below).

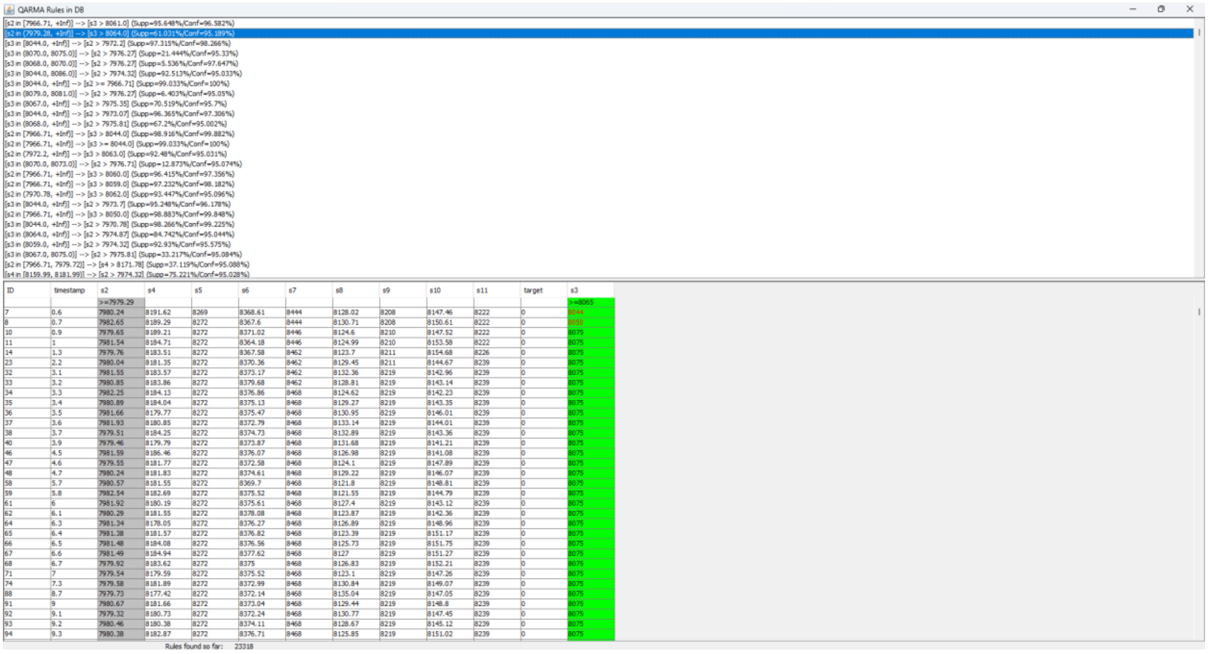

Figure 8: Results of using QARMA for analysing the 3rd and final EKSO dataset. Every sensor can appear as a pre-condition for the value reading of every other sensor in this dataset. The screenshot shows a GUI developed for the QARMA family of algorithms, visualizing the derives rules in the bottom part of the screen.

3.4 BSC Scenario

3.4.1 General Description

The scenario intends to demonstrate the purpose of the Personalized Medicine use case which is to utilize EVENFLOW technology to forecast events related to critical stages in tumor evolution in oncological, virtual patient applications. A first major outcome of this use case has been the development of a Variational Auto-Encoder (VAE) that allowed the development of “virtual”, synthetic trajectories that follow the evolution of breast cancer, based on a number of real patients’ data. Following the “pairing” or “matching” of endpoints (patient data points that indicate early or late stages in the evolution of the disease from a molecular biology point of view using gene expression data) a neuro-temporal approach using LSTM technology allowed for the identification of certain genes that appear to be highly correlated with disease progression.

Disentangled latent spaces were extracted from the learnt VAE models. Such spaces map covarying, unobserved (latent) factors in the data generation process (e.g. the gene expression profiles of a cohort of patients) to specific observed parameters (generative factors) of the model that generates the data, (e.g. dysregulation of a set of genes).

The major partners in this scenario are BSC (data point provider, VAE developer) and NCSR (path generation, detection of biomarkers for disease progression). For more information, see D3.3 "Final Use Case Evaluation."

3.4.2 Scenario Details

Building on the synthetic trajectory generation pipeline described above, BSC explored data-driven strategies for pairing patient representations across disease stages in high-dimensional molecular feature spaces. By operating in latent representations learned from gene expression profiles, the approach enables the construction of smooth disease progression trajectories that interpolate between early and late cancer stages while preserving biologically meaningful structure. These trajectories allow downstream temporal analysis techniques to be applied in settings where longitudinal patient data are scarce or incomplete. The generated synthetic paths were subsequently used to study the temporal activation of molecular signatures and pathway-level dynamics, supporting the identification of candidate biomarkers associated with disease progression. Although primarily evaluated in an offline, exploratory setting, this approach illustrates how synthetic trajectory generation can complement neuro-symbolic forecasting and reasoning pipelines by enriching training and analysis data in personalized medicine scenarios.

3.5 Verification Scenario

3.5.1 General Description

The Verification scenario revolves around the verification of particular neural networks accepting raw data (in this case, image data). We used data from the DFKI synthetic image data produced by the DFKI Robot Simulation and Data Generator toolkit.

NeSy tools were used to assist verification methods to produce better bounds and subsequently NN verification results.

The data flow for this scenario is shown in the figure shown below.

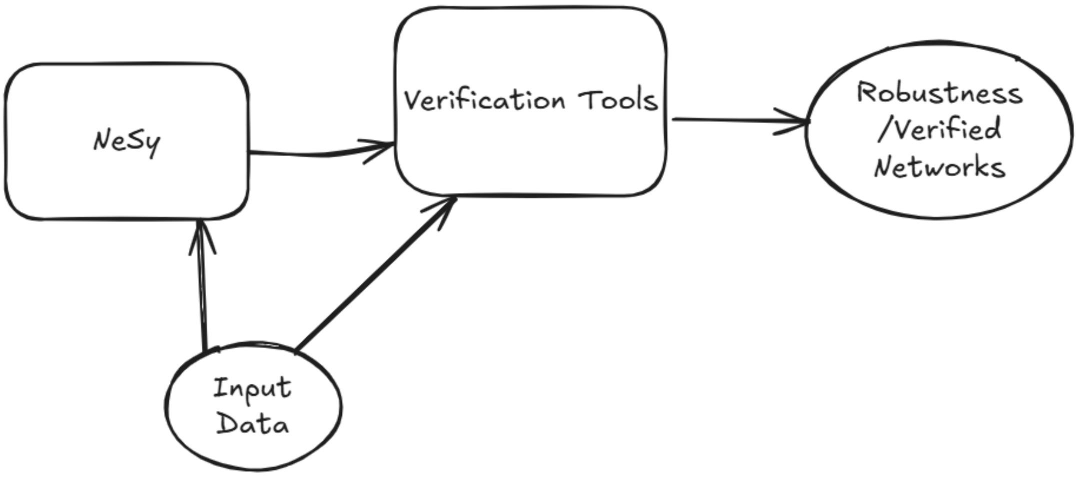

Figure 9: Data flow in integrated scenario #5.

The relevant partners are DFKI (data provider), NCSR (neuro-symbolic toolkit provider), and ICL (verification toolkit provider). For more information, See D5.2 "Final Version of Verification and Scalability Techniques."

3.5.2 Scenario Details

The verification scenario is applied to the industry 4.0 neuro-symbolic collision-avoidance model developed for the DFKI robotic perception use case. The neural component of the system was trained using a 5-fold cross-validation protocol, ensuring that each fold produced an independently optimized model. This approach enables a robust evaluation of safety guarantees by examining verification performance across diverse training distributions. Verifying all five splits provides insights into the stability, generalization, and reliability of the model's safety-critical behaviour.

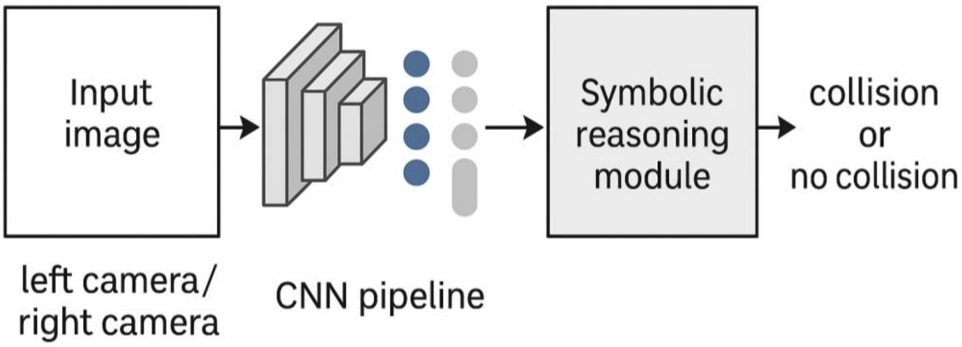

Figure 10: Verification pipeline for Neuro-symbolic system using CNN architecture.

Robustness evaluation was conducted by computing certified guarantees under multiple ϵ-bounded perturbation magnitudes applied to input camera images. These perturbations emulate realistic noise patterns and environmental variations observed during robot navigation, such as lighting changes or minor sensor distortions. Verification was performed using a framework that we devised extending auto_LiRPA and interval bound propagation (IBP) to generate sound lower and upper bounds on model outputs. These certified bounds allow us to determine whether the model's collision-prediction decisions remain unchanged within specified perturbation regions, thereby evaluating its formal robustness.

The resulting certified robustness values were aggregated across all five cross-validation folds, enabling a systematic comparison of verification performance and highlighting the degree to which the neuro-symbolic system maintains robustness under input perturbations.

3.6 Other Scenarios

Most partners have developed small independent scenarios to demonstrate (or simply showcase) the tools they have developed in the EVENFLOW project. These scenarios are explained in the deliverables of the respective Work Packages.

4 Discussion & Conclusions

The final architecture and integration work presented in this deliverable confirm the technical feasibility and practical relevance of the EVENFLOW approach to neuro-symbolic complex event forecasting. Through the systematic integration of neural learning, symbolic reasoning, scalable data processing, and formal verification, EVENFLOW demonstrates that hybrid AI systems can move beyond conceptual prototypes toward deployable, end-to-end solutions.

One of the main conclusions of the project is that neuro-symbolic models can be trained and executed effectively in streaming and evolving environments, provided that appropriate architectural abstractions and integration mechanisms are in place. The modular design of the EVENFLOW architecture proved essential in accommodating heterogeneous tools developed by different partners, allowing them to interoperate through well-defined data flows and deployment patterns. The use of a shared messaging backbone and containerized deployment facilitated experimentation, scalability, and reuse across scenarios.

From a methodological perspective, EVENFLOW shows that symbolic structures—such as automata, logical rules, and event patterns—can be tightly coupled with neural components without sacrificing learnability or scalability. The successful application of gradient-based optimization to hybrid neural-symbolic models, including the learning of symbolic guard conditions, represents a significant step toward unifying data-driven and knowledge-driven approaches. At the same time, the project highlights the importance of complementary techniques, such as synopsis-based training optimization and distributed learning, to manage the computational demands of such models.

Another important outcome concerns robustness and trustworthiness. The integration of formal verification techniques into the learning and evaluation pipeline enables the assessment of certified guarantees for safety-critical predictions. This is particularly relevant for applications such as autonomous systems and infrastructure monitoring, where predictive accuracy alone is insufficient. The verification scenarios illustrate how neuro-symbolic representations can support tighter bounds and more informative robustness analyses compared to purely neural models.

Despite these achievements, the project also identified several limitations and open challenges. Integration complexity remains non-trivial, especially when combining tools with different assumptions about data formats, execution models, and learning workflows. While the architecture supports interoperability at the system level, deeper semantic integration between components often requires additional engineering effort. Moreover, although several tools have demonstrated scalability in controlled settings, further work is needed to evaluate their performance and stability under sustained real-world operational loads.

Looking forward, several future research and development directions emerge from EVENFLOW. From a scientific perspective, there is clear potential to extend neuro-symbolic learning to richer forms of temporal and probabilistic reasoning, as well as to explore tighter integration between learning and verification during training rather than post-hoc analysis. Advancing methods for continual and lifelong learning in neuro-symbolic systems remains another important avenue, particularly for long-running deployments with evolving data distributions. From an engineering standpoint, future work could focus on further standardization of interfaces and workflows, enabling easier composition and reuse of neuro-symbolic components across projects and domains. Strengthening support for automated deployment, monitoring, and lifecycle management would facilitate the transition from research prototypes to production-ready systems. In addition, closer alignment with emerging regulatory frameworks for trustworthy AI could help position neuro-symbolic architectures as a practical response to forthcoming compliance requirements. In particular, the alignment with the EU AI Act (Regulation (EU) 2024/1689) and the Ethics Guidelines for Trustworthy AI (High-Level Expert Group on AI, European Commission, 2019) could strengthen the case for neuro-symbolic architectures as a compliance-oriented design choice rather than a purely academic alternative. In this sense, EVENFLOW demonstrates that by embedding explicit reasoning, traceability, and human-interpretable decision paths into learning systems, neuro-symbolic approaches directly support regulatory expectations around explainability, accountability, and oversight, particularly for high-risk applications. As regulatory pressure increases, these architectures may offer a pragmatic way to operationalize legal and ethical requirements without sacrificing adaptive performance.

In conclusion, EVENFLOW provides strong evidence that neuro-symbolic AI is not only a promising research paradigm but also a viable architectural choice for complex event forecasting in real-world settings. The project has laid a solid foundation for future work that builds on its architectural principles, tools, and lessons learned, paving the way toward more robust, interpretable, and trustworthy AI systems.