

**depict and describe the Figure 2 model diagram that is available on FullText.pdf**

The model diagram from "A Framework for Automatic IT Architecture Modeling: Applying Truth Discovery" illustrates an 11-step process for automating the creation of IT architecture models from diverse data sources [1]. The framework is designed to be independent of specific tools or metamodels, providing a structured method for handling data conflicts and creating comprehensive models [1].

[image:1]

**Overview of the Automatic Model Creation Framework**

The primary goal of this framework is to support the automatic generation of IT architecture models by using heterogeneous, non-streaming data [1]. It establishes a common language and data structure to standardize information gathered from multiple sources. The process is divided into several key stages: initial setup, data transformation and storage, analysis and conflict resolution, and final model generation [1].

The main steps of the process are [1]:

1. **Select Metamodels** The initial step is to decide which metamodels will be used. The framework is adaptable and can work with various metamodels, such as ArchiMate for enterprise modeling, securiCAD for threat modeling, or Eclipse Modeling Framework-based Fault-Tree Analysis (EMFTA) [1]. Each metamodel is defined by a specification that outlines its elements, associations, and properties [1].
2. **Identify Data Sources** The next step involves identifying relevant data sources that cover the necessary architectural layers. These can include network scanners, configuration data, and inventory systems [1]. The trustworthiness of data varies depending on the source type and how frequently data is collected [1].
3. **Define Common Language and Structure** A common data structure and language are created to standardize the input from various sources. This is essential for merging heterogeneous data. The common language is typically based on the most expressive metamodel being used [1].
4. **Set Up a Central Database** A central, hybrid database is established to store the collected data using the common language and structure. This ensures data consistency and scalability [1].
5. **Create Data Source Adapters** Adapters are built to transform data from the native format of each source into the common data structure and language defined in step three [1].
6. **Store Transformed Data** The transformed data is then regularly stored in the central database. The framework keeps temporal versions to track changes in the IT architecture over time [1].
7. **Create Model Language Adapters** A second set of adapters is created to transform the data from the common language into the specific format required by the analysis tools and their respective model languages [1].
8. **Save Transformed Model** The model, now in its specific language format, is saved separately in the database before analysis [1].
9. **Run Analysis for Conflict Resolution** This core step involves analyzing the data to resolve conflicts. Data is preprocessed and converted into individual claims. A truth discovery algorithm, such as Latent Credibility Analysis (LCA), is then used to assess the credibility of these claims and the trustworthiness of the data sources. This helps to resolve contradictions when different sources provide conflicting information [1].
10. **Store Analysis Results** The output of the analysis is a set of unique, non-conflicting claims about the IT architecture. These results are written back to the database for future use and troubleshooting [1].
11. **Generate Model File** In the final step, a file is created from the analyzed data. This file can be imported directly into the chosen modeling tools, such as Archi or securiCAD, to produce the final IT architecture model [1].

**describe this 11 step process in presentation worthy words for a 4 min presentation including a demo that we created. use appropriate keywords which are unique selling points.**

Good morning. We all know that manually creating and updating IT architecture models is a slow, complex, and often inaccurate process. By the time a model is finished, the real-world architecture has already changed, leaving you to make critical decisions with outdated or conflicting information.

**The Challenge: A Maze of Conflicting Data**

Today’s IT environments are a complex web of systems. To get a complete picture, you must pull data from everywhere: network scanners, configuration files, vulnerability reports, and more. The problem is that these sources often disagree. One scanner might report a server is running Windows 7, while another claims it's running Windows XP. Which one do you trust? Manually reconciling this is a tedious process prone to human error.

**Our Solution: An Intelligent Framework for Automated Modeling**

We have developed a groundbreaking framework that automates this entire process, delivering accurate, up-to-date models on demand. Our solution is built on a powerful and unique approach to data integration and analysis.

First, we perform **Holistic Data Aggregation**. Our system connects to all of your heterogeneous data sources to create the most comprehensive view of your architecture possible [[1]](#fn1). This raw data is then standardized into a **Common Language**, ensuring all information is comparable and ready for analysis [[1]](#fn1).

**The Secret Sauce: Our "Truth Discovery" Engine**

Here is our unique selling point: a **Truth Discovery Engine**. We don’t just merge data; we intelligently resolve conflicts. Our framework uses a sophisticated probabilistic algorithm that analyzes every piece of conflicting information. It assesses the historical reliability of each data source to determine the most credible "truth" [[1]](#fn1). This provides a single, trustworthy source of truth, eliminating guesswork and manual fact-checking.

**The 3-Phase Process: From Raw Data to Actionable Models**

Our streamlined process makes this possible in three simple phases:

* **Phase 1: Foundation** You begin by defining what you want to model—whether it's an enterprise view in ArchiMate, a threat model, or a fault-tree analysis. You simply point the framework to your various data sources [[1]](#fn1).
* **Phase 2: Intelligent Integration** Next, the automation kicks in. The framework automatically pulls the data, transforms it into our common language, and prepares it for the core analysis [[1]](#fn1).
* **Phase 3: The Payoff** Finally, our Truth Discovery engine analyzes the data, resolves all contradictions, and generates a final, validated model file. This file can be imported directly into your preferred modeling tool [[1]](#fn1).

**Why This Matters: The Unique Benefits**

This approach offers several key advantages:

* **Extensible and Tool-Independent** You can use any data source to create any type of model you need. Our framework is not locked into a specific tool or metamodel, offering maximum flexibility [[1]](#fn1).
* **Probabilistic Accuracy** We move beyond simple data aggregation to provide a model with the highest probability of being correct, filtering out disputed and low-credibility information [[1]](#fn1).
* **Unmatched Speed and Agility** This framework reduces modeling efforts from weeks to mere minutes. Your architecture models are no longer static snapshots; they become living representations of your environment, enabling truly agile and informed decision-making.

Now, to show you just how powerful this is, we’ve prepared a short demo. We applied our framework to a complex SCADA system environment, using eleven different data sources to automatically generate three distinct architectural models. Let's take a look.

⁂

1. Article\_20\_2-2.pdf