Probabilistic Data Management for Pervasive Computing: The *Data Furnace* Project

Minos Garofalakis[†], Kurt P. Brown[†], Michael J. Franklin[*], Joseph M. Hellerstein[*], Daisy Zhe Wang[*]
Eirinaios Michelakis[*], Liviu Tancau[*], Eugene Wu[*], Shawn R. Jeffery[*], Ryan Aipperspach[*]

[†]Intel Research Berkeley and [*]University of California, Berkeley

Abstract

The wide deployment of wireless sensor and RFID (Radio Frequency IDentification) devices is one of the key enablers for next-generation pervasive computing applications, including large-scale environmental monitoring and control, context-aware computing, and "smart digital homes". Sensory readings are inherently unreliable and typically exhibit strong temporal and spatial correlations (within and across different sensing devices); effective reasoning over such unreliable streams introduces a host of new data management challenges. The Data Furnace *project at Intel Research and UC-Berkeley aims to build a probabilistic data management infrastructure for pervasive computing environments that handles the uncertain nature of such data as a first-class citizen through a principled framework grounded in probabilistic models and inference techniques.*

1 Introduction

Pervasive Computing is an area that has seen significant interest since it was first identified by the late Mark Weiser over a decade ago, with research contributions from a breadth of computer science disciplines including distributed and mobile systems, machine learning, and human-computer interaction. The broad challenge in pervasive computing is the creation of environments that embed computation and communication in a way that organically interacts with humans to enhance or ease their daily behavior. One typical scenario for pervasive computing is in the design of "smart digital homes", which are instrumented to observe, learn, and facilitate the typical behaviors of occupants. As one concrete example, smart homes can automate control of utilities like lighting, heating and cooling, with the goal of minimizing energy usage without adversely impacting occupant comfort.

Recent advances in distributed sensing and wireless communication enable pervasive applications that are quite information-rich, capturing and utilizing large numbers of potentially high-bandwidth streams of data. This raises a number of interesting research opportunities at the nexus of database management, stream query processing, sensor networks, machine learning, and human-computer interaction. We highlight some of the challenges of data-intensive pervasive computing in what follows.

• *Diverse Data Sources.* A wide variety of sensors can be reasonably deployed in pervasive applications, from richly semantic, high-bandwidth sensors like video cameras, to simple boolean sensors like door-ajar switches, with a variety of sensing modalities in between (audio, motion, temperature, etc.).

Bulletin of the IEEE Computer Society Technical Committee on Data Engineering

• *The Realities of Sensory Data.* Data from sensors is different from what is typically seen in enterprise databases. First, sensor data is typically a very noisy, *uncertain* representation of the phenomena it is intended to capture, due to issues like miscalibration and sensitivity to environmental factors. This can make sensor data hard to "trust". On the other hand, many physical phenomena represented by sensor data can exhibit significant spatial and temporal correlation, and this structure in the data can be exploited both for efficiency and for better understanding the true properties of the underlying physical phenomena.
• *Streams and Storage.* Pervasive applications typically have a "real-time" nature, since they are intended to interact with human behaviors. On the other hand, in order for pervasive applications to build good temporal models of human behavior and use those models for prediction, they may need to track significant volumes of archival data. Hence, the underlying data-management infrastructure needs to support both rich, continuous queries/triggers over streams, as well as offline mining/learning of useful patterns from archival data.
• *Integration of Probabilities and Logic.* Traditional interfaces and languages for data management have strong roots in logic. By contrast, pervasive applications that deal with sensory data most often operate with probabilistic methods, which must reason about uncertainty for a variety of reasons: to model the true phenomena generating the sensor readings, to model future states of the environment, to predict human desires or behaviors, and so on. As is well known in the database literature, data-intensive applications perform better by pushing computation into data access, rather than extracting data and shipping it to computations. Hence, the uncertain, probabilistic nature of the data needs to be a first-class citizen across all layers of the data-management system: the data model (both logical and physical), the query processing techniques, triggering engines, pattern detection, etc.
• *Complex Modeling.* There is often a significant semantic gap between sensed data (e.g., a sequence of plumbing usage, power consumption, and certain sound signatures) and meaningful human activities (e.g., "making a cup of tea"). This gap needs to be bridged by a modeling framework that can capture specifications of high-level "complex" events in terms of lower-level sensed events. These specifications may be generated manually, automatically, or semi-automatically, but in any case there needs to be a modeling framework to capture them. Moreover, this modeling framework has to interact meaningfully with the uncertainty and correlations present in the base sensory data, as well as with other, more conventional data sources.

The *Data Furnace*. In our work at Intel Research and UC Berkeley, we are embarking on an effort to develop information-rich infrastructure for pervasive computing, suitable to applications like the Digital Home. In that spirit, we refer to our system as a *Data Furnace*: a low-maintenance, low-profile, but critical piece of infrastructure that provides general-purpose data management infrastructure for pervasive computing applications. Briefly, the *Data Furnace* aims to manage data uncertainty as a first-class citizen through a principled probabilistic framework, and provide a uniform, declarative means for higher-level applications to store, query, and learn from such probabilistic data. In this paper, we outline some of the unique research challenges that arise in the context of the *Data Furnace*, and discuss some of the basic ideas underlying our approach.

2 A Motivating Application Scenario: The "Smart Home"

Harper [9] defines the term "smart home" as "a residence equipped with computing and information technology which anticipates and responds to the needs of the occupants, working to promote their comfort, convenience, security and entertainment through the management of technology within the home and connections to the world beyond". The recent revolutions in personal digital media and sensing technologies has rendered such "futuristic" visions much more credible, and the home is currently one of the most frequently targeted markets for these new technologies.

Data management plays a critical role in the smart-home vision. Overwhelming amounts of data float around in our home today, such as music, movies, contact lists, calendars, photos, financial data, news articles, web pages, e-mail messages, and so on. The smart home of the future will only exacerbate the situation with the addition of many more devices and systems, including hundreds of sensors (of different modalities) and RFID readers/tags for sensing several different physical phenomena and activities. Through such sensory (input) devices as well as additional control (output) mechanisms, smart homes will be able to support *richer, real-time interactions* with their users; for instance, energy management and demand-response applications (`http://dr.berkeley.edu`) can employ motion sensors to track context information for Alice and appropriately actuate electric devices like lighting and water heating. A different application might require correlating sensory readings and actuation with more conventional information sources in the home or even historical patterns of user behavior; for example, a security subsystem sensing motion patterns and activities that do not match any known historical patterns for the home users, might choose to take some precautionary action (e.g., notify Bob or start video-recording the "suspect" individual).

Allowing future smart homes to support such rich user-interaction models over streaming, uncertain sensor data (of various modalities) raises a number of novel data-management challenges. Providing persistent storage and querying (that, of course, satisfies all other traditional database requirements of availability, consistency, security/privacy, etc.) for diverse, heterogeneous data-types has received attention recently through different research projects, such as MyLifeBits [8]. Still, as our discussion above indicates, this is only part of the equation: data-management infrastructures for the smart home will also have to effectively combine and correlate large streams of low-level, uncertain sensory data (possibly, in real time) to accurately track and react to higher-level events and activities. We use motivating examples from the smart-home setting in the remainder of this paper.

3 The *Data Furnace* System: Architecture and Challenges

We envision our *Data Furnace* system as the centerpiece of the data-centric architecture for pervasive applications. The *Data Furnace* serves as the central repository for application data and metadata, and offers a number of diverse services at different layers of abstraction (ranging from device and web connectivity, to data archiving, to pattern learning and probabilistic reasoning). The high-level logical architecture of the *Data Furnace* is depicted in Figure 1. In a nutshell, the *Data Furnace* architecture comprises three broad layers: (1) The *Hardware Layer* manages physical system resources, such as processing, storage, and communication. (2) The *Metadata Layer* serves as the repository for environment metadata, including, for instance, "schema" information for the application context (e.g., home architecture, floorplans, wiring and pipe layouts), data on the users and (possibly) their routines, as well as information on devices, events, and API definitions; this layer defines the basic *Data Furnace* interface with the physical world and higher-level applications. (3) The *Service Layer* is the heart of the *Data Furnace* engine, essentially providing the key information-management functionality for our target application scenarios, including query processing and optimization, data archiving, complex event processing, pattern and model learning, probabilistic reasoning, and so on.

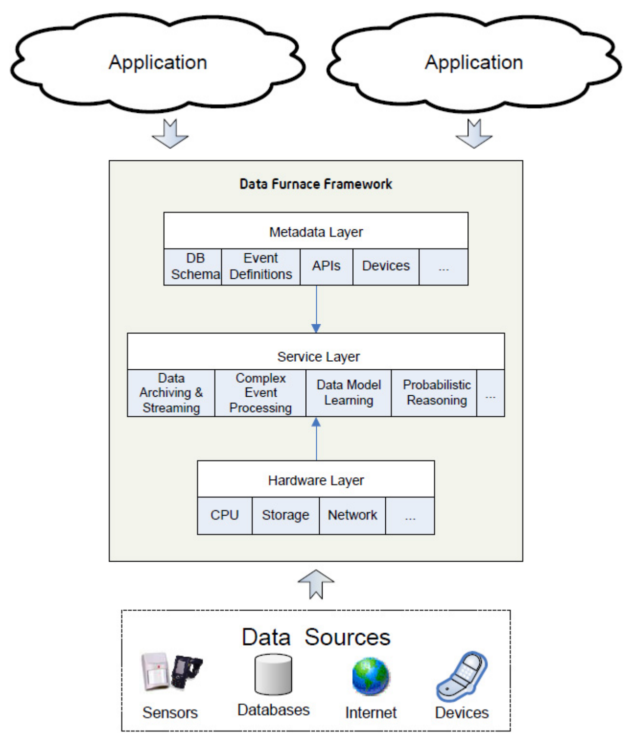

Figure 1: The *Data Furnace* Architecture.

Given the inherently uncertain nature of several of its key data sources, as well as the complexity of human behavioral patterns and automated recognition of higher-level activities, it is clear that uncertain data and probabilistic information will play a central role in our *Data Furnace* system. We now outline some of the key probabilistic data management challenges for the *Data Furnace* engine, and briefly discuss some of the basic ideas underlying our approach. (Pervasive computing environments such as the smart home also raise several other challenging database and system-design issues with respect to multi-device integration, availability, privacy, self-management, longevity, unobtrusiveness/invisibility, and so on; while acknowledging their importance, we do not focus on these concerns in this short paper.)

Voluminous Streams of Uncertain, Low-Level, Correlated Sensory Data. Raw readings from sensing and RFID devices are inherently unreliable and subject to various physical-world phenomena, such as battery-power fluctuations, noise, and interference. As a result, the streams of readings emanating from such devices are *uncertain* (to various degrees), with several missing and/or inaccurate values. For instance, the observed read rates (i.e., percentage of tags in a reader's vicinity that are actually reported) in real-world RFID deployments is often in the $60-70\%$ range [10, 12] (i.e., over 30% of the tag readings are routinely dropped); even higher drop rates are possible depending on environmental characteristics (e.g., in the presence of metal [5]).

Sensor and RFID readings also tend to exhibit strong *correlation patterns*, a fact that has already been exploited in the area of acquisitional query processing in sensornet deployments [1, 4]. Such correlation patterns include both (1) *spatial correlations*, where sensing devices in close physical proximity capture the same physical event or highly-correlated readings (for instance, temperature and light readings in the same room); and, (2) *temporal correlations*, where correlated sequences of events are often sensed together (for instance, turning on the stove is typically followed by turning on the hood fan). *Causality* relationships are also quite common in sensor data streams; for example, detecting a person at the door can be causally connected to the event "Bob returning home from work".

Due to their highly unreliable nature, raw, low-level readings streams are often useless for the purposes of providing useful knowledge to higher-level applications (e.g., accurate inventory or people tracking). Instead, such applications are interested in more concise, *semantically-rich events* that can be *inferred* (with some level of *probabilistic confidence*) from the base sensing data. This implies that a probabilistic data-management framework is needed in the *Data Furnace* engine. And, of course, capturing the (possibly, complex) correlation and causality patterns in such data is a crucial requirement for both efficient and accurate probabilistic inference — base-tuple independence assumptions typically employed in earlier work on probabilistic database systems (e.g., [2, 3]) are rarely valid in our setting and almost certainly will lead to poor probabilistic estimates for higher-level events. Continuing with our earlier example, it is clear that the conditional probability $\mathbf{Pr}\,[\text{Bob returning from work}|\text{person at the door}]$ is likely to be very different from $\mathbf{Pr}\,[\text{Bob returning from work}]$ (while the two are equal assuming independence); similarly, probabilistic computations across, say, sensors in the same room, also need to account for such strong correlations. Thus, effectively capturing probabilistic (base and derived) data correlations is an important requirement for the *Data Furnace* engine.

Our basic approach here is to accurately model both the probabilistic nature of the sensor data and the underlying correlation/causality patterns by incorporating *statistical learning techniques* and *probabilistic models* [11, 13, 14] as first-class citizens in the *Data Furnace* engine. The *Data Furnace* can learn such models either incrementally (over the input data streams) or off-line (from archived data), and employ probabilistic-inference methods to efficiently query these models at run-time. Model learning is a computationally-intensive process [11], so maintaining our probabilistic models over continuous streams from possibly hundreds of sensory streams poses difficult technical challenges; we intend to exploit semantic knowledge (such as floorplan and sensor-layout information) to enable scalable incremental learning.

Definition and Real-Time Tracking of Complex, Hierarchical Probabilistic Events. Effective real-time monitoring and management of the target pervasive-computing environment (e.g., smart home or supply chain) is an important requirement, and mechanisms for composing and tracking complex events (in real time) can address this need. Such events typically require correlating (e.g., through joins or aggregation) multiple uncertain sensing streams under intricate notions of time, space, ordering, and Boolean connections.

In addition to the uncertain nature of *Data Furnace*'s sensor inputs, in most real-life scenarios, meaningful definitions of high-level, semantically-rich events over sensor readings are inherently *probabilistic* in nature. For instance, a "making tea" event might be connected with "stove usage" 70% of the time and "microwave usage" 30% of the time. In general, such events can be thought of as *"occurrences of significance"* defined in a *hierarchical manner* using appropriate (probabilistic) *rules* for composing several streams of real-time sensing data, and perhaps other (lower-level) events. Detecting such an occurrence (with some appropriate level of confidence) can, in turn, trigger higher-level events or appropriate actions. As an example, detecting Bob in the living room and room-temperature readings below 60^oF with confidence, say, above 95% can lead to the system automatically turning on the heat in the living room; similarly, stationary readings from Alice in the TV room over a window of time combined with a "TV on" event can trigger a higher-level event "Alice is watching TV", with some level of confidence.

Abstractly, at any point in time, the state of the system can be seen as a *probability distribution* over possible states and high-level events, and the *Data Furnace* engine needs to support effective mechanisms for (1) defining hierarchies of complex probabilistic events that are of interest to users/applications, and (2) accurately tracking such events (and corresponding confidences) in real time over numerous sensor and RFID streams. Obviously, accurate probabilistic estimations are needed to avoid erroneous inferences that can potentially result in event false positives/negatives and improper actions; for instance, the environment might overreact in response to low-confidence sensed user activity, thus defeating the purpose of *calm computing*. And, again, principled techniques for modeling existing correlation/causality patterns within and across base data and events are needed for correct probabilistic reasoning. Semantically-rich probabilistic events also provide a fairly clean, natural abstraction for higher-level applications that wish to employ *Data Furnace* services (Figure 1), without worrying about the details and uncertainties of raw sensor readings. Our initial design of the *Probabilistic Complex Event Triggering (PCET)* subsystem (Section 4) addresses several of these challenges.

Efficient Querying and Learning over both Probabilistic and Deterministic Information. Uncertain and probabilistic information needs to co-exist in the *Data Furnace* engine with more traditional (relational and non-relational) data. Event and query processing techniques must be able to effectively reason with both types of data and provide reasonable performance to higher-level applications. For instance, both complex event tracking and ad-hoc queries over streams from smart-home sensors and RFID readers may require direct correlation with home metadata, such as floorplans and users' daily schedules. In addition to continuous monitoring and ad-hoc query processing, it is also important to effectively *mine* the information collected from both probabilistic and conventional data sources for longer-term trends and patterns at different time and/or space granularities; for example, learning users' daily or weekly routines is critical for effective energy management or detecting "suspicious" behavior in the smart home.

Effective querying and mining of uncertain, probabilistic data (perhaps in combination with other, deterministic information) raises a host of new challenges for the *Data Furnace*. Efficiency, in particular, is an important concern, given the recent negative results of Dalvi and Suciu for general query processing over probabilistic data tuples [2]. We believe that, through the effective use of rich probabilistic models (which can, in a sense, be seen as concise approximations of the *possible-worlds distribution* [2, 3]), the *Data Furnace* query processor can avoid such inherently intractable problems. The *granularity of the probabilistic information* is a key parameter here — while earlier probabilistic-database work that focuses on integration and/or lineage problems [2, 3, 15] connects probabilities with individual tuples or even attribute values, it is not clear that such fine-grain information is needed in our settings. For instance, it may be possible to associate probabilistic noise or dropped-readings models for individual sensing devices, which can essentially be tied to all readings (tuples) from a given physical device. This leads to more concise probabilistic representations and, hence, more efficient (albeit, approximate) techniques for probabilistic query processing and event tracking through model inference. Of course, for sizeable data collections and complex probabilistic models incorporating notions of time and space, the *Data Furnace* needs to support efficient methods for *approximate probabilistic inference* (e.g., particle filtering [14]) and *approximate query processing* (e.g., histograms, wavelets, sketches [6, 7]). Designing appropriate *query and data-definition languages* for such rich, diverse data collections also raises several interesting research challenges.

System Support for Managing, Maintaining, and Reasoning over a Multitude of Probabilistic Models. Managing large collections of probabilistic models (perhaps built for different inferencing tasks) is a crucial requirement for our *Data Furnace* architecture. Supporting a variety of such models as first-class citizens brings up a number of interesting systems issues, including appropriate declarative interfaces and data structures (e.g., indexes) for probabilistic-model maintenance and inference (querying). *Data Furnace* models can also be viewed as *probabilistic views* built over the base data, and can be supported (as either virtual or materialized structures) to model the state of the pervasive-computing environment at potentially different levels of abstraction (e.g., at the sensor-readings layer or at the user-activities layer). As with traditional view materialization, the key tradeoff lies in the view maintenance cost vs. its query-processing benefits; of course, the probabilistic nature of our model views makes the problem somewhat unique. Similarly, multiple inferencing tasks over probabilistic models can possibly share processing costs, leading to interesting (probabilistic) multi-query optimization issues. This is only a small sample of the interesting optimization questions arising in the query/event-processing engine of the *Data Furnace*.

4 The Probabilistic Complex Event Triggering (*PCET*) Subsystem

We are currently building a first incarnation of the *PCET* subsystem of the *Data Furnace*, which aims to provide a general probabilistic event triggering service over uncertain sensory data. In this section, we briefly touch upon some of the key elements of the *PCET* logical architecture (Figure 2).

The Probabilistic Inference Engine (PIE) essentially employs statistical learning techniques to build, maintain, and run inferences over probabilistic models that capture correlations across base-level events (i.e., individual sensor readings), as discussed in Section 3. For our initial implementation, we plan to use parametric (e.g., Gaussian) models to model sensor noise and a variant of *dynamic Bayesian networks* [14] as our basic probabilistic model; we also plan to explore ways of capturing and exploiting *spatial information* (e.g., home floorplans) in *PCET*'s learning and modeling tools. *PCET*'s Application Layer Interface employs *probabilistic events*, perhaps with associated *confidence thresholds* (e.g., Bob is in the living room with probability $\geq 90\%$), as the key abstraction for communicating with higher-level applications. Through this simple interface, applications can hierarchically define and subscribe to new complex events based on either base-level events (e.g., sensor readings exceeding a certain value) or other event-tracking "services" already offered within or on top of *PCET*. For instance, basic smart-home services, such as a "People Tracker" that tracks the location of individual users in the home can be provided as part of the *Data Furnace* distribution with an easy-to-use customization GUI; other, higher-level applications like a "Personalized Environment Controller" can be built using both base sensor readings and "People Tracker" events.

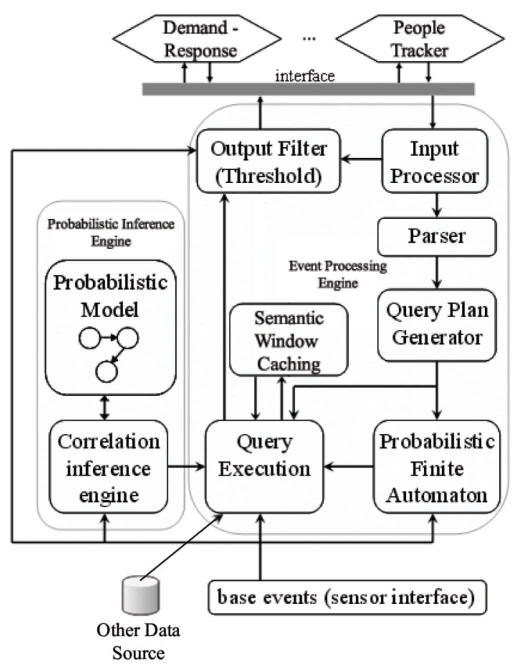

Figure 2: The *PCET* Subsystem.

PCET supports the definition of new complex events through a small, yet expressive composite-event algebra that includes: sequencing ($e_1; e_2$), conjunction ($e_1 \& e_2$), disjunction ($e_1 | e_2$), negation ($!e$), and temporal ($\{e\}t$) as well as location ($e@loc$) constraints. Of course, events at different levels of abstraction can be defined by referencing previously-defined events, as shown in the following simple example:

```
LightOn := LightSensor1='ON'
LightSwitchedOn := !LightOn ; LightOn
PersonEntry := {Motion@Outside ; DoorOpen ; DoorClose ; Motion@Hall}10s
Cooking := {StoveOn | MicrowaveOn}10m
```

PCET's Event Processing Engine (EPE) is responsible for (1) compiling event definitions into self-contained execution plans that are continuously run over the input data streams; and, (2) possibly tracking the current "state" of the environment (based on observed events) using what we term a Probabilistic Finite Automaton representation. Naturally, EPE employs the probabilistic inference services provided by PIE to ensure that probabilistic beliefs are correctly propagated through the composite-event definitions.

5 Conclusions

We have discussed some of the unique probabilistic data management challenges arising in pervasive-computing environments, and provided a quick overview of our proposed approach for the *Data Furnace* system, which aims to provide a general data-management infrastructure for pervasive applications. Our current research agenda includes building a first prototype of the probabilistic-event tracking subsystem and exploring different alternatives and algorithms for capturing and querying probabilistic information in the *Data Furnace*.

References

[1] David Chu, Amol Deshpande, Joseph M. Hellerstein, and Wei Hong. "Approximate Data Collection in Sensor Networks using Probabilistic Models". In *ICDE*, 2006.

[2] Nilesh Dalvi and Dan Suciu. "Efficient Query Evaluation on Probabilistic Databases". In *VLDB*, 2004.

[3] Nilesh Dalvi and Dan Suciu. "Answering Queries from Statistics and Probabilistic Views". In *VLDB*, 2005.

[4] Amol Deshpande, Carlos Guestrin, Samuel R. Madden, Joseph M. Hellerstein, and Wei Hong. "Model-Driven Data Acquisition in Sensor Networks". In *VLDB*, 2004.

[5] Kenneth P. Fishkin, Bing Jiang, Matthai Philipose, and Sumit Roy. "I sense a disturbance in the force: Unobtrusive detection of interactions with RFID-tagged objects". In *UbiComp*, 2004.

[6] Minos Garofalakis, Johannes Gehrke, and Rajeev Rastogi. "Querying and Mining Data Streams: You Only Get One Look". Tutorial in *VLDB*, 2002.

[7] Minos Garofalakis and Phillip B. Gibbons. "Approximate Query Processing: Taming the Terabytes". Tutorial in *VLDB*, 2001.

[8] Jim Gemmel, Gordon Bell, and Roger Lueder. "MyLifeBits: A Personal Database for Everything". *Comm. of the ACM*, 49(1), 2006.

[9] Richard Harper. *"Inside the Smart Home"*. Springer, 2000.

[10] Shawn R. Jeffery, Gustavo Alonso, Michael J. Franklin, Wei Hong, and Jennifer Widom. "A Pipelined Framework for Online Cleaning of Sensor Data Streams". In *ICDE*, 2006.

[11] Michael I. Jordan. *"An Introduction to Probabilistic Graphical Models"*. (Book, in preparation), 2006.

[12] Laurie Sullivan. "RFID Implementation Challenges Persist, All This Time Later". *Information Week*, October 2005.

[13] Judea Pearl. *"Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference"*. Morgan Kaufmann Publishers, 1988.

[14] Sumit Sanghai, Pedro Domingos, and Daniel Weld. "Relational Dynamic Bayesian Networks". *Journal of AI Research*, 24:1–39, 2005.

[15] Jennifer Widom. "Trio: A System for Integrated Management of Data, Accuracy, and Lineage". In *CIDR*, 2005.