Visionary Paper

Self-Curating Databases

Mohammad Sadoghi[#], Kavitha Srinivas[#], Oktie Hassanzadeh[#],
Yuan-Chi Chang[#], Mustafa Canim[#], Achille Fokoue[#], Yishai Feldman[*]

[#]IBM T.J. Watson Research Center
[*]IBM Research - Haifa

ABSTRACT

The success of relational databases is due in part to the simplicity of the tabular representation of data, the clear separation of the physical and logical view of data, and the simple representation of the logical view (meta-data) as a flat schema. But we are now witnessing a paradigm shift owing to the explosion of data volume, variety and veracity, and as a result, there is a real need to knit together data that is naturally heterogeneous, but deeply interconnected. To be useful in this world, we argue that today's tabular *data model* must evolve into a *holistic data model* that views meta-data as a new semantically rich source of data and unifies data and meta-data such that the data becomes *descriptive*. Furthermore, given the dynamicity of data, we argue that fundamental changes are needed in how data is consolidated continuously under uncertainty to make the data model naturally more *adaptive*. We further envision that the entire *query model* must evolve into a *context-aware model* in order to automatically discover, explore, and correlate data across many independent sources in real-time within the context of each query. We argue that enriching data with semantics and exploiting the context of the query are the two key prerequisites for building *self-curating databases* in order to achieve a *real-time exploration and fusion of enriched data at web scale*. These needs highlight a series of interesting challenges for database research and alter some of the tenets of Codd's rules for how we think about data.

1. INTRODUCTION

We believe that the relational database system will remain the de facto standard for well-structured data. The success of relational database theory is partly due to its simple tabular representation over a predefined relational schema. Tabular data is manipulated through a well-defined declarative relational algebra that is written over the data schema (a logical view). Expressing queries over the logical view has led to decades of query optimization in order to transform queries written over logical views into efficient access methods over the physical layout. As database engines advance, the logical view remains constant, and this has been a key success factor for relational database systems (i.e., data independence).

However, the tabular data represented by the relational schema is limited to a flat schema for describing each table column.[1] One may argue that the relational schema, in addition to forming a logical view for querying the data, is nothing but a simple blueprint of how to parse the data at the physical level. This blueprint ends at the granularity of columns, which is why often it is referred to as a table schema because it is not at the record level. Homogeneity at the record level is also pre-assumed in the relational theory, in fact, the Boyce-Codd normal forms to some extent already penalize any column heterogeneity [6]. Similarly, NoSQL databases such as key-value stores are still fundamentally tabular, but the "value" column is now heterogeneous with a flexible schema [4].[2]

Although databases were designed for a system of records in order to maintain corporate transactional data, the tabular data model in databases can represent many types of non-transactional data. However, it has certain fundamental limitations. The chief limitation is that the tabular model does not natively capture instance-level relations, which is why a whole class of functional dependency (FD) and referential integrity (RI) constraints had to be developed to express schema-level relationships (e.g., RI) and to avoid record-level inconsistencies (e.g., FDs). In general, integrity constraints are used to ensure that data instances conform to a given schema while only limited knowledge (e.g., relation transitivity) can be expressed using constraints because the primary role of constraints is to restrict the data as opposed to enriching the data [12].

We observe that today's data is no longer limited to systems of records; we now have a variety of data coming from thousands of sources. Data is being generated at an astonishing rate of 2.5 billion gigabytes daily, and further, 80% of data is unstructured and comes in the form of images, video, and audio data to social media (e.g., Twitter, Facebook, Blogsphere) and from embedded sensors and distributed devices [1]. The explosion is partly due to the Internet of Things (IoT) that increasingly connects data sources (including objects and devices) to form a complex network, a network that is expected to exceed one trillion nodes [1].

These emerging data sources are heterogeneous by nature and are independently produced and maintained, yet the data are inherently related. For example, sales patterns correlate with the popularity of the product in social media, and the popularity of the product itself can be measured in terms of how often images or tweets are posted of the product. Even if one considers only the "structured" data after the extraction from the unstructured data, the task of integrating all these disparate data sources leaves islands of data with thousands (if not millions) of tables and schemata that are simply impossible to understand and query by any individual.

Arguably data is a new natural resource in the enterprise world with an unprecedented degree of proliferation and heterogeneity and countless possible ways of aggregating and consuming it to find

[1] Although the database schema has remained a simple flat structure, there have been attempts to at least model the data conceptually as a hierarchy, e.g., the entity-relation (ER) or object-oriented models.

[2] In fact, several NoSQL initiatives even motivate the need for a *schema-less* paradigm [4] that is in a diametrically opposite position from our self-curating database vision.

actionable insights [1]. However, this inherently interconnected data is trapped in disconnected islands of information, which forces analytics-driven decision making to be carried out in isolation and on stale (and possibility irrelevant) data; thus, making today's *first-ingest-then-process* model insufficient and unnecessary at a time when the cloud is disrupting the entire computing landscape.[3] More importantly, existing database technologies fail to alleviate the data exploration challenges that continue to be a daunting process especially at a time when an army of data scientists are forced to manually and continuously refine their analyses as they sift through these islands of disconnected data sources, a labor-intensive task occupying 50-80% of time spent [11].

We argue that today's database systems need to be fundamentally re-designed to capture data heterogeneity (within local and external data sources) and the semantic relationship among data instances (i.e., data interconnectedness) as first class-citizens. To address these requirements, we propose a *holistic data model* to capture all dimensions of the data, so that we can push the burden of semantic enrichment and integration of the data in a systematic and transparent way into the database engines. We view the data fusion as a *gradual curation* process that transforms the raw data into a new unified entity that has *knowledge-like characteristics*; thereby, we envision the evolution of database systems into *self-curating databases* to meet the continuous enrichment and integration challenges of the information explosion.

In moving from databases to self-curating systems, the schema is no longer a table schema as a separate entity that is limited to necessary information to only parse the data. Instead, the data schema becomes part of the data in order to make the data self-descriptive. Furthermore, it is expected that both the data and meta-data continuously evolve either by ingesting new data sources or through the process of *context-aware query* execution. By context-aware query execution, we emphasize redefining the existing *query model* to enable the discovery of new data linkage and semantic relationships in the specific context of a given query. Thus, while the query must automatically be refined to enable discovery, the data will also become sufficiently enriched in order to enable continuous integration and adjustment of the interconnectedness of instances/types.

In short, our broader vision is a systematic methodology to achieve a *real-time fusion and enrichment of data at web scale* hosted virtually. We argue for a unified view of semantically enriched data by introducing a novel *holistic data model* (i.e., *rethinking the data model*), so queries can be answered by an online consolidation of the most up-to-date data from a variety of sources at query time without the need for offline ingestion and curation. Furthermore, we envision that querying and analytics in general will become explorative in nature to provide deeper and quicker insights by proactively refining and raising new queries based on the context of the query submitted by the user, making the query model *context-aware* (i.e., *rethinking the query model*).

Thus far, we have provided the desired properties of self-curating databases. In the subsequent sections, we elaborate on the specific properties of self-curating databases and highlight the challenges and short- and long-term research opportunities they bring in a systematic fashion. We further broadly classify our proposed statements as either *functional statements* (for adding new capabilities) or *optimization statements* (for improving system performance); these open problems are summarized in Table 1.

2. RELATED WORK

Our vision is partly motivated by the recent shift towards semantically enriched information retrieval. We observe a trend among Web search engines such as Google and Bing in moving away from a pure information retrieval system towards knowledge-based retrieval by not only retrieving a set of documents relevant to the users' queries, but also identifying *entities* and returning *facts* regarding the identified entities [13]. Another prominent initiative is IBM *Watson*, which is an open-domain question answering system for outperforming the best players in Jeopardy [7]. Such knowledge-based retrieval has become possible through the use of rich knowledge bases created by academic and community efforts such as Freebase, DBpedia, and YAGO [13].

What we observe in all these emerging projects [13, 7] is that moving forward, simple information retrieval will be insufficient, and that information will continuously be expanded and semantically enriched as a result of the continuous integration of heterogeneous sources (i.e., the evolution of information to knowledge). Consequently, we argue the need for the evolution of relational databases to handle the challenges in this new enriched data era. We envision that the database systems of the future will no longer be solely responsible for the storage and retrieval of structured data, but they will transform into self-curating databases that are capable of *real-time exploration and fusion of enriched data at web scale*.

We acknowledge that we are not the first to argue for the high-level concepts such as semantic enrichment or continuous integration; in fact, there are several existing efforts in Semantic Web technologies (e.g., [13]) and dataspaces and pay-as-you-go integration models (e.g., [8]) that strive for similar high-level objectives. But to achieve these objectives, we argue for a fundamental rethinking of how we view the data and query. We envision the need for a new *holistic data model* to unify the data and meta-data, and to view the meta-data as a new semantically rich source of data. Furthermore, we envision a simpler and more effective way to query and compute answers by automatically refining the query and continuously discovering new data sources within the context of each query, giving rise to a novel *context-aware query model*. Further, what is unique to our vision, in addition to extending past attempts in light of new applications and possibilities (e.g., [7]), is systematically sketching the requisite properties of a self-curating database and providing an extensive list of the concrete research challenges and opportunities needed to make such a vision a reality.

Statement	Description
FS.1	Continuous incremental entity resolution
FS.2	Formalism for assessing interconnectedness richness
FS.3	All-encompassing logical formalism for uncertainty
FS.4	Simplifying logical view of data
FS.5	A Unified language for relational, logical & numerical models
FS.6	Context-aware query refinement semantics
FS.7	Query refinement using query-by-example
FS.8	Incompleteness resolution through crowd
FS.9	Context-aware materialization of ranked & discovered data
FS.10	Parallel world semantics and representational model
FS.11	Concurrency controls for non-deterministic and enriched data model
OS.1	Fine-grained dynamic data clustering
OS.2	Locality-aware multi-hop traversal representation
OS.3	Semantic query optimization
OS.4	Data placement in distributed shared memory

Table 1: Open problems in self-curating databases

3. DATA MODEL: UNIFIED & ENRICHED

In our view, a self-curating database must have a hierarchy of layers to transform raw data incrementally into *a holistic data model* (depicted in Figure 1). First is the *instance layer*, to store the raw data (or data instances) spanning both structured and unstructured. The second layer is the *relation layer*, a horizontal expansion of data to formulate and capture the interconnectedness of data instances within and across data sources (i.e., the fine-grained instance-level linkage). In cases where the raw data layer is unstructured, this layer may additionally capture the results of information extraction. The third layer is the *semantic layer*, a vertical expansion of data to conceptualize data instances and their rela-

[3] We anticipate that in the near future all data sources and analytics computation will be hosted virtually on the cloud [1]; thus, there is no need to first ingest data from one computing infrastructure to another before querying the data.

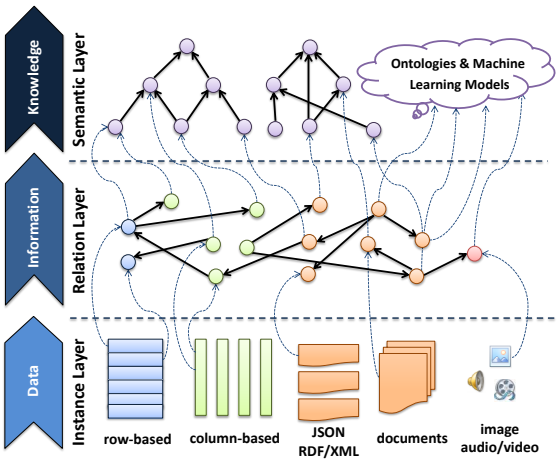

Figure 1: Holistic data model.

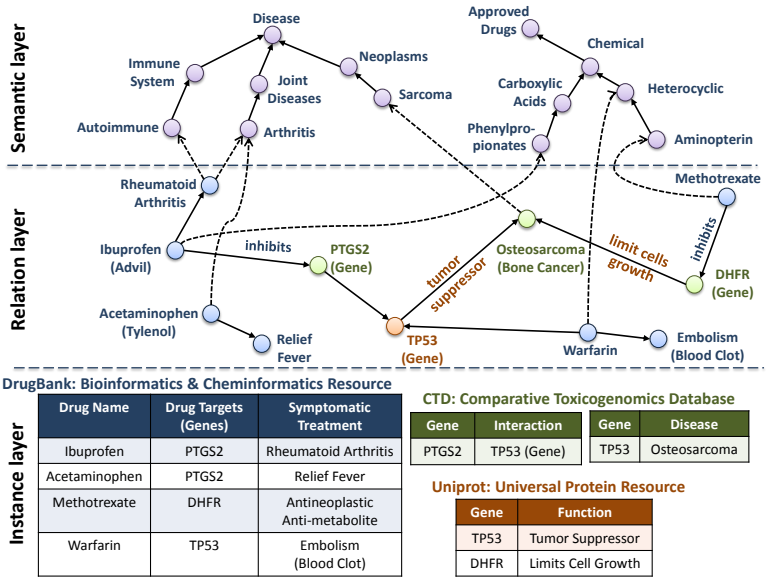

Figure 2: An example of an enriched data model in the life science domain.

tions as semantic types and to formulate the interconnectedness of archetypes and data instantiated types (e.g., ontology). The semantic layer is a way of succinctly capturing conceptual relationships among data instances. This final layer will bring an unprecedented level of expressiveness power and discovery potentials. The last two layers of self-curating databases can be viewed as meta-data, but such a distinction no longer holds in a self-curating database as meta-data is also seen as a rich source of data.

We argue that such a holistic approach is essential to efficiently represent, enrich, manipulate, and query both data and meta-data. Our running example of an enriched data model is extracted from the life science domain, as illustrated in Figure 2. This example is motivated by the overwhelming challenge to unify and enrich data from a variety of heterogeneous sources to develop an assisted diagnosis and personalized treatment and medicine [7].

3.1 Instance Layer: Raw Data

The first layer is what today's relational database systems heavily rely on to represent structured data. But future databases must naively also support semi-structured data such as XML and JSON (already supported by most commercial databases) and unstructured data such as text documents, images, audio, and video. In the example shown in Figure 2, the data comes from different external sources such as DrugBank that offers data about known drugs and diseases, Comparative Toxicogenomics Database that provides information about gene interaction, and Uniprot that provides details about the functions and structure of genes.

One may argue that the proposed instance layer shares similar properties to those already found in the tabular representation of the relational model. However, a deeper question here is whether a tabular representation is an optimal choice for a *holistic data model*. Analytical workloads, for instance, benefit greatly from a columnar decomposition of tabular representation. In contrast, a self-curating database must manage data and meta-data in a unified way, but it is unclear what the optimal representation is for such systems. For example, could the relational model be further decomposed in non-linear and non-tabular form in order to cluster data based on the instance relations and semantic relationships of higher layers?

OPTIMIZATION STATEMENT 1. *Given the abundance of instance relations and semantic relationships, what are the data clustering opportunities to improve retrieval, access locality, and compression? Is it possible to develop dynamic instance-level, fine-grained clustering in the presence of the enriched data model?*[4]

[4]Imagine a representation that could adapt to the locality of access for a workload based on the interconnectness of data. The frequently accessed data could be packed together to be used efficiently in the limited, but fast-access memory of modern hardware including CPU cache or GPU and FPGA on-chip memory.

3.2 Relation Layer: Horizontal Expansion

A relational model has no notion of which columns refer to real world entities (i.e., data instances). But a holistic data model must possess a clear notion of what the entities are, and what relations exist for each instance in order to capture the data interconnectedness. These may be relations to other entities, or the relations of the attributes of the entity to data values. As an example, a database might have a table for *Drug*, and have the columns *Name*, *Targets*, *Symptomatic Treatment*. A rich data model has an identifier for a real world drug *Methotrexate*, and captures its attributes such as *Molecular Structure*, as well as relations to other entities including *Genes* that *Methotrexate* targets (e.g., *DHFR*), and subsequently, *Conditions* that it treats such as *Osteosarcoma (bone cancer)* that are reachable through its target genes, as shown in Figure 2.

The key characteristics of the relation layer are to capture entity interconnectedness and to establish the identity of an entity within and across multiple data sources – a process we term *horizontal data expansion* to transform data into information. An important challenge of the relation layer is to uniquely identify similar entities even when external sources are dynamically changing. There is a long history of entity resolution in the database literature, but the real challenge in this layer is that there is no ability to rely on manual ETL jobs to perform offline schema alignment, and it is not wise to assume that as each source is added to the self-curating database, an all-to-all entity resolution is performed comprehensively across all data sources.

FUNCTIONALITY STATEMENT 1. *A self-curating database must adaptively manage instance relations in light of new information. How does one adapt existing entity resolution techniques so they work across different schemata without requiring prior knowledge about external data sources to enable efficient incremental schema evolution in local data sources?*

FUNCTIONALITY STATEMENT 2. *Furthermore, what is the right formalism to express and capture the interconnectedness in order to assess and measure the richness of each data source based on the connectivity and density? For example, information content and capacity are a common measure for assessing the richness, and graph-theocratic approaches are well suited for studying the connectivity, flow properties, partitioning, and topology, but there is a lack of general formalism to assess the interconnectedness of data.*

Another challenge is how to efficiently manage relation interconnectedness. One may argue that a graph is the right abstraction model, but it leaves open the question of how to provide fast traversal abilities. Alternatively, one may argue that traditional indexes (e.g., B-Tree) may improve lookup, but at the high-level, indexes only provide one-hop away direct accesses, which are already captured in the explicit interconnectedness of the data. Thus, direct access is no longer beneficial, but rather the open challenge is how to improve the locality of multi-hop traversal.

OPTIMIZATION STATEMENT 2. *Given that the instance interconnectedness already encompasses the benefit of one-hop away direct access, what is an optimal representation that provides efficient locality-aware traversal that is tightly coupled with the instance and semantic layers and is update-friendly?*

3.3 Semantic Layer: Vertical Expansion

The instance layer together with relations between the instances, as discussed thus far, constitute what is often conceptually referred to as the ABox in the description logic and semantic web literature [3]. That is, instances refer to individual entities in the real world, relations among them are expressed in terms of semantic properties, and each instance is a member of one or more concepts or types. The concepts and semantic properties that are used in the ABox constitute meta-data about the instance data. Concepts themselves may have relationships to each other and semantic properties.

As a somewhat simple example, a *Drug* can be defined as a chemical with an existential quantification over the relation *has Target*. This means that if the actual instance data only stated that *Acetaminophen (Tylenol)* is a *Drug*, a self-curating database could infer that *Acetaminophen* has a target, even if the specific relation has yet to be discovered and expressed as a relationship between *Acetaminophen* and any particular gene. In fact *Acetaminophen* targets *PTGS2* (even though it is not shown in Figure 2).

These richer semantic reasonings are formulated and expressed in taxonomies or web ontology language (OWL), a subset of first-order logic (FOL). Relationships among the concepts and properties are typically referred to in the semantic web as the TBox [3]. To formalize our discussion, we focus on a widely employed OWL-DL language, which is based on the semantics of SHIN. The SHIN semantics is defined as $\mathcal{I}$= ($\Delta^{\mathcal{I}}$, $.^{\mathcal{I}}$), where $\mathcal{I}$ refers to an interpretation, $\Delta^{\mathcal{I}}$ is a non-empty set (the domain of the interpretation), and $.^{\mathcal{I}}$ is the interpretation function that maps every atomic concept C to a set $C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ (*Approved Drugs* is an example of the concept), every atomic role R to a binary relation $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ (e.g., *has Therapeutic Efficacy* as a role), and every individual a to $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$.

An RBox $\mathcal{R}$ is a finite set of transitivity axioms and role inclusion axioms of the form $R \sqsubseteq P$ where R and P are roles. A Tbox $\mathcal{T}$ is a set of concept inclusion axioms of the form $C \sqsubseteq D$, where C and D are concept expressions (e.g., *Neoplasms* $\sqsubseteq$ *Disease*). An Abox $\mathcal{A}$ is a set of axioms of the form $a : C$ (a is a member of the concept C), $R(a, b)$ (there is an R relationship between a and b), e.g., *has Target(Acetaminophen, PTGS2)*. Finally, an interpretation $\mathcal{I}$ is a model of an Abox $\mathcal{A}$ with respect to a Tbox $\mathcal{T}$ and a Rbox $\mathcal{R}$ *iff* it satisfies all the axioms in $\mathcal{A}$, $\mathcal{R}$, and $\mathcal{T}$.

A key strength of knowledge representation (KR) formalisms (such as OWL) derived from FOL stems from their capability to represent complex information in a knowledge rich domain, e.g., the biomedical domain. Unfortunately, FOL is incapable of dealing with inconsistency and uncertainty, which naturally arise when information from independent data sources is combined. The KR formalism should capture and aggregate information from both hard and soft sources. Hard sources may have a clear mathematical model of uncertainty, e.g., sensor data. Soft sources, on the other hand, provide vague statements of truth (often fuzzy), such as "a sudden stomach bleed was attributed to the *recent intake* of Ibuprofen". In contrast, there have been only isolated efforts to extend the KR languages to handle only a particular form of uncertainty, e.g., probabilistic or fuzziness [3]; thus, we raise the following question.

FUNCTIONALITY STATEMENT 3. *Is it possible to define a new unifying approach, but perhaps less expressive, to aggregate these isolated forms of uncertainty in a single tractable formalism?*

In general, we view the enrichment of data with semantics as *vertical data expansion* because this layer allows the database to infer new facts about the lower layers. We note that there is an increasing need for the vertical data expansion layer to be more general than the current notion of TBox $\mathcal{T}$. Increasingly, conceptual statistical models are being derived from the data to derive new connections between instance data. We therefore propose that the vertical data expansion be enriched by adding statistical models, such as those offered by machine learning, specifically to improve the linkage coverage and accuracy as well, considering that the purpose of this layer is to add semantic inference and reasoning capabilities about the instance types and the relationships among types.

FUNCTIONALITY STATEMENT 4. *In the semantic web literature, the assumption is that a user can specify the ontology as a logical view that can be applied over data with respect to a given query. Is it reasonable to have users be aware of the meta-models needed to understand the structure of the data, especially as one allows statistical models? And how does one describe a specific statistical model that should be applied over the data declaratively?*

To further benefit from the enriched data, there is a need for new formalism to combine the expressiveness power of database querying languages (e.g., SQL) with the semantic formalism of description logic (e.g., OWL) to capture the knowledge about the data.

FUNCTIONALITY STATEMENT 5. *Is it possible to develop a new semantically enriched query language that combines the expressiveness and declarativeness power of SQL (subset of FOL) and the leading semantic formalisms such as OWL (also subset of FOL) while retaining decades long advancement of query optimization and scalable query execution? Furthermore, is it possible to extend this new combined language with machine learning models that are based on non-declarative statistical, mathematical, or numerical formalism rather than the logical FOL formalism?*

Through semantically enriched data, there is an enormous opportunity to improve query optimization by inferring statistics given that today's optimizers fail completely in the absence of statistics on the data.

OPTIMIZATION STATEMENT 3. *How to extend the predominant rule- and cost-based query optimization to leverage the explicit semantics within our data model, so the optimizers are no longer limited to only statistics on data (e.g., selectivity estimates) to guide the query optimization (often missing or unavailable for external sources)? Is it possible to exploit the available semantics (e.g., exploiting class and subclass relationships) by inferring the selectivity and rewriting the query to a more efficient query (e.g., by inferring that certain predicates can be collapsed together semantically or can be dropped because they are redundant or unsatisfiable)?*

4. QUERY MODEL: EMBRACE CONTEXT

There is a compelling case to make queries less complicated through automatic exploration and refinement given the query context while the results must become evidence-based and justified (not limited to just a confidence score). Considering our proposed *holistic data model*, there are new opportunities to formalize and leverage the context of queries throughout the entire query pipeline, giving rise to a new way of thinking about how to query islands of data. We declare a pressing need to *rethink the entire query model* in a self-curating database; in particular, we focus on refining queries and computing answers through the continuous discovery and integration of data made possible by the rich data model.

4.1 Continuous Discovery and Refinement

In the database literature, we have the notion of adaptive query processing for collecting more accurate statistics during query execution to proactively optimize the query plan [2]. But conceptually our proposed *context-aware query model* opens up new avenues of research, in which not only more accurate statistics are gathered, but the query is also refined. In addition to refining the query, the data is also being adapted. Specifically, new instance relations or semantics relationships are discovered within the context of a given query (and its refined queries) as part of an online incremental integration, a step towards achieving the continuous integration.

Consider the task of determining an effective dosage of a drug by querying multiple clinical data sources. It is well-known that ethnicity and race have a major role in determining drug responses [9]. Now if these isolated data sources correspond to populations that are biased to genetic, ethnicity, and environmental conditions, then there is a tremendous value in automatically and judiciously navigating through these data sources without forcing the user to be fully aware of the semantics and interpretation of data that would be embedded in the enriched data model.

Suppose the initial query is "What is an effective dosage of Warfarin for preventing a blood clot?" (captured in Figure 2). Now to offer an accurate and justified answer in the presence of many disconnected data sources, there is a crucial need to develop an explorative querying framework that exploits the context of the query. To discover the necessary information and to fill the gap, the following refined queries may be posed automatically: "Is Warfarin sensitive to ethnic background?" (necessary to be aware of any medical facts); "What are the disjoint classes of population with respect to Warfarin?" (necessary for drilling down further); or "Does Warfarin have a narrow therapeutic range?" (necessary to quantify the dosage sensitivity and its range). We argue that such exploration is only possible by enriching data with sufficient semantics in order to interpret the context of queries and raise additional questions.

FUNCTIONALITY STATEMENT 6. *A new formalism is needed to express and execute the context-aware query model such that the discovery of new data connections and the refinement of query are feasible. Is it possible to formulate the discovery and refinement process as a random walk problem, where the initial seeds or the probability of each step taken is driven by query predicates and/or query partial results?*

FUNCTIONALITY STATEMENT 7. *Alternatively, is it possible to extend the query-by-example formalism [14] for filling missing data to introduce an incremental process so the query answer is partially computed, and the partial answer becomes an example with incompleteness (missing values) for raising/refining additional queries?*

To judge and choose the right formalism for context-aware query answering, we also need to revisit the existing evaluation criteria both in terms of completeness and feasibility.

FUNCTIONALITY STATEMENT 8. *To improve the discovery, is it possible to extend the crowdsourcing formalism to identify and assess the necessity to fetch incomplete data given certain qualitative (to improve the accuracy and coverage of answers) or quantitative (to find information faster) cost functions?*

4.2 Continuous Online Integration

The importance of online incremental integration for the *context-aware query model* is twofold. First, in a setting consisting of independently managed (but linked) data sources, individual data sources may change over time and one cannot be assured that all updates are propagated in a timely fashion. In fact, one of the main shortcomings of today's linked data initiative, in which large data sources are linked statically once, is the inability to deal with stale linked data [13]. Second, large scale one-time integration requires *a priori* knowledge to perform the integration, and this is not always feasible [8]. Moreover, it discards the knowledge of the users of the systems. Every time a user submits a query, the query may contain knowledge about the data, e.g., how two pieces of data are connected. One can think of a submitted query as a small scale but more focused and accurate integration that is at the instance-level and not necessarily at the schema-level.

FUNCTIONALITY STATEMENT 9. *There is a need for a new formalism to assess the correctness of query answering within the context of a single query while we discover and consult overlapping or even conflicting sources of information. More importunately, how do we formulate the feedback mechanism to materialize the discovered information guided by the context of query? If the discovered information is conflicting, then how could we automatically assess the richness or validity of discovered entities based on the degree of richness of each source (e.g., information content)?*

Today's formalisms for computing query answers focus on the inconsistencies, incompleteness, and uncertainty that arise within each data source or a set of integrated sources (i.e., a single consolidated view of data). The traditional probabilistic query answering relies on possible world semantics to assess the likelihood of answers by enumerating all possible worlds [5]. A well-known expressive representational model is a conditional table (c-table), in which each tuple t_i is associated with a Boolean formula (the condition c_i) [10]. The existence of a tuple in a possible world is subject to the satisfaction of its condition [10], c-tables are formally expressed as the valuation function of conditions $v(c)$.

Given an instance of data with uncertainty, we have a discrete probability space of $\mathcal{P} = (W, \mathbf{P})$, where W is a set of all the possible worlds given by $W = \{I_1, \cdots, I_n\}$ and $\mathbf{P}$ is a probability model that assigns probability $\mathbf{P}(I_i)$ to each possible world I_i such that $0 \leq \mathbf{P}(I) \leq 1$ and $\Sigma_{i=1}^{n} \mathbf{P}(I_i) = 1$. The probability of any tuple t is the total probability of all worlds in which t exists and can simply be computed by $\Sigma_{i=1, t \in I_i}^{n} \mathbf{P}(I_i)$.

Similarly, the incompleteness semantics $[\![]\!]$ is defined for an incomplete database D as a set of complete databases $[\![D]\!]$ that are constructed given an interpretation of null values $\mathcal{I}^{\mathsf{null}}$ under either as open- or closed-world assumption, $[\![]\!]_{\mathsf{OWA}}$ or $[\![]\!]_{\mathsf{CWA}}$, respectively [10]. The domain of the database consists of a set of constants (denoted by Const) and a set of nulls (denoted by Null), where the null represents the missing/unknown values. An example of a different interpretation of null values $\mathcal{I}^{\mathsf{null}}$ is Codd's three-valued logic.

Subsequently, the problem of query answering is reformulated as finding certain answers for the query Q. Given an interpretation of nulls $\mathcal{I}^{\mathsf{null}}$: the certain answer is defined as $\mathsf{certain}(Q, D) = \bigcap_i \{Q(D_i) | D_i \in [\![D]\!]\}$, which amounts to finding an intersection among a set of possible worlds. Notably, an incomplete database can be represented by a c-table [5], an important step towards unifying the representation model for both uncertainty and incompleteness [5, 10]. For instance, to capture both incompleteness and uncertainty, the c-tables semantics can be extended to included the valuation of nulls $v(t_i)$ and the valuation of conditions $v(c_i)$ so that a possible complete database instance I can be computed.

The existing techniques based on possible world semantics focus on deriving possible data instances from a single consolidated representation of data with uncertainty/incompleteness. However, there is no formalism to deal with multiple databases, where each source is complete and certain, but when viewed together without sufficient semantics, then uncertainty, incompleteness, and inconsistencies could arise. Let us revisit querying a set of independent sources, where each source captures clinical trials carried out in a different country and data is demographically biased; thus, naively combining the data from these sources may result in conflicting outcomes, even if data in each source is consistent/certain [9].

Consider a simple Boolean query "Is 5.0 mg an effective dosage

of Warfarin for preventing blood clot?". If the data was collected in white-dominant population, the effective daily dosage is expected to be around 5.1 mg, while in Asian and black population, daily doses of 3.4 mg and 6.1 mg are recommended, respectively [9]. Now, a naive evaluation may return false as the certain answer to our question (because not all sources report a 5.0 mg dosage rate) while semantically enriched data can infer that these reported dosage rates belong to three disjoint ethnic classes, and to compute the certain answer it is sufficient to have at least one dataset with a daily dose of "close" to 5.0 mg. Now the notion of closeness can further be formulated based on fuzzy logic in light of the fact that "Warfarin has a very narrow therapeutic range" [9]. Therefore, we argue that sufficient semantics are needed to capture the knowledge about the data premises (beyond today's lineage and provenance information) when integrating multiple data sources, and a new query answering formalism is needed to leverage the added knowledge.

In general, *derived possible worlds* are all constructed from a single integrated and consolidated actual world with incompleteness and/or uncertainty. But data at the web scale consisting of a large set of actual worlds (independent data sources) not just postulated probable worlds. These independent actual worlds, which we refer to as "*parallel worlds*" to distinguish them from the existing *possible worlds* semantics, may have conflicting facts, an alternative view of worlds, or relative facts that are only locally consistent given the premise of the particular world (i.e., semantics of the data). In short, information is relative with respect to the perspective of each independent source, and even in the absence of local inconsistency or uncertainty, the data may become contradictory when combined in the absence of sufficient semantics.

FUNCTIONALITY STATEMENT 10. *Firstly, is the exiting c-table formalism sufficiently expressive and concise to model our notion of parallel worlds with our proposed enriched and unified data model? For example, is the c-table representation required to be extended with relation and semantic layers (analogous to our holistic model) to faithfully capture the answers? Now, assuming a representational model, how do we formulate the notion of parallel world semantics for computing justified answers that may not always be globally justified in the presence of overlapping, complementary, and/or opposite relative views of worlds, where "justify" is taken as a fuzzy definition of "certain" to capture, possibly in a relaxed form, correctness and consistency for query semantics?*

In addition to the need for formalism and the semantics of query answering, there are other research challenges related to the execution semantics. As we continuously seek to discover and integrate new data sources and our holistic data model becomes more expressive, a whole set of challenges arise for transaction processing. For example, how do we ensure repeatability and guard against non-deterministic phantoms in transaction processing?

FUNCTIONALITY STATEMENT 11. *If the relation and semantic layers can be changed continuously, even when the instance layer does not change, and these layers are further enhanced with non-deterministic predictive inference power, could the classical isolation semantics (e.g., repeatibility or snapshot) ever be satisfied? In what ways must concurrency control be extended to account for the non-determinism that is not the result of explicit update queries? Is it possible to introduce relaxed isolation semantics (e.g., eventual consistencies) to account not only for a delay in receiving changes (i.e., pushed and eventually received), but also to account for situations where changes may never be sent explicitly and once received may be non-deterministic (i.e., pulled and eventually received with uncertainty)? These fundamental changes to the concurrency model will inevitably have implication for other components such as logging and recovery protocols.*

A system-level dimension of continuous integration and avoidance of today's pre-dominant *first-ingest-then-process* arises when considering the landmark shift of pushing both query execution and hosting of data sources on the cloud [1]. This synergy will introduce a whole new class of workload orchestration and optimization to reduce the cost of online integration and query answering.

OPTIMIZATION STATEMENT 4. *How can existing placement strategies be adapted to transition from disk data placement to placing data in distributed main memory at cloud scale? How can the data be judiciously placed in distributed shared memory with close affinity when online integration of data sources is likely in order to eliminate the storage access cost and to reduce the main memory footprint by avoiding data cache duplication?*

5. REVISITING DATABASE PRINCIPLES

In conclusion, to characterize our vision of self-curating databases, we revisit Codd's classical rules for relational systems and elaborate on how these rules must be extended to account for self-curating databases. In the process, we develop a comprehensive list of criteria that may serve as a test for self-curating databases.

- Deviation from *the foundation rule*: A self-curating database cannot assume that all data is managed locally and all data is in a relational model as was prescribed by Codd.
- Deviation from *the information rule*: Information is not limited to only the tabular form. A richer representation is essential to store information about the data. Meta-data and data representations must be unified and their distinction eliminated. Furthermore, every piece of information needs to be represented in the hierarchical multi-layered data model, where each layer semantically enriches the data, unlike Codd's vision that information is represented in only one way, namely, as a value in a table.
- Extending *the systematic treatment of null values rule*: The data model must allow each data item to be noisy, fuzzy, uncertain, or incomplete so that it can be manipulated systematically, in addition to the need for the nulls to represent missing values as advocated by Codd.
- Extending *the comprehensive data sublanguage rule*: The employed language must also support (1) data discovery and refinement operators and (2) multi-source transactions with limited access and concurrency enforcement on external sources, in addition to the language requirements stated by Codd.
- Deviation from *the view updating rule*: External views may not be updatable or forced to be updated incrementally and lazily, whereas Codd assumes all views must be strictly updatable.
- Deviation from *the integrity independence rule*: Constraints on data and meta-data are not limited to an independent set of rules maintained in the catalog (as required by Codd) because constraints are now modeled at the relation and semantic layers and data instances are physically linked.

6. REFERENCES

[1] The IBM strategy. Annual Report'13. http://www.ibm.com/annualreport/2013/, 2013.
[2] R. Avnur and J. M. Hellerstein. Eddies: Continuously adaptive query processing. In *SIGMOD'00*.
[3] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. 2003.
[4] R. Cattell. Scalable SQL and NoSQL data stores. *SIGMOD Rec.'10*.
[5] N. Dalvi, C. Ré, and D. Suciu. Probabilistic databases: Diamonds in the dirt. *Commun. ACM'09*.
[6] C. J. Date. *Date on Database: Writings 2000-2006*.
[7] D. A. Ferrucci, A. Levas, S. Bagchi, D. Gondek, and E. T. Mueller. Watson: Beyond Jeopardy! *Artif. Intell.'13*.
[8] M. Franklin, A. Halevy, and D. Maier. From databases to dataspaces: A new abstraction for information management. *SIGMOD Rec.'05*.
[9] J. A. Johnson. Ethnic differences in cardiovascular drug response potential contribution of pharmacogenetics. *Circulation'08*.
[10] L. Libkin. Incomplete data: What went wrong, and how to fix it. In *PODS'14*.
[11] S. Lohr. For big-data scientists, "janitor work" is key hurdle to insights. *The New York Times*. 2014.
[12] R. Reiter. Towards a logical reconstruction of relational database theory. In *On Conceptual Modelling (Intervale)*, 1982.
[13] F. M. Suchanek and G. Weikum. Knowledge bases in the age of big data analytics. *PVLDB'14*.
[14] M. M. Zloof. Query-by-example: the invocation and definition of tables and forms. In *VLDB'75*.