Υ-DB: Managing Scientific Hypotheses as Uncertain Data

Bernardo Gonçalves
LNCC – National Laboratory for Scientific Computing
Petrópolis, Brazil
bgonc@lncc.br

Fabio Porto
LNCC – National Laboratory for Scientific Computing
Petrópolis, Brazil
fporto@lncc.br

ABSTRACT

In view of the paradigm shift that makes science ever more data-driven, we consider deterministic scientific hypotheses as uncertain data. This vision comprises a probabilistic database (p-DB) design methodology for the systematic construction and management of U-relational hypothesis DBs, viz., Υ-DBs. It introduces hypothesis management as a promising new class of applications for p-DBs. We illustrate the potential of Υ-DB as a tool for deep predictive analytics.

1. INTRODUCTION

> "*Originally, there was just experimental science, and then there was theoretical science, with Kepler's Laws, Newton's Laws of Motion, Maxwell's equations, and so on. Then, for many problems, the theoretical models grew too complicated to solve analytically, and people had to start simulating.*"
>
> — Jim Gray

Large-scale experiments provide scientists with *empirical* data that has to be extracted, transformed and loaded before it is ready for analysis [7]. In this vision we consider deterministic scientific hypotheses seen as *theoretical* data, which also needs to be pre-processed to be analyzed, deserving then a proper database approach.

Hypotheses as data. As part of the paradigm shift that makes science ever more data-driven, scientific hypotheses are: (i) formed as principles or ideas, (ii) then mathematically expressed and (iii) implemented as a program that is run to give (iv) their *decisive* form of data (see Fig. 1).

Uncertain data. The semantic structure of item (iv) as shown in Fig. 1 can be expressed by the functional dependency (FD) $t \to \mathrm{v}\ s$. This is typical semantics assigned to empirical data in the design of experiment databases. A space-time dimension (like time t in our example) is used as a key to observables (like velocity v and position s). In *empirical* uncertainty, it is such "physical" dimension keys like t that may be violated, say, by alternative sensor readings.

Hypotheses, as tentative explanations of phenomena [10], are a different kind of uncertain data. In order to manage such *theoretical* uncertainty, we need two special attributes

Law of free fall

"If a body falls from rest, its velocity at any point is proportional to the time it has been falling."

(i)

$$a(t) \;=\; -g$$
$$\mathrm{v}(t) \;=\; -gt \,+\, \mathrm{v}_0$$
$$s(t) \;=\; -(g/2)t^2 \,+\, \mathrm{v}_0\, t \,+\, s_0$$

(ii)

```
for k = 0:n;
   t = k * dt;
   v = -g*t + v_0;
   s = -(g/2)*t^2 + v_0*t + s_0;
   t_plot(k) = t;
   v_plot(k) = v;
   s_plot(k) = s;
end
```

(iii)

FALL	t	v	s
	0	0	5000
	1	-32	4984
	2	-64	4936
	3	-96	4856
	4	-128	4744
	...	...	...

(iv)

Figure 1: Multi-fold view of a scientific hypothesis.

to compose, say, the epistemological dimension of keys to observables: ϕ, identifying the studied phenomena; and υ, identifying the hypotheses aimed at explaining them. That is, we shall leverage the semantics of item (iv) to $\phi\, \upsilon\, t \to \mathrm{v}\ s$. This leap is a core abstraction in the Υ-DB vision (see §3).

Predictive data. Scientific hypotheses are tested by way of their predictions [10]. In the form of mathematical equations, hypotheses symmetrically relate aspects of the studied phenomenon. However, for computing predictions, deterministic hypotheses are used asymmetrically as *functions* [12]. They take a given valuation over input variables (parameters) to produce values of output variables (the predictions). By observing that, in §3 we introduce a method to extract the FD schema of a hypothesis from its equations.

Big data. Scientific hypotheses qualify to at least four of the five v's associated to the notion of big data: *veracity*, due to their uncertainty; *value*, because of their role in advancing science and technology; *variety*, due to their structural heterogeneity (as noticeable in their FD schemes); and *volume*, because of the large scale of modern scientific problems.

Applications. Computational Science research programs such as the Human Brain Project or Cardiovascular Mathematics are highly-demanding applications challenged by such theoretical big data. Users need to analyze results of thousands of data-intensive simulation trials. Also, there is a pressing call for *deep* predictive analytic tools to support users assessing what-if scenarios in business enterprises [6]. All that motivates why *hypothesis management* is a promising class of applications for probabilistic databases (p-DBs).

However, despite the advanced state of the art of probabilistic data management techniques, a lack of systematic methods for the design of p-DBs may prevent wider adoption. In analogy with the field of Graphical Models (GM), considered to inform research in p-DBs [13, p. 14], one of the key success factors for the rapid growth of applications was the availability of systematic methods of construction [4].

The vision of Υ-DB addresses that gap by bringing forward one such methodology on top of U-relations and probabilistic world-set algebra (p-WSA) [8].

Predictive analytics. Deep predictive analytics [6] is meant to support users in assessing the consequences of alternative hypotheses. If these can be identified (see §3), and their uncertainty is quantified by some probability distribution (see §4), then they can be ranked and browsed by the user under selectivity criteria. Furthermore, their probabilities can be conditioned on observed data such that they are possibly re-ranked in the presence of evidence (see §5).

2. RELATED WORK

Haas et al. [6] propose a long-term *models-and-data* research program to address deep predictive analytics. Our vision, with roots in [11], is essentially an abstraction of *hypotheses as data*. It can be understood in comparison as putting models strictly into a data perspective. Thus it is directly applicable by building upon recent work on p-DBs [13]. Our vision comprises a *p-DB design methodology* for the systematic construction and management of U-relational hypothesis DBs, viz., Υ-DBs. It applies classical FD theory [14] and the U-relational representation system with its p-WSA query algebra [8]. It is not to be confused, say, with initiatives to revisit FD theory in view of uncertain DB design [5].

U-relations and p-WSA were developed in the influential MayBMS project.[1] As implied by its design principles, e.g., compositionality and the ability to introduce uncertainty, MayBMS' query language [8] fits well to hypothesis management. Noteworthy, the `repair key` operation gives rise to alternative worlds as maximal-subset repairs of an argument key. We shall look at it from the point of view of p-DB design, for which no methodology has yet been proposed.

Again in analogy with GMs, it may be clarifying to distinguish methods for p-DB design in three classes [4]: (i) *subjective* construction, (ii) *synthesis* from other kind of formal specification, and (iii) *learning* from data. The first is the less systematic, as the user has to model for the data and correlations by steering all the p-DB construction process (MayBMS' use cases [8], e.g., are illustrated that way). The second is typified by the Υ-DB methodology, as we extract FDs and synthesize U-relations from mathematical equations as a kind of formal specification (see §3-§4). To our knowledge, this is the first work to propose a synthesis method for p-DB design. The third comprises analytical techniques to extract the data and model correlations from external sources, possibly unstructured, into a p-DB. This is the prevalent one, motivated by information extraction, data integration and data cleaning applications [13, p. 10-3].

Also related to the Υ-DB vision is the topic of conditioning a p-DB. It has been firstly addressed by Koch and Olteanu motivated by data cleaning applications [9]. They have introduced the `assert` operation to implement, as in AI, a kind of knowledge compilation, viz., world elimination in face of constraints (e.g., FDs). For hypothesis management, nonetheless, we need to apply *Bayes' conditioning* by asserting observed data (not constraints). In §5, we present an example that settles the kind of conditioning problem that is relevant to Υ-DB.

In order to provide a concrete feel of our vision, in the next sections we present preliminary results on our methodology for constructing an Υ-DB on top of MayBMS.

[1] `http://maybms.sourceforge.net/`

PHENOMENON	ϕ	Description
	1	Effects of gravity on an object falling in the Earth's atmosphere.

HYPOTHESIS	υ	Name
	1	Law of free fall
	2	Stokes' law
	3	Velocity-squared law

Figure 2: Descriptive (textual) data of Example 1.

3. HYPOTHESIS ENCODING

Let us consider Example 1 to illustrate our methodology.

EXAMPLE 1. *A research is conducted on the effects of gravity on a falling object in the Earth's atmosphere. Scientists are uncertain about the precise object's density and its predominant state as a fluid or a solid. Three hypotheses are then considered as alternative explanations of the fall (see Fig. 2). Because of parameter uncertainty, six simulation trials are run for* H_1*, and four for* H_2 *and* H_3 *each.* □

The construction of Υ-DB requires a simple user description of a research. Hypotheses must be associated to the phenomena they explain and then assigned a prior confidence distribution which may or may not be uniform (see Fig. 3a, top). Then the FD schema of each hypothesis has to be extracted from its mathematical equations. Let us examine (H_1) the *law of free fall* (Fig. 1) and its set Σ_1 of FDs.

$$\begin{aligned} \Sigma_1 = \{ \quad \phi &\rightarrow g\, \mathrm{v}_0\, s_0, \\ g\, \upsilon &\rightarrow a, \\ g\, \mathrm{v}_0\, t\, \upsilon &\rightarrow \mathrm{v}, \\ g\, \mathrm{v}_0\, s_0\, t\, \upsilon &\rightarrow s \quad \}. \end{aligned}$$

In order to derive Σ_1 from the equations in Fig. 1-(ii), we focus on their hidden data dependencies and get rid of constants and possibly complex mathematical constructs. Equation $\mathrm{v}(t) = -gt + \mathrm{v}_0$, e.g., written this way, roughly speaking, allows us to infer that v is a prediction variable *functionally dependent* on t (the physical dimension), g and v_0 (the parameters). Yet a dependency like $g\, \mathrm{v}_0\, t \rightarrow \mathrm{v}$ may hold for infinitely many equations. We need a way to precisely identify H_1's formulation, i.e., an abstraction of its data-level semantics. This is achieved by introducing hypothesis id υ as a special attribute in the FD (see Σ_1). This is a *data representation* of a scientific hypothesis. The other special attribute, the phenomenon id ϕ, is supposed to be a key to the value of parameters, i.e., determination of parameter values is an empirical, phenomenon-dependent task. FD $\phi \rightarrow g\, \mathrm{v}_0\, s_0$ is to be (expectedly) violated when the user is uncertain about the values of parameters.

The same rationale applies to derive $\Sigma_2 = \Sigma_3$ from the equations of H_2, H_3 below. These vary in structure w.r.t. H_1 (e.g., parameter D, the object's diameter). The key point here is that the method to extract the hypothesis FD schema from its equations is reducible to a language for mathematical modeling (based on W3C's MathML).

H_2. Stokes' law	H_3. Velocity-squared law
$a(t) = 0$	$a(t) = 0$
$\mathrm{v}(t) = -\sqrt{gD/\,4.6 \times 10^{-4}}$	$\mathrm{v}(t) = -gD^2/\,3.29 \times 10^{-6}$
$s(t) = -t\,\sqrt{gD/\,4.6 \times 10^{-4}} + s_0$	$s(t) = -(gD^2/\,3.29 \times 10^{-6})\,t + s_0$

$$\begin{aligned} \Sigma_2 = \Sigma_3 = \{ \quad \phi &\rightarrow g\, D\, s_0, \\ \upsilon &\rightarrow a, \\ g\, D\, \upsilon &\rightarrow \mathrm{v}, \\ g\, D\, s_0\, t\, \upsilon &\rightarrow s \quad \}. \end{aligned}$$

EXPLANATION	ϕ	υ	Conf
	1	1	0.6
	1	2	0.2
	1	3	0.2

H1_INPUT	tid	ϕ	g	v_0	s_0
	1	1	32	0	5000
	2	1	32	10	5000
	3	1	32	20	5000
	4	1	32.2	0	5000
	5	1	32.2	10	5000
	6	1	32.2	20	5000

H1_OUTPUT[a]	tid	ϕ	υ	a
	1	1	1	-32
	2	1	1	-32
	3	1	1	-32
	4	1	1	-32.2
	5	1	1	-32.2
	6	1	1	-32.2

(a) Simulation raw data: trials on H_1 identified by tid.

$\rightarrow$

W	$V \mapsto D$	Pr
	$x_1 \mapsto 1$	.6
	$x_1 \mapsto 2$	.2
	$x_1 \mapsto 3$	.2
	$x_2 \mapsto 1$	.5
	$x_2 \mapsto 2$	.5

Y[Exp]	$V \mapsto D$	ϕ	υ
	$x_1 \mapsto 1$	1	1
	$x_1 \mapsto 2$	1	2
	$x_1 \mapsto 3$	1	3

Y1[g]	$V \mapsto D$	ϕ	g
	$x_2 \mapsto 1$	1	32
	$x_2 \mapsto 2$	1	32.2

Y1[a]	$V_1 \mapsto D_1$	$V_2 \mapsto D_2$	ϕ	υ	a
	$x_1 \mapsto 1$	$x_2 \mapsto 1$	1	1	-32
	$x_1 \mapsto 1$	$x_2 \mapsto 2$	1	1	-32.2

Y[a]	$V_1 \mapsto D_1$	$V_2 \mapsto D_2$	ϕ	υ	a
	$x_1 \mapsto 1$	$x_2 \mapsto 1$	1	1	-32
	$x_1 \mapsto 1$	$x_2 \mapsto 2$	1	1	-32.2
	$x_1 \mapsto 2$	$-$	1	2	0
	$x_1 \mapsto 3$	$-$	1	3	0

(b) Probabilistic Υ-DB storing H_i as Υ_i, for $i = 1..3$, in MayBMS.

Figure 3: Synthesis of probabilistic Υ-DB from FD schemes and the simulation input/output data (Υ-DB's raw data).

Once each hypothesis FD schema has been extracted, some reasoning is to be performed to synthesize its certain relations. The decomposition and pseudo-transitivity inference rules [14] on $\{\phi \to g\, \mathrm{v}_0\, s_0,\ g\, \upsilon \to a\} \subset \Sigma_1$, e.g., give $\phi\, \upsilon \to a$. Yet there is an extra attribute, tid, added by default to such relations (see Fig. 3a) in order to identify each simulation trial and "pretend" completeness.[2] It is under this completeness that the "raw" data is loaded from input/output simulation files. Note, however, that it is held at the expense of redundancy and, mostly important, opaqueness for predictive analytics (since tid isolates or hides inconsistency). This is until the next stage of the Υ-DB construction method, when the uncertainty is introduced in a controlled manner.

4. UNCERTAINTY INTRODUCTION

The transformation of relations in Fig. 3a to probabilistic Υ-DB starts with query Q_1, creating relation Y[Exp] (Fig. 3b).

Q_1. **create table** Y_Exp **as select** phi, upsilon **from**
(**repair key** phi **in** EXPLANATION **weight by** Conf);

The Y-relations (Fig. 3b) have in their schema a set of pairs (V_i, D_i) of *condition columns* (cf. [8]) to map each discrete random variable x_i to one of its possible values (e.g., $x_1 \mapsto 1$). The world table W stores their marginal probabilities.

We create decompositions $\mathsf{Y1}[\vec{X}]$ for each independent uncertainty unit $\vec{X} \subseteq \vec{A}$ in $\mathsf{H1INPUT}(\mathsf{tid}, \phi, \vec{A})$. Query Q_2, e.g., maps the possible values of g to random variable x_2.

Q_2. **create table** Y1_g **as select** U.phi, U.g **from**
(**repair key** phi **in** (**select** phi, g, **count(*)** **as** Fr **from**
H1_INPUT **group by** phi, g) **weight by** Fr) as U;

The result set of Q_2 is stored in Y1[g] (see Fig. 3b). Note that we consider relation H1_INPUT as a joint probability distribution on the values of H_1's parameters and it may not be uniform: we count the frequency Fr of each possible value of an uncertainty factor $\vec{X} \subseteq \vec{A}$ (as done for g in Q_2) and pass it as argument to the `weight by` construct.

Then, by considering $g\, \upsilon \to a \in \Sigma_1$, we are able to synthesize prediction relation Y1[a] as a query: since a is functionally determined by υ and g only, and these are independent, we propagate their uncertainties onto a by query Q_3 in the local scope of Y1[a] (and similarly for Y2[a] and Y3[a]).

[2] Considering {tid, ϕ} and {tid, ϕ, υ} as keys in the relations.

Q_3. **create table** Y1_a **as select** H.phi, H.upsilon, H.a **from**
H1_OUTPUT_a **as** H, Y_Exp **as** E, Y1_g **as** G, (**select min**(tid)
as tid, phi, g **from** H1_INPUT **group by** phi, g) **as** U
where H.tid=U.tid **and** G.phi=U.phi **and** G.g=U.g
and H.phi=E.phi **and** H.upsilon=E.upsilon;

Query Q_3' (not shown) is a *union all* query selecting ϕ, υ and a from Yi[a] for each $i = 1..3$. The result sets of Q_3 and Q_3', resp. Y1[a] and Y[a], are shown in Fig. 3b.

Now, compare relations H1_OUTPUT[a] and Y1[a]. By accounting for the correlations captured in the FD $g\, \upsilon \to a$, we could propagate onto a the uncertainty coming from the hypothesis and the only parameter it is sensible to, thus precisely situating tuples of Y1[a] in the space of possible worlds. The same is done for predictive attributes v and s. In the end, we have Υ-DB ready for predictive analytics, i.e., with all competing predictions as possible alternatives which are mutually inconsistent.

The key point here is that all the synthesis process is amenable to algorithm design. Except for the user research description, the Υ-DB construction is fully automated based on the FD schemes and the simulation raw data.

5. PREDICTIVE ANALYTICS

Users of Ex. 1, has to be able, say, to query phenomenon $\phi = 1$ w.r.t. predicted position s at specific times t by considering all hypotheses υ admitted. That is illustrated by query Q_4, which creates integrative table Y[s]; and by query Q_5, which computes the confidence aggregate [8] for all s tuples where $t = 3$ (Fig. 4 shows Q_5's result, apart from column Posterior).

The confidence on each hypothesis for the specific prediction of Q_5 is split due to parameter uncertainty such that they sum up back to its total confidence. For H_2 and H_3, e.g., we have $\{g\, D\, s_0\, t\, \upsilon \to s\} \subset \Sigma_2 = \Sigma_3$. Since g and D are the parameter uncertainty factors of s (s_0 is certain), with 2 possible values (not shown) each, then there are only $2 \times 2 = 4$

Q_4. **create table** Y_s **as select** U.phi, U.upsilon, U.t, U.s **from**
(**select** phi, upsilon, t, s **from** Y1_s **union all**
select phi, upsilon, t, s **from** Y2_s **union all**
select phi, upsilon, t, s **from** Y3_s) **as** U, Y_Exp **as** E
where U.phi=E.phi **and** U.upsilon=E.upsilon;

Q_5. **select** phi, upsilon, s, **conf()** **as** Prior **from** Y_s **where** t=3
group by phi, upsilon, s **order by** Prior **desc**;

$Y[s]$	ϕ	υ	s	Prior	Posterior
	1	1	2188.36	.1	.167
	1	1	2205.82	.1	.168
	1	1	2320.51	.1	.167
	1	1	2337.97	.1	.165
	1	1	2452.66	.1	.149
	1	1	2470.12	.1	.145
	1	2	2930.59	.05	.020
	1	2	2943.44	.05	.019
	1	2	4991.92	.05	.000
	1	2	4991.97	.05	.000
	1	3	4778.87	.05	.000
	1	3	4779.56	.05	.000
	1	3	4944.72	.05	.000
	1	3	4944.89	.05	.000

Figure 4: Υ-DB query for analytics on predicted position s.

possible s tuples for H_2 and H_3 each. Considering all hypotheses υ for the same phenomenon ϕ, the confidence values sum up to one in accordance with the laws of probability.

Users can make decisions in light of such confidence aggregates. These are to be eventually conditioned in face of evidence (observed data). Example 2 features it for *discrete* random variables mapped to the possible values of predictive attributes (like position s) whose domain are *continuous*.

EXAMPLE 2. *Suppose position* $s = 2250$ *feet is observed at* $t = 3$ *secs, with standard deviation* $\sigma = 20$. *Then, by applying Bayes' theorem for normal mean with a discrete prior* [3], *Prior is updated to Posterior (see Fig. 4).* □

The procedure uses normal density function (1), with $\sigma = 20$, to get the likelihood $f(y \mid \mu_i)$ of each alternative prediction of s from $Y[s]$ as mean μ_i given y at observed $s = 2250$. Then it applies Bayes' rule (2) to get the posterior $p(\mu_i \mid y)$.

$$f(y \mid \mu_i) \;=\; \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{1}{2\sigma^2}(y-\mu_i)^2} \tag{1}$$

$$p(\mu_i \mid y) \;=\; f(y \mid \mu_i)\, p(\mu_i) \;/\; \textstyle\sum_{i=1}^{n} f(y \mid \mu_i)\, p(\mu_i) \tag{2}$$

6. RESEARCH CHALLENGES

Big data in general, and hypotheses as data in particular, challenge traditional DB design methodologies [1]. Meanwhile, p-DB models like MayBMS' extend the relational model opening new opportunities, in particular for *design by synthesis* [2]. New problems span from fast-varying schemas to uncertainty, probability and correlations in the raw data.

Structural variety. The user external view of the world is constantly changing. Our approach to this challenge consists in isolating or safeguarding alternative views under their own FD schemes and epistemological keys, allowing for their co-existance in the same p-DB in a controlled way.

Dependency extraction. It has been considered a critical failure in traditional DB design the lack of techniques to obtain important information (e.g., FDs) in the real world [1, p.62]. Synthesis methods for p-DB design shall provide novel abstractions and techniques to extract dependencies from other kinds of formal specification (e.g., equations).

Schema synthesis. Predictive data has correlations or, an *uncertainty chaining*, we capture in FDs. Reasoning to synthesize relations has to account for that, viz., it has to go beyond 3NF and compute the pseudo-transitive closure (PTC) of each FD schema. For example, running the classical 3NF synthesis algorithm [2] on Σ_1 produces relation $R_i(g, \mathrm{v}_0, s_0, \upsilon, t, s)$, whereas we target at the also lossless, but less redundant $R_i(\phi, \upsilon, t, s)$. That is, parameters are to be folded *for certainty* (run PTC on Σ_1), and then unfolded *for uncertainty* (re-run it on $\Sigma_1 \setminus \{\phi \to g\, \mathrm{v}_0\, s_0\}$).

Uncertainty factors. PTC for uncertainty (u-PTC) must synthesize prediction relations (e.g., $Y1[a]$) with the proper uncertainty factors in their condition columns. Besides υ, a trivial factor, identifying each independent uncertainty unit from trials (cf. H1_INPUT) with one, only one random variable is a combinatorial problem of *uncertainty factor learning*. Thus, u-PTC must be sensitive not exactly to parameters A but to the uncertainty factors $\vec{X} \subseteq \vec{A}$ they fall into.

Cyclic FDs. In the hypotheses of Ex. 1, no prediction variable is dependent on each other. Complex mathematical models, however, have coupled variables leading to cyclic FDs like $\{a\, x\, \upsilon \to y,\; b\, y\, \upsilon \to x\}$. This is a specific issue of cycles in the uncertainty chaining for the (u-)PTC algorithm.

Conditioning. The prior probability distribution assigned via repair key to uncertainty factors (cf. Q_1, Q_2) is to be eventually conditioned on observed data (Ex. 2). This is an applied *Bayesian inference* problem that translates into a *p-DB update* one to induce effects of posteriors back to table W. It is achievable (yet unclean) in MayBMS' update language.

7. CONCLUSIONS

We have presented the vision of Υ-DB, which is essentially an abstraction of hypotheses as uncertain data. It comprises a design methodology for the systematic construction and management of U-relational hypothesis DBs. To our knowledge this is the first design-by-synthesis method for constructing p-DBs from formal specifications.

We have introduced hypothesis management as a promising new class of applications for p-DBs, providing a principled approach to manage theoretical big data on top of MayBMS. The potential of Υ-DB for deep predictive analytics has also been illustrated. First results are to be delivered from a large-scale use case in Computational Hemodynamics.

8. ACKNOWLEDGMENTS

This research has been supported by the Brazilian funding agencies CNPq, grants n^o 141838/2011-6 and 309494/2012-5, and FAPERJ grant n^{o}E-26/100.286/2013. We thank IBM for awarding this project a Ph.D. Fellowship 2013-2014.

9. REFERENCES

[1] A. Badia and D. Lemire. A call to arms: Revisiting database design. *SIGMOD Record*, 40(3):61–9, 2011.

[2] P. Bernstein. Synthesizing third normal form relations from functional dependencies. *ACM TODS*, 1(4):277–98, 1976.

[3] W. M. Bolstad. *Introduction to Bayesian Statistics.* Wiley-Interscience, 2nd edition, 2007.

[4] A.Darwiche. Bayesian networks. *Comm ACM*, 53(12):80–90, 2010.

[5] A. Das Sarma, J. Ullman, and J. Widom. Schema design for uncertain databases. In *Proc. of AMW*, 2007.

[6] P. Haas, P. Maglio, P. Selinger, and W. Tan. Data is dead... without what-if models. *PVLDB*, 4(12):1486–9, 2011.

[7] T. Hey et al. *The Fourth Paradigm: Data-Intensive Scientific Discovery.* Microsoft Research, 2009.

[8] C. Koch. *MayBMS: A system for managing large uncertain and probabilistic databases.* In C. Aggarwal (ed.), Managing and Mining Uncertain Data, chapter 6. Springer-Verlag, 2009.

[9] C. Koch and D. Olteanu. Conditioning probabilistic databases. *PVLDB*, 1(1):313–25, 2008.

[10] J. Losee. *A historical introduction to the philosophy of science.* Oxford University Press, 4th edition, 2001.

[11] F. Porto and S. Spacappietra. *Data model for scientific models and hypotheses.* In R. Kaschek, L. Delcambre (ed.), The evolution of Conceptual Modeling, p. 285-305, Springer-Verlag LNCS vol. 6520, January 2011.

[12] H. Simon and N. Rescher. Cause and counterfactual. *Philosophy of Science*, 33(4):323–40, 1966.

[13] D. Suciu, D. Olteanu, C. Ré, and C. Koch. *Probabilistic Databases.* Morgan & Claypool Publishers, 2011.

[14] J. Ullman. *Principles of Databases and Knowledge-Base Systems.* Computer Science Press, 1988.