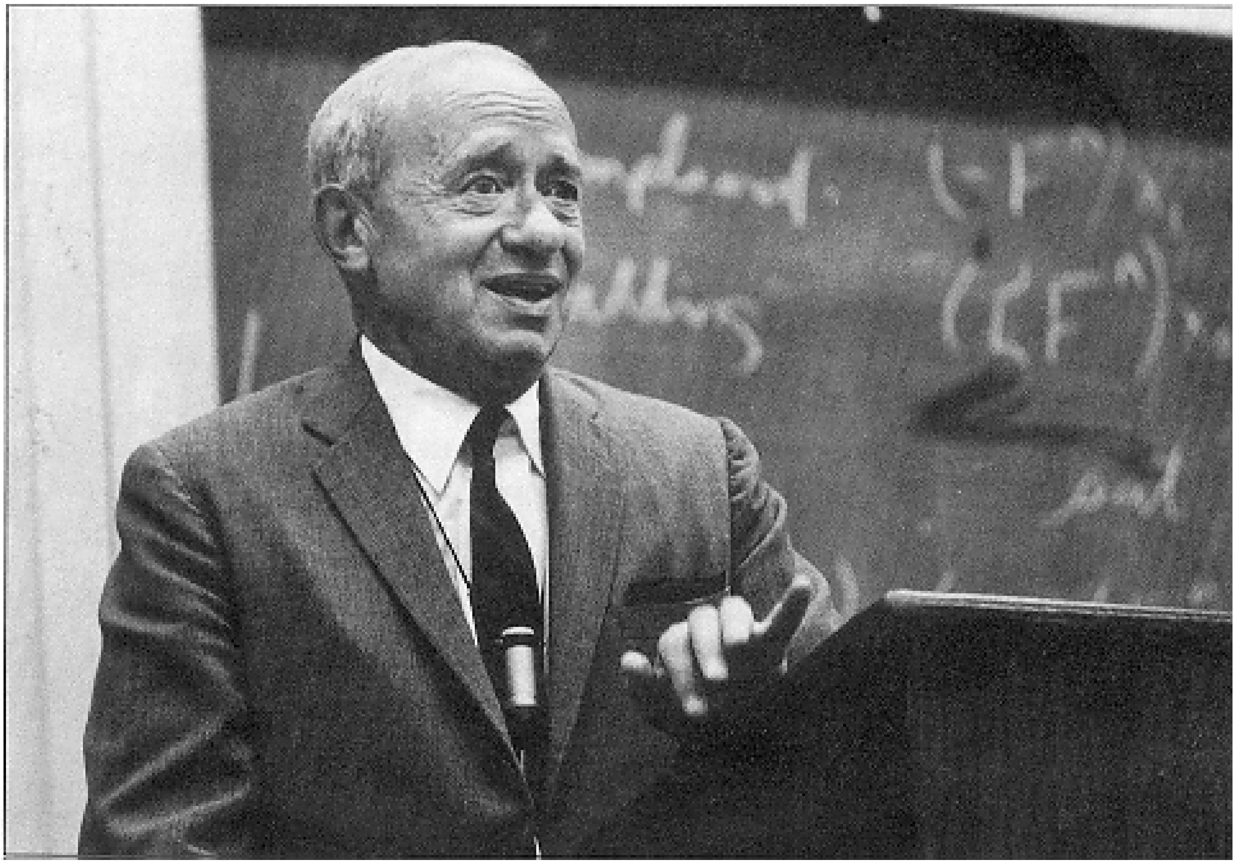

Applications of Alfred Tarski's Ideas in Database Theory

Jan Van den Bussche

U. Limburg, Belgium

Relational databases

Fix some infinite universe $\mathbb{U}$ of atomic data elements

Database schema: Finite set $\mathcal{S}$ of relation names

Relational database $\mathbf{D}$ *with schema* $\mathcal{S}$*:* Assigns to each $R \in \mathcal{S}$ a finite relation $R^{\mathbf{D}} \subseteq \mathbb{U}^n$

Examples of queries

Assume $\mathcal{S} = \{R\}$ with R binary: database is finite binary relation on $\mathbb{U}$

1. Is there an identical pair in R?

2. What are the elements occurring in the left column of R, but not in the right?

3. What are the 5-tuples $(x_1, x_2, x_3, x_4, x_5)$ such that (x_1, x_2), (x_2, x_3), (x_3, x_4), and (x_4, x_5) are all in R?

4. What is the transitive closure of R?

5. Which pairs of elements (x_1, x_2) are such that the sets

$$\{y \mid (x_1, y) \in R\} \quad \text{and} \quad \{y \mid (x_2, y) \in R\}$$

are nonempty and have the same cardinality?

6. Is the cardinality of R a prime number?

A formal definition of query

Answer of query is again a relation

$\Rightarrow$ A *query on* $\mathcal{S}$ is a function q:

- from databases $\mathbf{D}$ with schema $\mathcal{S}$
- to finite relations $q(\mathbf{D}) \subseteq \mathbb{U}^n$

This definition is much too liberal

A query that is "illogical"

$$\boxed{\begin{array}{cc} a & b \\ a & c \end{array}} \quad \mapsto \quad \boxed{b}$$

There is no reason to favor b above c

None of the example queries has this illogical nature

A query must be answerable purely on the basis of the information present in the database

How to formalize this?

Tarski's logical notions

Cumulative hierarchy:

$$\mathbb{U}_0 := \mathbb{U}, \quad \mathbb{U}_{n+1} := \mathbb{U} \cup \mathcal{P}(\mathbb{U}_n), \quad \mathbb{U}^* := \bigcup_n \mathbb{U}_n$$

Many mathematical objects constructed on top of $\mathbb{U}$ live in $\mathbb{U}^*$

In particular databases and queries

Tarski: $P \in \mathbb{U}^*$ is *logical* if $f(P) = P$ for every permutation of $\mathbb{U}$

- No individual element of $\mathbb{U}$ is logical
- $\mathbb{U}$ and $\varnothing$ are logical
- identity and diversity relations are logical

The higher up we go, the more complex logical notions we find

Generic queries

All six example queries are logical

Our "illogical" query is indeed not logical

Genericity: Consistency criterion for queries from early days of database theory, based on practical considerations

[Aho&Ullman, Chandra&Harel]

Query q is generic if for all permutations f of $\mathbb{U}$:

$$f(\mathbf{D}_1) = \mathbf{D}_2 \quad \Rightarrow \quad f(q(\mathbf{D}_1)) = q(\mathbf{D}_2)$$

A query is generic iff it is logical in Tarski's sense!

Codd's relational algebra

Operations on data files expressed as combinations of five basic operators on relations

1. union $r \cup s$

2. difference $r - s$

3. cartesian product $r \times s$

4. projection

   $$\pi_{i_1,\ldots,i_p}(r) = \{(x_{i_1},\ldots,x_{i_p}) \mid (x_1,\ldots,x_n) \in r\}$$

5. selection

   $$\sigma_{i=j}(r) = \{(x_1,\ldots,x_n) \in r \mid x_i = x_j\}$$

Example expressions

(2) What are the elements occurring in the left column of R, but not in the right?

$$\pi_1(R) - \pi_2(R)$$

(3) What are the 5-tuples $(x_1, x_2, x_3, x_4, x_5)$ such that (x_1, x_2), (x_2, x_3), (x_3, x_4), and (x_4, x_5) are all in R?

$$\pi_{1,2,4,6,8}\sigma_{2=3}\sigma_{4=5}\sigma_{6=7}(R \times R \times R \times R)$$

First-order queries

Query q on $\mathcal{S}$ is called *first-order* if there is a first-order formula $\varphi(x_1, \ldots, x_n)$ over $\mathcal{S}$ such that

$$q(\mathbf{D}) = \{(a_1, \ldots, a_n) \in |\mathbf{D}|^n \mid \mathbf{D} \models \varphi[a_1, \ldots, a_n]\}$$

$|\mathbf{D}|$: *active domain* of $\mathbf{D}$

Codd's Theorem: q expressible in Codd's relational algebra $\Leftrightarrow$ q first-order

Tarskian definition of $\models$

First-order queries are generic: anything definable in higher-order logic is logical
[Lindenbaum&Tarski 1934]

Relational completeness

Codd: completeness result for relational algebra

$\Rightarrow$ "Relational completeness" of database query languages

However, many interesting queries are not first-order:

(4) What is the transitive closure of R?

(5) Which pairs of elements (x_1, x_2) are such that the sets

$$\{y \mid (x_1, y) \in R\} \quad \text{and} \quad \{y \mid (x_2, y) \in R\}$$

are nonempty and have the same cardinality?

(6) Is the cardinality of R a prime number?

BP-completeness

So, Codd's relational algebra (FO) is hardly complete

Still: completeness on the input level
[Bancilhon, Paredaens]

> For any generic query q and database $\mathbf{D}$ there exists a first-order query $q_{\mathbf{D}}$ such that $q_{\mathbf{D}}(\mathbf{D}) = q(\mathbf{D})$

Tarski: Finite structures that are elementary equivalent are isomorphic

Together with Beth's Theorem, this readily implies BP-completeness of FO

$\Rightarrow$ **CSPs:** Even without $\cup$ and $-$ (but with $\sigma_{i \neq j}$) relational algebra is already BP-complete
[Cohen, Gyssens, Jeavons]

Cylindric set algebra

Take first-order formula φ with all variables (free or bound) among $x_1, \ldots, x_n$

$\Rightarrow$ For database $\mathbf{D}$, to determine $\mathbf{D} \models \varphi$, we inductively apply operations on n-ary relations over $|\mathbf{D}|$:

1. union (for $\vee$)

2. complementation w.r.t. $|\mathbf{D}|^n$ (for $\neg$)

3. cylindrification along dimension i (for $\exists x_i$)

$$\gamma_i(r) = \{(a_1, \ldots, a_n) \in |\mathbf{D}|^n \mid \exists a \in |\mathbf{D}| : (a_1, \ldots, a_{i-1}, a, a_{i+1}, \ldots, a_n) \in r\}$$

Together with constant relations

$$\delta_{ij} = \{(a_1, \ldots, a_n) \in |\mathbf{D}|^n \mid a_i = a_j\}$$

constitute *full n-dimensional cylindric set algebra with base $|\mathbf{D}|$*

Codd's Theorem avant la lettre

Build up n-*CSA expressions* from relation names in $\mathcal{S}$ using operators and constants of n-CSA

Interpret k-ary relation R in $\mathbf{D}$ as $R^{\mathbf{D}} \times |\mathbf{D}|^{n-k}$ to make everything n-ary

Must assume $k < n$ for every R

Theorem: q in n-CSA $\Leftrightarrow$ q in FO^n (first-order formulas with at most n variables)

$\Rightarrow$ Cylindric algebra as relational algebra avant la lettre

Proof is trick also invented by Tarski to give substitution-free axiomatization of first-order logic with equality

Relation algebras

Proper relation algebra with base A consists of operations on binary relations on A:

1. union

2. complementation w.r.t. A^2

3. composition

   $r \odot s := \{(x, y) \mid \exists z : (x, z) \in r \text{ and } (z, y) \in s\}$

4. conversion: $\tilde{r} := \{(x, y) \mid (y, x) \in r\}$

Schema $\mathcal{S}$ with all relation names binary

$\Rightarrow$ Build *RA-expressions* from relation names in $\mathcal{S}$ using these operators and constant Id (identity relation)

To evaluate expression on $\mathbf{D}$, use base $|\mathbf{D}|$

From FO^3 to FO

Tarski&Givant: q in RA $\Leftrightarrow$ q in FO^3

But also: In structures with pairing, RA becomes equally powerful as full FO

$\Rightarrow$ Add pairing operators to RA [Van Gucht et al]

- left tagging: $r^{\triangleleft} = \{(x,(x,y)) \mid (x,y) \in r\}$

- right tagging: $r^{\triangleright} = \{((x,y),y) \mid (x,y) \in r\}$

These operations work over $\mathbb{U}^+$ rather than $\mathbb{U}$:

$$\mathbb{U}_0^+ := \mathbb{U}, \quad \mathbb{U}_{n+1}^+ := \mathbb{U}_n^+ \cup (\mathbb{U}_n^+)^2, \quad \mathbb{U}^+ := \bigcup_n \mathbb{U}_n^+$$

Resulting query language RA$^+$ equivalent to FO

Computational completeness

Make RA$^+$ into programming language:

- variables (hold binary relations on $\mathbb{U}^+$)
- assignment statements: $X := e$
- composition, while-loops

$X := R$;
while $(X \odot R) - X \neq \varnothing$ **do**
  $X := X \cup X \odot R$

Every computable query is expressible
[Chandra&Harel, Abiteboul&Vianu]

$\surd$ Computable queries with answers over $\mathbb{U}^+$

Answers over $\mathbb{U}^*$:

$$r^{\triangle} = \Big\{ \Big(x, \{ y \mid (x,y) \in r \} \Big) \;\Big|\; \exists y : (x,y) \in r \Big\}$$

Spatial databases

Up to now, $\mathbb{U}$ was unstructured

$\Rightarrow$ Generic bulk-processing nature of database operations

However, in reality $\mathbb{U}$ does have structure

Some applications want to use this structure

E.g. *spatial databases:* $\mathbb{U}$ is $\mathbb{R}$

Set of points in $\mathbb{R}^2$ $\Rightarrow$ binary relation S

First-order queries over $\mathbb{R}$

Make predicates and operations on $\mathbb{R}$ available

Do all points in S lie on a common circle around the origin?

$$\exists r \forall x, y (S(x,y) \rightarrow x^2 + y^2 = r^2)$$

Incorrect under active-domain semantics of FO

$$\exists x_0, y_0 \forall x, y (S(x,y) \rightarrow x^2 + y^2 = x_0^2 + y_0^2)$$

$\Rightarrow$ Active-domain semantics / Natural semantics for FO

Over uninterpreted $\mathbb{U}$ easily equivalent, but over $\mathbb{R}$?

Benedikt&Libkin: For any φ there exists ψ such that

$$\mathbf{D} \models_{\text{natural}} \varphi \quad \Leftrightarrow \quad \mathbf{D} \models_{\text{active}} \psi$$

$\Rightarrow$ From now on use natural semantics

Evaluating FO queries over $\mathbb{R}$

Natural semantics can yield infinite answers to queries

What is the convex closure of S?

$$\begin{aligned}\{(x,y) \mid \exists x_1, y_1, x_2, y_2, \lambda :\ & \\ & S(x_1,y_1) \wedge S(x_2,y_2) \wedge 0 \leqslant \lambda \leqslant 1 \\ & \quad \wedge (x,y) = \lambda(x_1,y_1) + (1-\lambda)(x_2,y_2)\}\end{aligned}$$

$\Rightarrow$ *Plug-in evaluation*

E.g. $\mathbf{D}$ with $S^{\mathbf{D}} = \{(0,0),(1,1)\}$:

$$\begin{aligned}\{(x,y) \mid \exists x_1, y_1, x_2, y_2, \lambda :\ & \\ & ((x_1,y_1) = (0,0) \vee (x_1,y_1) = (1,1)) \\ & \wedge ((x_2,y_2) = (0,0) \vee (x_2,y_2) = (1,1)) \\ & \wedge 0 \leqslant \lambda \leqslant 1 \\ & \wedge (x,y) = \lambda(x_1,y_1) + (1-\lambda)(x_2,y_2)\}\end{aligned}$$

$\Rightarrow$ Symbolic representation of query answer by formula over $\mathbb{R}$

Semi-algebraic sets

Sets in $\mathbb{R}^n$ definable by formulas over $\mathbb{R}$

Quite nice properties

Tarski: The first-order theory of $\mathbb{R}$ is decidable: it effectively admits quantifier elimination

$\Rightarrow$ Symbolic representation of semi-algebraic sets using formulas is workable

- Better and better algorithms

- Number of quantifiers is database-independent

Constraint databases

Allow semi-algebraic sets not only as outputs, but also as inputs

$\Rightarrow$ Relations in database need no longer be finite; only semi-algebraic

Constraint database: store for each relation a quantifier-free formula over $\mathbb{R}$

$\surd$ Works for any interpreted universe $\mathbb{U}$ with effective q.e.

Tarski: Every semi-algebraic subset of $\mathbb{R}$ is a finite union of intervals

$\Rightarrow$ O-minimality, tame topology

Natural/active equivalence for FO holds over any o-minimal $\mathbb{U}$ with q.e.

Geometric queries

What is genericity for spatial database queries?

$\frown$ Query invariant under all permutations of $\mathbb{R}$?

Atomic data elements in a spatial database:

– real numbers

\+ points in space ($\mathbb{R}^d$)

$\smile$ Query invariant under all permutations of $\mathbb{R}^d$

Smaller groups of permutations correspond to geometrical ($\leftrightarrow$ purely logical) queries
[Felix Klein's Erlanger Programm]

Tarski: Logic as an extreme kind of geometry

Affine-generic queries

Query is *affine-generic* if invariant under all affinities

+ Is S nonempty?

+ Is S convex?

– Is S a circle?

$\Rightarrow$ Is there a logic for the affine-generic queries?

Tarski: Elementary affine geometry in $\mathbb{R}^d$ as first-order logic over $(\mathbb{R}^d, \beta)$

$\beta(p,q,r) \Leftrightarrow p$ lies on close line segment between q and r

Geometric databases

Spatial database:

Implementation level: constraint database over $(\mathbb{R}, <, +, \cdot, 0, 1)$

Geometrical level: constraint database over $(\mathbb{R}^d, \beta)$

$\Rightarrow$ First-order formula:

FO[$\mathbb{R}$]: over $(<, +, \cdot, 0, 1, \mathcal{S})$

FO[β]: over $(\beta, \mathcal{S}')$

FO[β] vs affine-generic FO[$\mathbb{R}$]

Is S nonempty?

$$\exists x, y : S(x,y)$$
$$\exists p \, S(p)$$

Is S convex?

$$\forall x_1, y_1, x_2, y_2, \lambda : (S(x_1,y_1) \wedge S(x_2,y_2) \wedge 0 \leqslant \lambda \leqslant 1) \\ \rightarrow S(\lambda(x_1,y_1) + (1-\lambda)(x_2,y_2))$$

$$\forall p, q, r : (S(p) \wedge S(q) \wedge \beta(r,p,q)) \rightarrow S(r)$$

Is S a circle? Not affine-generic, not in FO[β]

Theorem: q affine-generic and in FO[$\mathbb{R}$] $\Leftrightarrow$ q in FO[β]

Tarski: Geometric constructions of $+$ and $\times$ can be expressed in FO over β

Conclusion

Database theory relies heavily on logic

Not surprising that many of Tarki's ideas find application

$\Rightarrow$ Tarski's 100th anniversary good excuse to talk about database theory at CSL

Thanks: Janos Makowski, Dirk Van Gucht