Database Query Languages
expressiveness and complexity

Leonid Libkin

ENS, bases de données
DATABASE LANGUAGES: EXPRESSIVENESS AND COMPLEXITY

- Standard foundational language: FO (First-Order Logic)
- We study logics over finite structures
- Foundational role in the study of the theory of relational databases:
  - relational database = finite relational structure
  - query languages are logic based
- The field is also known as finite model theory (since model theory studies logics on arbitrary structures)
- Finite model theory was described as the “backbone of database theory” (Vianu, 1995)
- Connections work both ways: much of the motivation for finite model theory research came from databases.
- Many other applications in CS: verification, AI, constraint satisfaction, algorithms, complexity …
Key problems

1. **Expressiveness** of query languages
   - Limited expressiveness (e.g., relational calculus = first-order logic)
   - What is *not* expressible? When to add new language constructs?
   - Adding new constructs is not free — optimizations!

2. **Complexity** of query languages
   - Do we know the complexity of query evaluation from the logical formalism?

3. **Equivalence** of query languages
   - Can we lower the complexity by changing the syntax?
   - Important in the study of languages for XML.

4. **Satisfiability** (usually, finite satisfiability)
   - Used in: static analysis, incomplete information.
First-Order (FO)

• The most fundamental query language — first-order logic (FO)
• Database people often refer to it as relational calculus.
• The core of SQL (minus aggregation — we’ll address it later).
• A basic question: what are the limitations of FO?
• Intuition: FO cannot express:
  1. nontrivial counting properties, and
  2. queries requiring recursion
• We’ll now see some “canonical” examples of inexpressible queries.
Even cardinality

- **Active domain** = set of all elements stored in a database
- A Boolean query

\[ \text{EVEN}(D) = \text{true} \iff |\text{ActiveDomain}(D)| = 0 \pmod{2} \]
Transitive closure
Transitive closure

\[
\begin{align*}
\text{trcl}(x, y) & :\leftarrow e(x, y) \\
\text{trcl}(x, y) & :\leftarrow e(x, z), \text{trcl}(z, y)
\end{align*}
\]
Same Generation
Same Generation

\[
\begin{align*}
sg(x, x) & \quad :\quad \vdash \\
sg(x, y) & \quad :\quad e(x', x), e(y', y), sg(x', y')
\end{align*}
\]
Same Generation

\[
sg(x, x) \quad :- \\
sg(x, y) \quad :- \quad e(x', x), e(y', y), sg(x', y')
\]
A bit of history

- How to prove that these queries are not expressible in FO?
- Classical model theory offers us powerful tools, like compactness.
- They can be used to prove that \textsc{Even} is not FO-definable (also shown in the paper).
- They can also be used to prove that graph connectivity (and hence transitive closure) are not definable over arbitrary graphs.
- But the proof does not work for finite graphs: compactness fails in the finite.
- However, databases are finite!
A bit of history cont’d

• Fagin 1975: Transitive closure is not FO-expressible over finite graphs. Technique: games.
• Afterwards (1970s, 1980s): more and more advanced game proofs.
• Require nontrivial combinatorial arguments.
• A notable exception: 0-1 laws (Fagin 1976) – an easily applicable tool.
• 1990s: proper tools are being developed. They do not require complicated combinatorial proofs.
First-Order Logic (FO)

• Assumption: databases are graphs – one binary relation $E(\cdot, \cdot)$. Just to make things simple for the talk.

• Syntax of FO:
  - Atomic formulae: $E(x, y)$, $x = y$
  - Boolean combinations: $\varphi \lor \psi$, $\varphi \land \psi$, $\neg \varphi$
  - Quantification: $\exists x \varphi$, $\forall x \varphi$
  - $\varphi(\bar{x})$ means that $\bar{x}$ is the tuple of free variables of $\varphi$

• Quantifier rank $qr(\varphi)$:
  - The depth of quantifier nesting in $\varphi$.
  - Example: the quantifier rank of $\exists x (\forall y E(x, y) \lor \forall z \neg E(x, z))$ is 2 and not 3.
First-Order Logic – Examples

- There are at least $k$ elements

$$\exists x_1 \ldots \exists x_k \quad \bigwedge_{i,j \leq k, \ i \neq j} \neg (x_i = x_j)$$

- There is a path of length $k$ from $x_0$ to $x_k$

$$\exists x_1 \ldots \exists x_{k-1} \quad \bigwedge_{0 \leq i < k} E(x_i, x_{i+1})$$

- There is no cycle of length $k$

$$\neg \exists x_1 \ldots \exists x_k \ E(x_k, x_1) \land \bigwedge_{i < k} E(x_i, x_{i+1})$$
First-Order Logic – Examples

• There are at least $k$ elements

  The number of elements is even – *inexpressible*.

• There is a path of length $k$ from $x_0$ to $x_k$

  There is a path from $x_0$ to $x_k$ – *inexpressible*.

• There is no cycle of length $k$

  There is no cycle – *inexpressible*. 

Leonid Libkin

ENS, bases de données
Classical logic tools

- Recall: a theory $\Phi$ is a set of sentences.
- $\Phi$ is consistent if it has a model, i.e., a structure that satisfies each $\varphi \in \Phi$.
- It is finitely consistent if each finite subset of $\Phi$ is consistent.
- Compactness Theorem: a theory is consistent iff it is finitely consistent.
- Löwenheim-Skolem theorem: if a theory (in a finite or countable language) is consistent, it has a countable model.
- With this, we shall prove that $\text{Even}$ is not FO-definable.
Even is not in FO

- Assume \( \varphi \) defines \textit{Even}. The vocabulary is consists of one unary relation \( U(\cdot) \).

- Let \( \lambda_n \) say that there are at least \( n \) elements:

\[
\exists x_1 \ldots \exists x_n \bigwedge_{1 \leq i < j \leq n} \neg (x_i = x_j)
\]

- Two theories:

\[
\Phi_1 = \{ \varphi \} \cup \{ \lambda_i \mid i > 0 \} \quad \text{and} \quad \Phi_2 = \{ \neg \varphi \} \cup \{ \lambda_i \mid i > 0 \}
\]

- Each is consistent. Indeed each is finitely consistent. Take a finite subset of \( \Phi_1 \) and let \( n \) be the largest index of \( \lambda_n \) in it. Then the set with \( 2n \) elements is the model. For \( \Phi_2 \), take the set with \( 2n + 1 \) elements. Hence both are consistent by compactness.

- By Löwenheim-Skolem, each has a countable model, \( M_1 \) and \( M_2 \) respectively.

- But these are just sets, hence isomorphic. And yet \( M_1 \models \varphi \) and \( M_2 \models \neg \varphi \), contradiction.
Can’t we always use this?

- No, Compactness and analogs of Löwenheim-Skolem fail in the finite.
- Finite Compactness Theorem: if each finite subset of $\Phi$ has a finite model, then $\Phi$ has a finite model.
- Counterexample: $\Phi = \{\lambda_i \mid i > 0\}$.
- Each finite subset has a finite model
  - simply take the set of $n$ elements where $n$ is the largest index of $\lambda_n$
  - but $\Phi$ only has infinite models.
Ehrenfeucht-Fraïssé games

- The tool of choice for the neolithic period of finite model theory.
- Played on two databases (graphs) $\mathcal{A}$ and $\mathcal{B}$.
- Two players:
  - Spoiler (the bad guy): tries to show that $\mathcal{A}$ and $\mathcal{B}$ are different.
  - Duplicator (the good guy): tries to show that $\mathcal{A}$ and $\mathcal{B}$ are the same.
- Play for $k$ rounds.
- In each round:
  - the spoiler moves first: selects a database and an element there.
  - the duplicator responds by an element in the other database.
- The duplicator wins if at the end, the played elements form a partial isomorphism between $\mathcal{A}$ and $\mathcal{B}$. 
Ehrenfeucht-Fraïssé game - example 1

 Spoiler and Duplicator play for 3 rounds.
Ehrenfeucht-Fraïssé game - example 1

Spoiler and Duplicator play for 3 rounds.
Ehrenfeucht-Fraïssé game - example 1

Spoiler and Duplicator play for 3 rounds.
Ehrenfeucht-Fraïssé game - example 1

A

B

Spoiler and Duplicator play for 3 rounds.
Ehrenfeucht-Fraïssé game - example 1

A

B

Spoiler and Duplicator play for 3 rounds.
Ehrenfeucht-Fraïssé game - example 1

A

B

Spoiler and Duplicator play for 3 rounds.
Ehrenfeucht-Fraïssé game - example 1

**A**

**B**

Spoiler and Duplicator play for 3 rounds.
Ehrenfeucht-Fraïssé game - example 1

Spoiler and Duplicator play for 3 rounds.
Ehrenfeucht-Fraïssé game - example 1

Spoiler and Duplicator play for 3 rounds.

The duplicator wins in 3 rounds.
Ehrenfeucht-Fraïssé game - example 2

\[ \blacktriangle \triangleleft \triangleleft \triangleleft \rightarrow \] 

\[ \blacktriangle \rightarrow \triangleleft \triangleleft \triangleleft \rightarrow \]
Ehrenfeucht-Fraïssé game - example 2

A

B

\[ \rightarrow \]
Ehrenfeucht-Fraïssé game - example 2

A

B
Ehrenfeucht-Fraïssé game - example 2

\[ A \]

\[ B \]
Ehrenfeucht-Fraïssé game - example 2
Ehrenfeucht-Fraïssé game - example 2
Ehrenfeucht-Fraïssé game - example 2

A

B

Leonid Libkin  ENS, bases de données
Ehrenfeucht-Fraïssé game - example 2
Ehrenfeucht-Fraïssé game - example 2

The spoiler wins in 3 rounds.
Ehrenfeucht-Fraïssé games and FO

The duplicator has a winning strategy in the $k$-round Ehrenfeucht-Fraïssé game if he can win in $k$ rounds no matter how the spoiler plays.

**Theorem**

The duplicator has a winning strategy in the $k$-round Ehrenfeucht-Fraïssé game on $\mathcal{A}$ and $\mathcal{B}$ if and only if $\mathcal{A}$ and $\mathcal{B}$ cannot be distinguished by FO sentences of quantifier rank up to $k$. 
How to prove that a property $\mathcal{P}$ is not expressible in FO?

Find families of graphs $\mathcal{A}_k$ and $\mathcal{B}_k$, for $k \in \mathbb{N}$ so that:

1. All $\mathcal{A}_k$ have property $\mathcal{P}$;
2. None of $\mathcal{B}_k$ has property $\mathcal{P}$;
3. The duplicator has a winning strategy in the $k$-round Ehrenfeucht-Fraïssé game on $\mathcal{A}_k$ and $\mathcal{B}_k$

If $\mathcal{P}$ were expressible by a sentence $\varphi$ of quantifier rank $k$,

- $\mathcal{A}_k$ and $\mathcal{B}_k$ must agree on $\varphi$ — by 3,
- but by 1 and 2 they disagree on $\varphi$. 
A surprisingly powerful example

Let $L_n$ be a linear ordering of length $n$.

**Theorem**

If $m, n \geq 2^k$, then the duplicator has a winning strategy in the $k$-round Ehrenfeucht-Fraïssé game on $L_n$ and $L_m$. 
A surprisingly powerful example

Let $L_n$ be a linear ordering of length $n$.

**Theorem**

If $m, n \geq 2^k$, then the duplicator has a winning strategy in the $k$-round Ehrenfeucht-Fraïssé game on $L_n$ and $L_m$.

**Corollary**

Query $\text{EVEN}$ is not expressible over linear orderings.

Because: take $\mathcal{A}_k$ to be $L_{2^k+1}$ and $\mathcal{B}_k$ to be $L_{2^k}$.
The result about **EVEN** shows that none of the following is definable in FO: Graph connectivity; graph acyclicity; transitive closure.
The early 1980s: the era of tricks

The result about \textbf{Even} shows that none of the following is definable in FO: Graph connectivity; graph acyclicity; transitive closure.

Edges: to the 2nd successor, modulo the length of the chain.
The early 1980s: the era of tricks

The result about **EVEN** shows that none of the following is definable in FO: Graph connectivity; graph acyclicity; transitive closure.

**Edges:** to the 2nd successor, modulo the length of the chain.
The early 1980s: the era of tricks

The result about \textbf{EVEN} shows that none of the following is definable in FO: Graph connectivity; graph acyclicity; transitive closure.

Edges: to the 2nd successor, modulo the length of the chain.
The early 1980s: the era of tricks

The result about **EVEN** shows that none of the following is definable in FO: Graph connectivity; graph acyclicity; transitive closure.

**Edges:** to the 2nd successor, modulo the length of the chain.
The early 1980s: the era of tricks

The result about \textbf{EVEN} shows that none of the following is definable in FO: Graph connectivity; graph acyclicity; transitive closure.

\begin{tikzpicture}
\draw[dashed,->] (0,-1) -- (1,-1);
\draw[dashed,->] (1,0) -- (2,0);
\draw[dashed,->] (2,-1) -- (3,-1);
\draw[dashed,->] (3,0) -- (4,0);
\draw[dashed,->] (4,-1) -- (5,-1);
\draw[dashed,->] (5,0) -- (6,0);
\draw[dashed,->] (6,-1) -- (7,-1);
\draw[dashed,->] (7,0) -- (8,0);
\end{tikzpicture}

\textbf{Edges:} to the 2nd successor, modulo the length of the chain.
The early 1980s: the era of tricks

The result about \textbf{EVEN} shows that none of the following is definable in FO: Graph connectivity; graph acyclicity; transitive closure.

\textbf{Edges}: to the 2nd successor, modulo the length of the chain.
The early 1980s: the era of tricks

The result about \textbf{EVEN} shows that none of the following is definable in FO: Graph connectivity; graph acyclicity; transitive closure.

Edges: to the 2nd successor, modulo the length of the chain.
The early 1980s: the era of tricks

The result about **EVEN** shows that none of the following is definable in FO: Graph connectivity; graph acyclicity; transitive closure.

**Edges:** to the 2nd successor, modulo the length of the chain.
The early 1980s: the era of tricks

The result about \textbf{Even} shows that none of the following is definable in FO: Graph connectivity; graph acyclicity; transitive closure.

\textbf{Edges}: to the 2nd successor, modulo the length of the chain.
The early 1980s: the era of tricks

The result about **EVEN** shows that none of the following is definable in FO: 
Graph connectivity; graph acyclicity; transitive closure.

**Edges**: to the 2nd successor, modulo the length of the chain.
The early 1980s: the era of tricks

The result about \textbf{EVEN} shows that none of the following is definable in FO: Graph connectivity; graph acyclicity; transitive closure.

\textbf{Edges}: to the 2nd successor, modulo the length of the chain.
The early 1980s: the era of tricks

The result about **EVEN** shows that none of the following is definable in FO: **Graph connectivity; graph acyclicity; transitive closure.**

Edges: to the 2nd successor, modulo the length of the chain.
The early 1980s: the era of tricks

The result about \texttt{EVEN} shows that none of the following is definable in FO: Graph connectivity; graph acyclicity; transitive closure.

Connected if \texttt{EVEN} is false; disconnected if \texttt{EVEN} is true.
The era of tricks cont’d

**Acyclicity** is not FO-expressible:
- a similar trick – with one backedge instead of two.

**Transitive closure** is not FO-expressible:
- the symmetric-transitive closure of $G$ is a complete graph iff $G$ is connected.
The 1990s: the era of tools

- The iron age of FMT.
- For more complicated problems, stone (pebble) tools – games – become very hard to use.
- More and more complicated winning conditions are used:
  - Fagin, Ajtai, Vardi, Stockmeyer, Schwentick, Kolaitis, Väänänen, etc
- Fagin, Stockmeyer, Vardi, 1993: Let’s build a library of winning strategies for the duplicator.
- Key idea: locality.
  - Already present in earlier work by Gaifman 1980 and Hanf 1965.
Transitive closure revisited

Degrees of nodes: 0, 1
Transitive closure revisited

Degrees of nodes: $0, 1, \ldots, n$ – depends on the input.
Transitive closure revisited

Degrees of nodes: 0, 1, \ldots, n – depends on the input.

This cannot happen for FO queries!
A useful property: BNDP

- A query $Q$ from graphs to graphs has the Bounded Number of Degrees Property if there is a function $f_Q : \mathbb{N} \rightarrow \mathbb{N}$ such that:

  all degrees in $G$ are bounded by $k$
  \[ \Downarrow \]
  the number of different degrees in $Q(G)$ is at most $f_Q(k)$

- We’ve just seen that transitive closure violates the BNDP.
A useful property: BNDP

- A query $Q$ from graphs to graphs has the **Bounded Number of Degrees Property** if there is a function $f_Q : \mathbb{N} \rightarrow \mathbb{N}$ such that:

  \[
  \begin{align*}
  &\text{all degrees in } G \text{ are bounded by } k \\
  &\Rightarrow \\
  &\text{the number of different degrees in } Q(G) \text{ is at most } f_Q(k)
  \end{align*}
  \]

- We’ve just seen that transitive closure violates the BNDP.

**Theorem**

*Every FO query has the BNDP.*

- **Corollary:** transitive closure is not FO-definable.
Another application of BNDP – Same Generation

Degrees: 0, 1, 2
Another application of BNDP – Same Generation
Another application of BNDP – Same Generation

Degrees: $1, 2, 4, 8, \ldots, 2^{\text{depth}(G) - 1}$  
Number of degrees: $\text{depth}(G)$
Another application of BNDP – Same Generation

Violates the BNDP — Hence same-generation is not FO-definable
What makes the BNDP work?

- Locality of FO.
- There are two tools based on locality:
  1. Gaifman-locality (Gaifman 1982)
- Key concept: neighborhood.
- A neighborhood of radius $r$ of $\bar{a}$ in a graph $G$ is denoted by $N_r^G(\bar{a})$.
- It is the subgraph induced by all the nodes of distance $\leq r$ from one of the nodes in $\bar{a}$.
- Nodes $\bar{a}$ are distinguished:
  - if we have an isomorphism $h : N_r^G(a_1, \ldots, a_n) \rightarrow N_r^{G'}(b_1, \ldots, b_n)$
    then $h(a_1) = b_1$, $\ldots$, $h(a_n) = b_n$. 
Gaifman-locality

Theorem (Gaifman 1982, his bound was $7^k$ then improved to $2^k$ in 1998)

For every FO formula $\varphi(\bar{x})$ of quantifier rank $k$ and for every graph $G$:

$N^G_{2^k}(\bar{a})$ and $N^G_{2^k}(\bar{b})$ are isomorphic $\Rightarrow$ $G \models \varphi(\bar{a}) \iff \varphi(\bar{b})$
**Gaifman-locality**

**Theorem (Gaifman 1982, his bound was $7^k$ then improved to $2^k$ in 1998)**

For every FO formula $\varphi(\bar{x})$ of quantifier rank $k$ and for every graph $G$:

$$N_{2^k}^G(\bar{a}) \text{ and } N_{2^k}^G(\bar{b}) \text{ are isomorphic} \implies G \models \varphi(\bar{a}) \leftrightarrow \varphi(\bar{b})$$

**Application:** Transitive closure is not definable in FO.

---

Leonid Libkin  
ENS, bases de données  
28/43
Gaifman-locality

Theorem (Gaifman 1982, his bound was $7^k$ then improved to $2^k$ in 1998)

For every FO formula $\varphi(\bar{x})$ of quantifier rank $k$ and for every graph $G$:

\[
N_{2^k}^G(\bar{a}) \text{ and } N_{2^k}^G(\bar{b}) \text{ are isomorphic } \Rightarrow \ G \models \varphi(\bar{a}) \leftrightarrow \varphi(\bar{b})
\]

Application: Transitive closure is not definable in FO.

If $\varphi(x, y)$ is of quantifier rank $k$ and $r = 2^k$ then both $\varphi(a, b)$ and $\varphi(b, a)$ are true, or both are false.
Hanf-locality

- Write $G \leftrightarrow_r G'$ if there exists a bijection $f : G \to G'$ such that $N_r^G(a)$ and $N_r^{G'}(f(a))$ are isomorphic for every node $a$.
- Locally two graphs look the same, up to a bijection $f$. 
Hanf-locality

- Write $G \leftrightarrow_r G'$ if there exists a bijection $f : G \to G'$ such that $N_r^G(a)$ and $N_r^{G'}(f(a))$ are isomorphic for every node $a$.
- Locally two graphs look the same, up to a bijection $f$.

**Theorem (Fagin, Stockmeyer, Vardi, ’93, their bound was $3^k$)**

For every FO sentence $\varphi$ of quantifier rank $k$,

$$
\text{if } G \leftrightarrow_{2^k} G' \text{ then } G \models \varphi \iff G' \models \varphi
$$

- Can be extended to arbitrary queries, but most often this notion is used for Boolean queries.
Hanf-locality: application

• If $m > 2r + 1$ then $G \equiv_r G'$ (all $r$-neighborhoods are the same).
• Hence no FO sentence $\varphi$ defines connectivity: as long as $m > 2^{qr(\varphi)} + 1$, graphs $G$ and $G'$ cannot be distinguished by $\varphi$. 
Summary: locality notions

A query $Q$ is:

- **Hanf-local** if there is $r \geq 0$ so that $G \cong_r G'$ implies $Q(G) = Q(G')$.
  - for Boolean queries; a natural extension to non-Boolean queries exists.

- **Gaifman-local** if there is $r \geq 0$ such that if $N_r^G(\bar{a})$ and $N_r^G(\bar{b})$ are isomorphic, then $\bar{a} \in Q(G) \iff \bar{b} \in Q(G)$.

- has the **BNDP** if there is function $f_Q : \mathbb{N} \to \mathbb{N}$ so that for $G$ with all degrees $\leq k$, the number of different degrees in $Q(G)$ is $\leq f_Q(k)$.

**Theorem**

$\text{Hanf-local} \Rightarrow \text{Gaifman-local} \Rightarrow \text{BNDP}.$

**Corollary**

$\text{FO queries are Hanf-local, Gaifman-local, and have the BNDP.}$
Counting: towards 0-1 laws

- How to prove that nontrivial counting properties are not expressible?
- **Even**: roughly half of databases have the property, and half don’t.
- FO cannot exhibit such a behavior.
Counting: towards 0-1 laws

• How to prove that nontrivial counting properties are not expressible?
• **Even**: roughly half of databases have the property, and half don’t.
• FO cannot exhibit such a behavior.

• Pick a database “at random”.
• Check if it satisfies a property \( P \).
• What’s the probability of that?
• If \( P \) is FO-definable, it is 0 or 1: 0-1 law.
• Need to formalize: ‘pick a database at random’.
Towards 0-1 laws

- For each $n$ look at graphs with nodes $1, \ldots, n$.
- For a property $\mathcal{P}$, let
  $$\mu_n(\mathcal{P}) = \frac{|\{\text{graphs on } 1, \ldots, n \text{ that satisfy } \mathcal{P}\}|}{|\{\text{graphs on } 1, \ldots, n\}|}$$
- Proportion of graphs on $1, \ldots, n$ satisfy $\mathcal{P}$, or
- Probability that a randomly picked graph on $1, \ldots, n$ — with respect to the uniform distribution — satisfies $\mathcal{P}$.
- Asymptotic probabilities:
  $$\mu(\mathcal{P}) = \lim_{n \to \infty} \mu_n(\mathcal{P})$$
Asymptotic probabilities: examples

- \( \mu(\text{EVEN}) \) — does not exist: 
  \[ \mu_n(\text{EVEN}) = \begin{cases} 
  1, & \text{if } n \text{ is even} \\
  0, & \text{if } n \text{ is odd} 
\end{cases} \]

- \( \mu(\text{exists isolated node}) = 0 \).
  \( \exists x \forall y \neg E(x, y) \)

- \( \mu(\text{diameter} \leq 2) = 1 \).
  \( \forall x \forall y \exists z \ E(x, z) \land E(y, z) \)

- \( \mu(\text{graph is connected}) = 1 \).

- Two sets \( A \) and \( B \) with \( B \subseteq A \).
  \( \text{Parity} \) is true iff \( |B| \) is even.
  \( \mu(\text{Parity}) = \frac{1}{2} \).
0-1 law

Theorem (Fagin 1976)

If $\mathcal{P}$ is FO-definable, then $\mu(\mathcal{P})$ exists and equals 0 or 1.
0-1 law

Theorem (Fagin 1976)
If $\mathcal{P}$ is FO-definable, then $\mu(\mathcal{P})$ exists and equals 0 or 1.

- If you like truly beautiful proofs, this is the one for you!
- Immediate corollaries: **EVEN** and **Parity** are not FO-definable.
- Warning: the result does not hold when we consider specific classes of structures.
- For example, 0-1 law fails over ordered graphs:
  - $\mu(\text{there is an edge between the first and the last element}) = \frac{1}{2}$. 
0-1 law: proof idea

• Assume the vocabulary of one unary relation $U(\cdot)$. Take a sentence $\varphi$.

• Let $\lambda_n$ say that there are at least $n$ elements:

$$\exists x_1 \ldots \exists x_n \bigwedge_{1 \leq i < j \leq n} \neg (x_i = x_j)$$

• Take $\Phi = \{ \lambda_i \mid i > 0 \}$

• Claim: either $\varphi$ or $\neg \varphi$ is a logical consequence of $\Phi$.
  • Assume not; then both $\Phi \cup \{ \varphi \}$ and $\Phi \cup \{ \neg \varphi \}$ are consistent
  • Both have countable models by Löwenheim-Skolem
  • These models are isomorphic
  • Cannot happen since $\varphi$ is true in one but not in the other
0-1 law: proof idea continued

- Assume $\varphi$ is a logical consequence of $\Phi$.
- Then $\Phi$ is a logical consequence of a finite subset $\Phi' \subset \Phi$.
  - By compactness (a small exercise)
- Let $n$ be the highest index of $\lambda_n$ in $\Phi'$.
- Then as soon as $|U| \geq n$, sentence $\varphi$ is true.
- Hence $\mu(\varphi) = 1$.
- If $\neg \varphi$ is a logical consequence of $\Phi$, the same proof gives us $\mu(\neg \varphi) = 1$ and $\mu(\varphi) = 0$. 
0-1 law: general idea

- Look at graphs, sentences in the vocabulary $E(\cdot, \cdot)$.
- **Extension Axiom** $EA_{n,m}$: for two disjoint sets of nodes $X$ and $Y$ of cardinalities $n$ and $m$, there is a node connected to every mode in $X$ and to no node in $Y$.
- **Probability bit**: $\mu(EA_{n,m}) = 1$
  - try to construct a model of $EA_{3,3}$ or $EA_{4,4}$; not so simple.
  - but eventually almost all graphs are such.
  - Also note that $EA_{1,1}$ ensures connectivity.
- Take $EA = \{EA_{n,m} \mid n, m > 0\}$
- **Combinatorics bit**: up to isomorphism, there is one countable model $RG$ of $EA$.
  - it is called the random graph
- **Logic bit** (using compactness): if $RG \models \varphi$ then $\mu(\varphi) = 1$ and if $RG \not\models \varphi$ then $\mu(\varphi) = 0$. 
A nice exercise

- Consider two infinite undirected graphs whose set of nodes is \{1, 2, 3, 4, \ldots\}
- In $G_1$, there is an edge between $n$ and $m$, when $n < m$, if the $n$th prime divides $m$
  - e.g., there is an edge between 3 and 25
- In $G_2$, there is an edge between $n$ and $m$ is the $n$th bit in the binary expansion of $m$ is set to 1.
  - e.g., there are edges between 1,2,3,4 and 15

- Prove that $G_1$ and $G_2$ are isomorphic.
- Hint: use the (proof of) 0–1 law.
Descriptive complexity

- Machine-independent characterization of complexity classes.
- A query language tells you a lot about the complexity.
- If your logic captures a complexity class, you have even more information:
  - complexity cannot be lowered.
- First result:

  **Theorem (Fagin 1974)**

  \[
  \text{NP = Existential Second-Order Logic (ESO)}
  \]

  - ESO = \( \exists R_1 \ldots \exists R_k \ \varphi(R_1, \ldots, R_k, E) \)
  - 3-colorability: \( \exists R \exists G \exists B \ \varphi \)
    - \( \varphi \) says that \( R, G, B \) partition the set of nodes and endpoints of an edge cannot be in the same set.
Descriptive complexity – other classes

- FO is contained in $AC^0$
  - $AC^0$ – constant parallel time: constant time with polynomially many processors. Suggests very efficient parallel algorithms.
  - Complexity of the relational calculus.
  - Uniform version of $AC^0$ is captured by FO with ‘$<, +, \times$’ on the finite universe.
- Basic SQL (FO+simple arithmetic+aggregation) is contained in $TC^0$
  - $TC^0$ – constant parallel time with more complex gates (including majority gates). Probably a small subset of DLOGSPACE but not yet separated from NP!
- FO+transitive closure is contained in $NLOGSPACE$.
- FO+fixed-points (including Datalog) is contained in $PTIME$
  - Over ordered databases, captures $PTIME$. 
Consider decision problems (yes/no answers)

- **Combined complexity**: Input $D, \varphi$, question: $D \models \varphi$?
- **Data complexity**: Fixed $\varphi$, Input $D$, question: $D \models \varphi$?

Summary: Data complexity is very low, $\text{AC}^0$.

- Combined complexity is high: $\text{PSPACE}$-complete.
Complexity of Conjunctive queries

- Recall: those are essentially joins
- Queries in the select-project-Cartesian product fragment of relational algebra
- In logic: $\exists, \land$ in first-order logic
- For example, there is a path of length 3:
  \[
  \exists x_1, x_2, x_3, x_4 \ E(x_1, x_2) \land E(x_2, x_3) \land E(x_3, x_4)
  \]

- Summary: Data complexity still the same: $AC^0$.
- Combined complexity is not as high: NP-complete.
- Idea: guess valuations for existential quantifiers; check if they work.