View-based query processing

Nadime Francis

Université Paris-Est Marne-la-Vallée
nadime.francis@u-pem.fr

EPIT 2019
Thursday, April, 11th
Introduction
View-based query processing

General setting
View-based query processing

General setting
View-based query processing
General setting
View-based query processing

General setting
View-based query processing

General setting
View-based query processing

General setting
View-based query processing

Scenario: query optimization and caching
View-based query processing

Scenario: data leak prevention

Secure database

Public queries

Private queries

Integrated answers
View-based query processing

Scenario: virtual data integration
Views
Let $\sigma$ and $\tau$ be two database schemas.

- **View definition** (or simply view): A view definition $V$ from $\sigma$ to $\tau$ is a set of queries over $\sigma$ indexed by $\tau$:

$$V = \{ Q_b \mid b \in \tau \}$$

such that:

$$\forall b \in \tau, \quad \text{arity}(b) = \text{arity}(Q_b)$$

- **View instance**: A view instance $E$ is a database over $\tau$.

- **View image**: Given a database $D$ over $\sigma$ and a view $V$ from $\sigma$ to $\tau$, the view image of $D$, $V(D)$, is a view instance such that:

$$\forall b \in \tau, \bar{x} \in D, \quad \bar{x} \in b(V(D)) \iff \bar{x} \in Q_b(D)$$
Example: view definition, view image, view instance

\[ \sigma = \{ b, f, w \} \]

Database \( D \) over \( \sigma \)
Example: view definition, view image, view instance

\[ \sigma = \{ b, f, w \} \]

\[ \tau = \{ g, s \} \]

Database \( D \) over \( \sigma \)
Example: view definition, view image, view instance

\[ \sigma = \{ b, f, w \} \quad \mathbf{V} = \left\{ \begin{array}{c}
Q_g(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \\
Q_s(x, y) = x \xleftarrow{f} z \xrightarrow{f} y \land x \neq y
\end{array} \right\} \quad \tau = \{ g, s \} \]

Database \( D \) over \( \sigma \)
Example: view definition, view image, view instance

\[ \sigma = \{ b, f, w \} \quad \mathbf{V} = \left\{ \begin{array}{l}
Q_g(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \\
Q_s(x, y) = x \xleftarrow{f} z \xrightarrow{f} y \land x \neq y
\end{array} \right\} \quad \tau = \{ g, s \} \]

Database \( D \) over \( \sigma \)
Example: view definition, view image, view instance

\[ \sigma = \{b, f, w\} \]

\[ \mathbf{V} = \left\{ \begin{array}{l}
Q_g(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \\
Q_s(x, y) = x \xleftarrow{f} z \xrightarrow{f} y \land x \neq y
\end{array} \right\} \]

\[ \tau = \{g, s\} \]

Database \( D \) over \( \sigma \)
Example: view definition, view image, view instance

\[ \sigma = \{b, f, w\} \quad \mathbf{V} = \left\{ \begin{array}{l} Q_g(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \\ Q_s(x, y) = x \xleftarrow{f} z \xrightarrow{f} y \land x \neq y \end{array} \right\} \quad \tau = \{g, s\} \]

Database \( D \) over \( \sigma \)
Example: view definition, view image, view instance

\[
\sigma = \{b, f, w\} \quad \mathbf{V} = \left\{ \begin{array}{l}
Q_g(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \\
Q_s(x, y) = x \xleftarrow{f} z \xrightarrow{f} y \land x \neq y
\end{array} \right\} \quad \tau = \{g, s\}
\]

Database \(D\) over \(\sigma\)
Example: view definition, view image, view instance

$$\sigma = \{b, f, w\} \quad \mathbf{V} = \left\{ \begin{array}{l} Q_g(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \\ Q_s(x, y) = x \xleftarrow{f} z \xrightarrow{f} y \land x \neq y \end{array} \right\} \quad \tau = \{g, s\}$$
Example: view definition, view image, view instance

\[ \sigma = \{ b, f, w \} \]

\[ \mathbf{V} = \left\{ \begin{array}{l}
Q_g(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \\
Q_s(x, y) = x \xleftarrow{f} z \xrightarrow{f} y \land x \neq y
\end{array} \right\} \]

\[ \tau = \{ g, s \} \]

View instance \( E \) over \( \tau \)
Example: view definition, view image, view instance

\[ \sigma = \{ b, f, w \} \quad \mathbf{V} = \begin{cases} Q_g(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \\ Q_s(x, y) = x \xleftarrow{f} z \xrightarrow{f} y \land x \neq y \end{cases} \quad \tau = \{ g, s \} \]

Is \( E \) the image of some database \( D \) through \( \mathbf{V} \)?

View instance \( E \) over \( \tau \)
Example: view definition, view image, view instance

\[ \sigma = \{ b, f, w \} \]

\[ V = \left\{ \begin{array}{l}
Q_g(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \\
Q_s(x, y) = x \xleftarrow{f} z \xrightarrow{f} y \land x \neq y
\end{array} \right\} \]

\[ \tau = \{ g, s \} \]

Is \( E \) the image of some database \( D \) through \( V \)?

View instance \( E \) over \( \tau \)
Example: view definition, view image, view instance

\[ \sigma = \{ b, f, w \} \quad \mathbf{V} = \left\{ \begin{array}{l} Q_g(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \\ Q_s(x, y) = x \xleftarrow{f} z \xrightarrow{f} y \land x \neq y \end{array} \right\} \quad \tau = \{ g, s \} \]

Is \( E \) the image of some database \( D \) through \( \mathbf{V} \)?

View instance \( E \) over \( \tau \)
Example: view definition, view image, view instance

\[\sigma = \{b, f, w\}\]

\[\mathbf{V} = \left\{\begin{array}{l}
Q_g(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \\
Q_s(x, y) = x \xleftarrow{f} z \xrightarrow{f} y \land x \neq y
\end{array}\right\}\]

\[\tau = \{g, s\}\]

Is \(E\) the image of some database \(D\) through \(\mathbf{V}\)?

View instance \(E\) over \(\tau\)
Example: view definition, view image, view instance

\[ \sigma = \{b, f, w\} \quad \mathbf{V} = \left\{ \begin{array}{l}
Q_g(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \\
Q_s(x, y) = x \xleftarrow{f} z \xrightarrow{f} y \land x \neq y
\end{array} \right\} \quad \tau = \{g, s\} \]

Is \( E \) the image of some database \( D \) through \( \mathbf{V} \)?

View instance \( E \) over \( \tau \)
What happened there?

- **Myth 1:**

- **Myth 2:**
What happened there?

- Myth 1:

  - because Aphrodite is the daughter of Uranus
  - And so is Cronus

- Myth 2:
What happened there?

- Myth 1:

  ![Image of Aphrodite and Cronus](image1.png)

  (because Aphrodite is the daughter of Uranus
   And so is Cronus)

- Myth 2:

  ![Image of Rhea and Aphrodite](image2.png)

  (because Aphrodite is the daughter of Zeus
   And Zeus is the son of Rhea)
What happened there?

- Myth 1:

  ![Diagram](image1)

  (because Aphrodite is the daughter of Uranus
   And so is Cronus)

- Myth 2:

  ![Diagram](image2)

  (because Aphrodite is the daughter of Zeus
   And Zeus is the son of Rhea)

We failed virtual data integration → the two myths are incompatible
### Problem: View Consistency

**Input**: A view $V$ from $\sigma$ to $\tau$, a view instance $E$

**Question**: Is there some $D$ over $\sigma$ such that $V(D) = E$?
Problem: View consistency for language $\mathcal{L}$
Input: An $\mathcal{L}$-view $V$ from $\sigma$ to $\tau$, a view instance $E$
Question: Is there some $D$ over $\sigma$ such that $V(D) = E$?
Testing view consistency

- **Combined** complexity:

<table>
<thead>
<tr>
<th><strong>Problem</strong></th>
<th>View consistency for language $\mathcal{L}$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input</strong></td>
<td>An $\mathcal{L}$-view $\mathbf{V}$ from $\sigma$ to $\tau$, a view instance $E$</td>
</tr>
<tr>
<td><strong>Question</strong></td>
<td>Is there some $D$ over $\sigma$ such that $\mathbf{V}(D) = E$?</td>
</tr>
</tbody>
</table>

- **Data** complexity:

  Let $\mathbf{V}$ be a fixed view from $\sigma$ to $\tau$ in some language $\mathcal{L}$:

<table>
<thead>
<tr>
<th><strong>Problem</strong></th>
<th>View consistency($\mathbf{V}$)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input</strong></td>
<td>A view instance $E$</td>
</tr>
<tr>
<td><strong>Question</strong></td>
<td>Is there some $D$ over $\sigma$ such that $\mathbf{V}(D) = E$?</td>
</tr>
</tbody>
</table>
Example: how hard is testing consistency?

\[ \sigma = \{c, e, p\} \quad \mathbf{v} = \{ \} \]
Example: how hard is testing consistency?

\[ \sigma = \{c, e, p\} \]

\[ V = \{ \]

\[ Q_{\text{edge}}(x, y) = e(x, y) \]

\[ \} \]
Example: how hard is testing consistency?

\[ \sigma = \{c, e, p\} \]

\[ V = \begin{cases} 
Q_{\text{edge}}(x, y) = e(x, y) \\
Q_{\text{palette}}(x) = p(x) 
\end{cases} \]
Example: how hard is testing consistency?

\[ \sigma = \{ c, e, p \} \]

\[ V = \{ \]

\[ Q_{\text{edge}}(x, y) = e(x, y) \]

\[ Q_{\text{palette}}(x) = p(x) \]

\[ Q_{\text{color}}(x) = \exists z \cdot p(z) \land c(x, z) \]

\[ \} \]

The view instance is consistent iff the graph is 3-colorable!
Example: how hard is testing consistency?

\[ \sigma = \{c, e, p\} \]

\[ V = \begin{cases} 
Q_{\text{edge}}(x, y) &= e(x, y) \\
Q_{\text{palette}}(x) &= p(x) \\
Q_{\text{color}}(x) &= \exists z \cdot p(z) \land c(x, z) 
\end{cases} \]
Example: how hard is testing consistency?

$\sigma = \{c, e, p\}$

$\mathbf{V} = \{ Q_{\text{edge}}(x, y) = e(x, y) \\
Q_{\text{palette}}(x) = p(x) \\
Q_{\text{color}}(x) = \exists z \cdot p(z) \land c(x, z) \}$

$Q_{\text{error}}(x, y) = \exists z \cdot c(x, z) \land c(y, z) \land e(x, y)$
Example: how hard is testing consistency?

\[ \sigma = \{ c, e, p \} \]

\[ V = \{ \]
\[ Q_{\text{edge}}(x, y) = e(x, y) \]
\[ Q_{\text{palette}}(x) = p(x) \]
\[ Q_{\text{color}}(x) = \exists z \cdot p(z) \land c(x, z) \]
\[ \} \]
Example: how hard is testing consistency?

\[ \sigma = \{ c, e, p \} \]

\[ \mathbf{V} = \begin{cases} 
Q_{\text{edge}}(x, y) = e(x, y) \\
Q_{\text{palette}}(x) = p(x) \\
Q_{\text{color}}(x) = \exists z \cdot p(z) \land c(x, z) 
\end{cases} \]
Example: how hard is testing consistency?

\[ \sigma = \{ c, e, p \} \]

\[ \mathbf{V} = \left\{ \begin{array}{l}
Q_{\text{edge}}(x, y) = e(x, y) \\
Q_{\text{palette}}(x) = p(x) \\
Q_{\text{color}}(x) = \exists z \cdot p(z) \land c(x, z) \\
\end{array} \right\} \]
Example: how hard is testing consistency?

$$\sigma = \{c, e, p\}$$

$$V = \begin{cases} 
Q_{\text{edge}}(x, y) = e(x, y) \\
Q_{\text{palette}}(x) = p(x) \\
Q_{\text{color}}(x) = \exists z \cdot p(z) \land c(x, z) \\
Q_{\text{error}}(x, y) = \exists z \cdot c(x, z) \land c(y, z) \land e(x, y) 
\end{cases}$$
Example: how hard is testing consistency?

\[ \sigma = \{c, e, p\} \]

\[ V = \{ \]
\[ Q_{\text{edge}}(x,y) = e(x,y) \]
\[ Q_{\text{palette}}(x) = p(x) \]
\[ Q_{\text{color}}(x) = \exists z \cdot p(z) \land c(x,z) \]
\[ Q_{\text{error}}(x,y) = \exists z \cdot c(x,z) \land c(y,z) \land e(x,y) \]
\[ \} \]

\[ Q_{\text{error}} \] should show if there is a coloring error... but it is empty.
Example: how hard is testing consistency?

$$\sigma = \{c, e, p\}$$

$$\mathbf{V} = \begin{cases} 
Q_{\text{edge}}(x, y) = e(x, y) \\
Q_{\text{palette}}(x) = p(x) \\
Q_{\text{color}}(x) = \exists z \cdot p(z) \land c(x, z) \\
Q_{\text{error}}(x, y) = \exists z \cdot c(x, z) \land c(y, z) \land e(x, y) 
\end{cases}$$

$Q_{\text{error}}$ should show if there is a coloring error... but it is empty.

The view instance is consistent iff the graph is 3-colorable!
How hard is testing consistency?

**Short answer:** it’s hard, even for simple languages and in data complexity.
Short answer: it’s hard, even for simple languages and in data complexity.

Theorem [Abiteboul, Duschka]

There exists a fixed view $V$ defined using conjunctive queries such that $\text{View Consistency}(V)$ is NP-complete.

Our proof sketch easily extends to UCQ, RPQ, CRPQ...
How hard is testing consistency?

Short answer: it’s hard, even for simple languages and in data complexity.

**Theorem** [Abiteboul, Duschka]

There exists a fixed view $V$ defined using **conjunctive** queries such that $\text{VIEW CONSISTENCY}(V)$ is NP-complete.

*Our proof sketch easily extends to UCQ, RPQ, CRPQ...*

The problem quickly becomes **undecidable** for more expressive languages.

**Theorem** [Abiteboul, Duschka]

There exists a fixed view $V$ defined using **Datalog** queries such that $\text{VIEW CONSISTENCY}(V)$ is undecidable.

*This also holds for context-free path queries, first-order queries...*
Certain answers
Example: answering queries using views

\[
\sigma = \{ b, f, w \} \quad \mathcal{V} = \left\{ \begin{array}{l}
Q_g(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \\
Q_f(x, y) = x \xrightarrow{f} y
\end{array} \right\} \quad \tau = \{ f, g \}
\]
Example: answering queries using views

\[ \sigma = \{b, f, w\} \]

\[ \mathcal{V} = \left\{ \begin{array}{l}
Q_g(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \\
Q_f(x, y) = x \xrightarrow{f} y
\end{array} \right\} \]

\[ \tau = \{f, g\} \]

Can we answer \( Q(x, y) = w(x, y) \) based on the view instance?
Example: answering queries using views

\[ \sigma = \{ b, f, w \} \quad V = \left\{ \begin{array}{l}
Q_g(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \\
Q_f(x, y) = x \xrightarrow{f} y
\end{array} \right\} \quad \tau = \{ f, g \} \]

Can we answer \( Q(x, y) = w(x, y) \) based on the view instance?

- \( w(Hera, Zeus) \)?
Example: answering queries using views

\[ \sigma = \{b, f, w\} \]
\[ V = \left\{ \begin{array}{l}
Q_g(x, y) = x \overset{w}{\rightarrow} z \overset{f}{\rightarrow} z' \overset{f}{\rightarrow} y \\
Q_f(x, y) = x \overset{f}{\rightarrow} y
\end{array} \right\} \]
\[ \tau = \{f, g\} \]

Can we answer \( Q(x, y) = w(x, y) \) based on the view instance?

- \( w(\text{Hera}, \text{Zeus})? \) → no clue Hera even exists...
Example: answering queries using views

\[ \sigma = \{b, f, w\} \quad \mathbf{V} = \left\{ \begin{array}{c} Q_g(x, y) = x \overset{w}{\rightarrow} z \overset{f}{\rightarrow} z' \overset{f}{\rightarrow} y \\ Q_f(x, y) = x \overset{f}{\rightarrow} y \end{array} \right\} \quad \tau = \{f, g\} \]

Can we answer \( Q(x, y) = w(x, y) \) based on the view instance?

- \( w(\text{Hera, Zeus})? \) → no clue Hera even exists...
- \( w(\text{Athena, Ares})? \)
Example: answering queries using views

\[ \sigma = \{ b, f, w \} \]
\[ \mathbf{V} = \left\{ 
\begin{array}{l}
Q_g(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \\
Q_f(x, y) = x \xrightarrow{f} y
\end{array}
\right\} \]
\[ \tau = \{ f, g \} \]

Can we answer \( Q(x, y) = w(x, y) \) based on the view instance?

- \( w(\text{Hera}, \text{Zeus})? \) → no clue Hera even exists...
- \( w(\text{Athena}, \text{Ares})? \) → possible, but no guarantee...
Example: answering queries using views

\[ \sigma = \{ b, f, w \} \quad \mathbf{V} = \left\{ \begin{array}{l}
Q_g(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \\
Q_f(x, y) = x \xrightarrow{f} y 
\end{array} \right\} \quad \tau = \{ f, g \} \]

Can we answer \( Q(x, y) = w(x, y) \) based on the view instance?

- \( w(\text{Hera, Zeus})? \rightarrow \text{no clue Hera even exists...} \)
- \( w(\text{Athena, Ares})? \rightarrow \text{possible, but no guarantee...} \)
- \( w(\text{Rhea, Cronus})? \)
Example: answering queries using views

\[
\sigma = \{ b, f, w \} \quad \mathbf{V} = \left\{ \begin{array}{l}
Q_g(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \\
Q_f(x, y) = x \xrightarrow{f} y
\end{array} \right\} \quad \tau = \{ f, g \}
\]

Can we answer \( Q(x, y) = w(x, y) \) based on the view instance?

- \( w(\text{Hera}, \text{Zeus})? \) → no clue Hera even exists...
- \( w(\text{Athena}, \text{Ares})? \) → possible, but no guarantee...
- \( w(\text{Rhea}, \text{Cronus})? \) → she has to be the wife of some grandfather
Example: answering queries using views

\[ \sigma = \{b, f, w\} \quad \mathbf{V} = \left\{ \begin{array}{l} Q_g(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \\ Q_f(x, y) = x \xrightarrow{f} y \end{array} \right\} \quad \tau = \{f, g\} \]

Can we answer \( Q(x, y) = w(x, y) \) based on the view instance?

- \( w(\text{Hera}, \text{Zeus})? \) → no clue Hera even exists...
- \( w(\text{Athena}, \text{Ares})? \) → possible, but no guarantee...
- \( w(\text{Rhea}, \text{Cronus})? \) → she has to be the wife of some grandfather
  → Cronus is a candidate; could there be another?
Answering queries using views

Given:
\( \mathbf{V} \): view from \( \sigma \) to \( \tau \)
\( E \): view instance over \( \tau \)
\( Q \): query over \( \sigma \)

- Certain answers:

\[
\text{cert}_{Q,\mathbf{V}}(E) = \bigcap_{D \mid \mathbf{V}(D) = E} Q(D)
\]
Answering queries using views

Given: \[\begin{align*}
V &: \text{view from } \sigma \text{ to } \tau \\
E &: \text{view instance over } \tau \\
Q &: \text{query over } \sigma 
\end{align*}\]

- Certain answers under the **exact view assumption**: 

\[
\text{cert}_{Q,V}^{\text{exact}}(E) = \bigcap_{D \mid V(D) = E} Q(D)
\]

- Certain answers under the **sound view assumption**: 

\[
\text{cert}_{Q,V}^{\text{sound}}(E) = \bigcap_{D \mid V(D) \supseteq E} Q(D)
\]
The problem(s) of computing certain answers

**Problem**: Certain answers

A view $V$ from $\sigma$ to $\tau$,

**Input**: A query $Q$ over $\sigma$,

A view instance $E$ and $\bar{u} \in E$

**Question**: $\bar{u} \in \text{cert}_{Q,V}(E)$?
The problem(s) of computing certain answers

<table>
<thead>
<tr>
<th><strong>Problem</strong></th>
<th>**Certain answers for ( \mathcal{L} ) and ( \mathcal{L'} )</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input</strong></td>
<td>An ( \mathcal{L} )-view ( \mathbf{V} ) from ( \sigma ) to ( \tau ), An ( \mathcal{L}' )-query ( Q ) over ( \sigma ), A view instance ( E ) and ( \bar{u} \in E )</td>
</tr>
<tr>
<td><strong>Question</strong></td>
<td>( \bar{u} \in \text{cert}_{Q,V}(E) )?</td>
</tr>
</tbody>
</table>


The problem(s) of computing certain answers

**Problem**: Certain answers for $\mathcal{L}$ and $\mathcal{L}'$

- An $\mathcal{L}$-view $V$ from $\sigma$ to $\tau$,

**Input**: An $\mathcal{L}'$-query $Q$ over $\sigma$,
- A view instance $E$ and $\bar{u} \in E$

**Question**: $\bar{u} \in \text{cert}^{\text{exact}}_{Q,V}(E)$? $\bar{u} \in \text{cert}^{\text{sound}}_{Q,V}(E)$?
The problem(s) of computing certain answers

- **Combined** complexity:
  
  **Problem**: Certain answers for $\mathcal{L}$ and $\mathcal{L}'$
  
  **Input**: An $\mathcal{L}$-view $V$ from $\sigma$ to $\tau$,
  
  An $\mathcal{L}'$-query $Q$ over $\sigma$,
  
  A view instance $E$ and $\bar{u} \in E$
  
  **Question**: $\bar{u} \in \text{cert}_{Q,V}^{\text{exact}}(E)$? $\bar{u} \in \text{cert}_{Q,V}^{\text{sound}}(E)$?

- **Data** complexity:
  
  Let $V$ be a fixed view from $\sigma$ to $\tau$ and $Q$ be a fixed query over $\sigma$:
  
  **Problem**: Certain answers $(Q, V)$
  
  **Input**: A view instance $E$ and $\bar{u} \in E$
  
  **Question**: $\bar{u} \in \text{cert}_{Q,V}^{\text{exact}}(E)$? $\bar{u} \in \text{cert}_{Q,V}^{\text{sound}}(E)$?
Example: how hard is computing certain answers?

$$\sigma = \{c, e, p\}$$

$$V = \left\{ \begin{array}{l}
Q_{\text{edge}}(x, y) = e(x, y) \\
Q_{\text{palette}}(x) = p(x) \\
Q_{\text{color}}(x) = \exists z \cdot p(z) \land c(x, z)
\end{array} \right\}$$
Example: how hard is computing certain answers?

\[ \sigma = \{ c, e, p \} \]

\[ \mathbf{V} = \begin{cases} 
Q_{\text{edge}}(x, y) = e(x, y) \\
Q_{\text{palette}}(x) = p(x) \\
Q_{\text{color}}(x) = \exists z \cdot p(z) \land c(x, z) 
\end{cases} \]

Can we answer \( Q_{\text{error}}() = \exists x, y, z \cdot c(x, z) \land c(y, z) \land e(x, y) \)?
Example: how hard is computing certain answers?

\[ \sigma = \{c, e, p\} \]

\[ V = \left\{ \begin{array}{l}
Q_{\text{edge}}(x, y) = e(x, y) \\
Q_{\text{palette}}(x) = p(x) \\
Q_{\text{color}}(x) = \exists z \cdot p(z) \land c(x, z)
\end{array} \right\} \]

Can we answer \( Q_{\text{error}}() = \exists x, y, z \cdot c(x, z) \land c(y, z) \land e(x, y) \)?

- **Exact view**: \( Q_{\text{error}} \) is certain iff the graph is **not** 3-colorable!
Example: how hard is computing certain answers?

\[ \sigma = \{ c, e, p \} \]

\[ \mathbf{V} = \left\{ \begin{array}{l}
Q_{\text{edge}}(x, y) = e(x, y) \\
Q_{\text{palette}}(x) = p(x) \\
Q_{\text{color}}(x) = \exists z \cdot p(z) \land c(x, z)
\end{array} \right\} \]

Can we answer \( Q_{\text{error}}() = \exists x, y, z \cdot c(x, z) \land c(y, z) \land e(x, y) \)?

- **Exact view**: \( Q_{\text{error}} \) is certain iff the graph is **not** 3-colorable!
- **Sound view**: we can always invent more colors!
Short answer: it’s hard, even for simple languages and in data complexity,
Short answer: it’s hard, even for simple languages and in data complexity, cert\textsuperscript{exact} is closely related to testing consistency.
Short answer: it’s hard, even for simple languages and in data complexity,
- $\text{cert}^{\text{exact}}$ is closely related to testing consistency.
  
  \textit{(it’s usually equivalent to testing inconsistency)}
How hard is computing certain answers?

Short answer: it’s hard, even for simple languages and in data complexity,

- $\text{cert}^{\text{exact}}$ is closely related to testing consistency.
  (it’s usually equivalent to testing inconsistency)

- $\text{cert}^{\text{sound}}$ is usually easier (but not strictly) than $\text{cert}^{\text{exact}}$
How hard is computing certain answers?

**Short answer:** it’s hard, even for simple languages and in data complexity,

- $\text{cert}^{\text{exact}}$ is closely related to testing consistency.
  
  *(it’s usually equivalent to testing inconsistency)*

- $\text{cert}^{\text{sound}}$ is usually easier *(but not strictly)* than $\text{cert}^{\text{exact}}$

Some results from [Abiteboul, Duschka’98]:

<table>
<thead>
<tr>
<th>view</th>
<th>query</th>
<th>CQ</th>
<th>$\text{CQ}\neq$</th>
<th>Datalog</th>
<th>FO</th>
</tr>
</thead>
<tbody>
<tr>
<td>CQ</td>
<td></td>
<td>$\text{PTIME/coNP}$</td>
<td>coNP</td>
<td>$\text{PTIME/coNP}$</td>
<td>Undec.</td>
</tr>
<tr>
<td>$\text{CQ}\neq$</td>
<td></td>
<td>$\text{PTIME/coNP}$</td>
<td>coNP</td>
<td>$\text{PTIME/coNP}$</td>
<td>Undec.</td>
</tr>
<tr>
<td>Datalog</td>
<td></td>
<td>$\text{coNP/Undec.}$</td>
<td>Undec.</td>
<td>Undec.</td>
<td>Undec.</td>
</tr>
</tbody>
</table>

Complexity of answering queries using sound or exact views.
Determinacy and rewriting
Example: determinacy and rewriting

\[ \sigma = \{ b, f, w \} \]

\[ \mathbf{V} = \left\{ \begin{array}{l}
Q_{s_1}(x) = \exists z \cdot x \xrightarrow{w} z \\
Q_{s_2}(x) = \exists z \cdot z \xrightarrow{w} x \\
Q_{gf}(x, y) = x \xrightarrow{f} z \xrightarrow{f} y
\end{array} \right\} \]

\[ \tau = \{ gf, s_1, s_2 \} \]
Example: determinacy and rewriting

\[ \sigma = \{ b, f, w \} \quad \text{and} \quad \tau = \{ gf, s1, s2 \} \]

\[ V = \left\{ \begin{array}{l}
Q_{s1}(x) = \exists z \cdot x \xrightarrow{w} z \\
Q_{s2}(x) = \exists z \cdot z \xrightarrow{w} x \\
Q_{gf}(x, y) = x \xrightarrow{f} z \xrightarrow{f} y
\end{array} \right\} \]

Can we answer \( Q(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \) based on the view instance?
Example: determinacy and rewriting

\[ \sigma = \{b, f, w\} \quad \mathbf{V} = \left\{ \begin{array}{l}
Q_{s1}(x) = \exists z \cdot x \xrightarrow{w} z \\
Q_{s2}(x) = \exists z \cdot z \xrightarrow{w} x \\
Q_{gf}(x, y) = x \xrightarrow{f} z \xrightarrow{f} y
\end{array} \right\} \quad \tau = \{gf, s1, s2\} \]

Can we answer \( Q(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \) based on the view instance?

Yes! Rhea is the grandmother of Athena. 

**And nothing else:** possible and certain answers coincide.
Example: determinacy and rewriting

\[ \sigma = \{ b, f, w \} \]
\[ \mathbf{V} = \left\{ \begin{array}{l}
Q_{s1}(x) = \exists z \cdot x \xrightarrow{w} z \\
Q_{s2}(x) = \exists z \cdot z \xrightarrow{w} x \\
Q_{gf}(x, y) = x \xrightarrow{f} z \xrightarrow{f} y
\end{array} \right\} \]
\[ \tau = \{ gf, s1, s2 \} \]

Can we answer \( Q(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \) based on the view instance?
Example: determinacy and rewriting

\[ \sigma = \{b, f, w\} \quad \mathbf{V} = \left\{ \begin{array}{l}
Q_{s1}(x) = \exists z \cdot x \xrightarrow{w} z \\
Q_{s2}(x) = \exists z \cdot z \xrightarrow{w} x \\
Q_{gf}(x, y) = x \xrightarrow{f} z \xrightarrow{f} y
\end{array} \right\} \quad \tau = \{gf, s1, s2\} \]

Can we answer \( Q(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \) based on the view instance?

There is no way to match husbands and wives... Nothing is certain and every match is possible.
Example: determinacy and rewriting (2)

\[
\sigma = \{ b, f, w \} \quad \mathbf{V} = \left\{ \begin{array}{l}
Q_w(x, y) = x \xrightarrow{w} y \\
Q_{gf}(x, y) = x \xrightarrow{f} z \xrightarrow{f} y
\end{array} \right\} 
\tau = \{ gf, w \}
\]
Example: determinacy and rewriting (2)

\( \sigma = \{b, f, w\} \)  \hspace{1cm} \mathbf{V} = \left\{ \begin{array}{l}
Q_w(x, y) = x \xrightarrow{w} y \\
Q_{gf}(x, y) = x \xrightarrow{f} z \xrightarrow{f} y
\end{array} \right\} \hspace{1cm} \tau = \{gf, w\}

Can we answer \( Q(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \) based on the view instance?
Example: determinacy and rewriting (2)

\[ \sigma = \{ b, f, w \} \quad \mathbf{V} = \left\{ \begin{align*}
Q_w(x, y) &= x \xrightarrow{w} y \\
Q_{gf}(x, y) &= x \xrightarrow{f} z \xrightarrow{f} y
\end{align*} \right\} \quad \tau = \{ gf, w \} \]

Can we answer \( Q(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \) based on the view instance?

- Yes! Rhea is grandmother to Ares and Athena, and Hera to Phobos.  
  And nothing else: possible and certain answers coincide.
Example: determinacy and rewriting (2)

\[ \sigma = \{ b, f, w \} \]
\[ \mathbf{V} = \left\{ \begin{array}{c}
Q_w(x, y) = x \xrightarrow{w} y \\
Q_{gf}(x, y) = x \xrightarrow{f} z \xrightarrow{f} y
\end{array} \right\} \]
\[ \tau = \{ gf, w \} \]

Can we answer \( Q(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \) based on the view instance?

- Yes! Rhea is grandmother to Ares and Athena, and Hera to Phobos. And nothing else: possible and certain answers coincide.
- Better yet: this is a static property of \( \mathbf{V} \) and \( Q \).
Example: determinacy and rewriting (2)

\[ \sigma = \{ b, f, w \} \]
\[ V = \left\{ \begin{array}{l}
Q_w(x, y) = x \xrightarrow{w} y \\
Q_{gf}(x, y) = x \xrightarrow{f} z \xrightarrow{f} y
\end{array} \right\} \]
\[ \tau = \{ gf, w \} \]

Can we answer \( Q(x, y) = x \xrightarrow{w} z \xrightarrow{f} z' \xrightarrow{f} y \) based on the view instance?

- Yes! Rhea is grandmother to Ares and Athena, and Hera to Phobos. And nothing else: possible and certain answers coincide.
- Better yet: this is a static property of \( V \) and \( Q \).
- \( Q \) can be rewritten as \( R(x, y) = x \xrightarrow{w} z \xrightarrow{gf} y \) over \( \tau \).
Determinacy and rewriting

Let \( V \) be a view from \( \sigma \) to \( \tau \) and \( Q \) be a query over \( \sigma \).

- **Determinacy**: we say that \( V \) determines \( Q \) iff for any database \( D \), the certain answers and the possible answers to \( Q \) on \( V(D) \) coincide. This is a **static analysis** property.
Let $V$ be a view from $\sigma$ to $\tau$ and $Q$ be a query over $\sigma$.

- **Determinacy**: we say that $V$ determines $Q$ iff for any database $D$, the certain answers and the possible answers to $Q$ on $V(D)$ coincide. This is a static analysis property.

- **Equivalent definition**: $V$ determines $Q$ (denoted $V \rightarrow Q$) iff:

\[
\forall D, D' \quad V(D) = V(D') \implies Q(D) = Q(D')
\]
Determinacy and rewriting

Let $V$ be a view from $\sigma$ to $\tau$ and $Q$ be a query over $\sigma$.

- **Determinacy**: we say that $V$ determines $Q$ iff for any database $D$, the certain answers and the possible answers to $Q$ on $V(D)$ coincide. This is a static analysis property.

- **Equivalent definition**: $V$ determines $Q$ (denoted $V \rightarrow Q$) iff:

  $\forall D, D' \quad V(D) = V(D') \implies Q(D) = Q(D')$

  $\implies \quad \bigcap_{D \geq D_0} Q(D_0) \subseteq Q(D) \subseteq \bigcup_{D_0} Q(D_0)$

  $V(D) = V(D_0) \quad \quad V(D) = V(D_0)$
Let $V$ be a view from $\sigma$ to $\tau$ and $Q$ be a query over $\sigma$.

- **Determinacy**: we say that $V$ determines $Q$ iff for any database $D$, the certain answers and the possible answers to $Q$ on $V(D)$ coincide. This is a static analysis property.

- **Equivalent definition**: $V$ determines $Q$ (denoted $V \hookrightarrow Q$) iff:

  \[
  \forall D, D' \quad V(D) = V(D') \implies Q(D) = Q(D')
  \]

  \[
  \Rightarrow \quad \bigcap_{D_0} Q(D_0) \subseteq Q(D) \subseteq \bigcup_{D_0} Q(D_0)
  \]

  \[
  V(D) = V(D_0) \quad V(D) = V(D_0)
  \]

  Assume possible and certain answers coincide, then:
Determinacy and rewriting

Let $V$ be a view from $\sigma$ to $\tau$ and $Q$ be a query over $\sigma$.

- **Determinacy**: we say that $V$ determines $Q$ iff for any database $D$, the certain answers and the possible answers to $Q$ on $V(D)$ coincide. This is a static analysis property.

- **Equivalent definition**: $V$ determines $Q$ (denoted $V \xrightarrow{} Q$) iff:

\[
\forall D, D' \quad V(D) = V(D') \implies Q(D) = Q(D')
\]

\[
\Rightarrow \\
\bigcap_{D_0} Q(D_0) \subseteq Q(D) \subseteq \bigcup_{D_0} Q(D_0)
\]

Assume possible and certain answers coincide, then:

\[
Q(D) = \text{cert}_{Q,V}(V(D))
\]
Let $V$ be a view from $\sigma$ to $\tau$ and $Q$ be a query over $\sigma$.

- **Determinacy:** we say that $V$ determines $Q$ iff for any database $D$, the certain answers and the possible answers to $Q$ on $V(D)$ coincide. This is a static analysis property.

- **Equivalent definition:** $V$ determines $Q$ (denoted $V \rightarrow Q$) iff:

$$\forall D, D': V(D) = V(D') \implies Q(D) = Q(D')$$

$$\Rightarrow \bigcap_{D_0} Q(D_0) \subseteq Q(D) \subseteq \bigcup_{D_0} Q(D_0)$$

$V(D) = V(D_0)$ $V(D) = V(D_0)$

Assume possible and certain answers coincide, then:

$$Q(D) = \text{cert}_{Q, V}(V(D)) = \text{cert}_{Q, V}(V(D'))$$
Determinacy and rewriting

Let $V$ be a view from $\sigma$ to $\tau$ and $Q$ be a query over $\sigma$.

- **Determinacy:** we say that $V$ determines $Q$ iff for any database $D$, the certain answers and the possible answers to $Q$ on $V(D)$ coincide. This is a static analysis property.

- **Equivalent definition:** $V$ determines $Q$ (denoted $V \dashrightarrow Q$) iff:

\[
\forall D, D' \quad V(D) = V(D') \implies Q(D) = Q(D')
\]

\[
\Rightarrow 
\bigcap_{D_0} Q(D_0) \subseteq Q(D) \subseteq \bigcup_{D_0} Q(D_0)
\]

$V(D) = V(D_0)$ $V(D) = V(D_0)$

Assume possible and certain answers coincide, then:

\[
Q(D) = \text{cert}_{Q,V}(V(D)) = \text{cert}_{Q,V}(V(D')) = Q(D')
\]
Let $V$ be a view from $\sigma$ to $\tau$ and $Q$ be a query over $\sigma$.

- **Determinacy**: we say that $V$ determines $Q$ iff for any database $D$, the certain answers and the possible answers to $Q$ on $V(D)$ coincide. This is a static analysis property.

- **Equivalent definition**: $V$ determines $Q$ (denoted $V \rightarrow Q$) iff:

\[
\forall D, D' \quad V(D) = V(D') \implies Q(D) = Q(D')
\]

Assume $V \rightarrow Q$. Let $D$ be any database, then:
Determinacy and rewriting

Let $V$ be a view from $\sigma$ to $\tau$ and $Q$ be a query over $\sigma$.

- **Determinacy**: we say that $V$ determines $Q$ iff for any database $D$, the certain answers and the possible answers to $Q$ on $V(D)$ coincide. This is a static analysis property.

- **Equivalent definition**: $V$ determines $Q$ (denoted $V \rightarrow Q$) iff:

  $$\forall D, D' \quad V(D) = V(D') \implies Q(D) = Q(D')$$

$\leftarrow$ Assume $V \rightarrow Q$. Let $D$ be any database, then:

$$\bigcap_{D_0} Q(D_0) = Q(D)$$

$$V(D) = V(D_0)$$
Determinacy and rewriting

Let \( V \) be a view from \( \sigma \) to \( \tau \) and \( Q \) be a query over \( \sigma \).

- **Determinacy**: we say that \( V \) determines \( Q \) iff for any database \( D \), the certain answers and the possible answers to \( Q \) on \( V(D) \) coincide. This is a static analysis property.

- **Equivalent definition**: \( V \) determines \( Q \) (denoted \( V \rightarrow Q \)) iff:

\[
\forall D, D' \quad V(D) = V(D') \implies Q(D) = Q(D')
\]

\[\begin{align*}
\Rightarrow & \quad \text{Assume } V \rightarrow Q. \text{ Let } D \text{ be any database, then:} \\
& \quad \bigcap_{D_0} Q(D_0) = Q(D) \quad \text{and} \quad \bigcup_{D_0} Q(D_0) = Q(D) \\
& \quad V(D) = V(D_0) \quad \text{and} \quad V(D) = V(D_0)
\end{align*}\]
Let $V$ be a view from $\sigma$ to $\tau$ and $Q$ be a query over $\sigma$.

- **Determinacy**: we say that $V$ determines $Q$ iff for any database $D$, the certain answers and the possible answers to $Q$ on $V(D)$ coincide. This is a static analysis property.

- **Equivalent definition**: $V$ determines $Q$ (denoted $V \twoheadrightarrow Q$) iff:

$$\forall D, D' \quad V(D) = V(D') \implies Q(D) = Q(D')$$

$\iff$ Assume $V \twoheadrightarrow Q$. Let $D$ be any database, then:

$$\bigcap_{D_0} Q(D_0) = Q(D) \quad \text{and} \quad \bigcup_{D_0} Q(D_0) = Q(D)$$

$V(D) = V(D_0)$ \quad $V(D) = V(D_0)$

Thus possible and certain answers coincide both with $Q(D)$. 


Determinacy and rewriting

Let \( V \) be a view from \( \sigma \) to \( \tau \) and \( Q \) be a query over \( \sigma \).

- **Determinacy**: we say that \( V \) determines \( Q \) iff for any database \( D \), the certain answers and the possible answers to \( Q \) on \( V(D) \) coincide. This is a static analysis property.

- **Equivalent definition**: \( V \) determines \( Q \) (denoted \( V \mathrel{\rightarrow} Q \)) iff:

\[
\forall D, D' \quad V(D) = V(D') \implies Q(D) = Q(D')
\]

- **Rewriting**: A rewriting of \( Q \) using \( V \) is a query \( R \) over \( \tau \) such that:

\[
\forall D, \quad Q(D) = R(V(D))
\]
Determinacy and rewriting

Let \( V \) be a view from \( \sigma \) to \( \tau \) and \( Q \) be a query over \( \sigma \).

- **Determinacy**: we say that \( V \) determines \( Q \) iff for any database \( D \), the certain answers and the possible answers to \( Q \) on \( V(D) \) coincide. This is a static analysis property.

- **Equivalent definition**: \( V \) determines \( Q \) (denoted \( V \hookrightarrow Q \)) iff:

  \[
  \forall D, D' \quad V(D) = V(D') \implies Q(D) = Q(D')
  \]

- **Rewriting**: A rewriting of \( Q \) using \( V \) is a query \( R \) over \( \tau \) such that:

  \[
  \forall D, \quad Q(D) = R(V(D))
  \]

**Remark**: Assume \( V(D) = V(D') \), then \( R(V(D)) = R(V(D')) \). Thus \( Q(D) = Q(D') \).
Let $V$ be a view from $\sigma$ to $\tau$ and $Q$ be a query over $\sigma$.

- **Determinacy**: we say that $V$ determines $Q$ iff for any database $D$, the certain answers and the possible answers to $Q$ on $V(D)$ coincide. This is a static analysis property.

- **Equivalent definition**: $V$ determines $Q$ (denoted $V \hookrightarrow Q$) iff:
  \[
  \forall D, D' \quad V(D) = V(D') \implies Q(D) = Q(D')
  \]

- **Rewriting**: A rewriting of $Q$ using $V$ is a query $R$ over $\tau$ such that:
  \[
  \forall D, \quad Q(D) = R(V(D))
  \]

**Remark**: Assume $V(D) = V(D')$, then $R(V(D)) = R(V(D'))$. Thus $Q(D) = Q(D')$.

The existence of a rewriting immediately implies that $V \hookrightarrow Q$. 
Example: proving non-determinacy

\[ \sigma = \{ a \} \quad \mathbf{V} = \left\{ Q_{a_3}(x, y) = x \xrightarrow{a} z \xrightarrow{a} z' \xrightarrow{a} y \right\} \quad \tau = \{ a_3 \} \]
Example: proving non-determinacy

$\sigma = \{a\}$ \hspace{1cm} $V = \left\{ Q_{a_3}(x, y) = x \xrightarrow{a_3} y \right\}$ \hspace{1cm} $\tau = \{a_3\}$
Example: proving non-determinacy

\[ \sigma = \{ a \} \quad V = \left\{ Q_{a_3}(x, y) = x \xrightarrow{a_3} y \right\} \quad \tau = \{ a_3 \} \]

How to prove that \( V \) does \textbf{not} determine \( Q(x, y) = x \xrightarrow{a_4} y \)?
Example: proving non-determinacy

\[ \sigma = \{ a \} \quad \mathcal{V} = \left\{ Q_{a_3}(x, y) = x \xrightarrow{a^3} y \right\} \quad \tau = \{ a_3 \} \]

How to prove that \( \mathcal{V} \) does not determine \( Q(x, y) = x \xrightarrow{a^4} y \)?
Example: proving non-determinacy

\[ \sigma = \{a\} \quad V = \left\{ Q_{a_3}(x, y) = x \xrightarrow{a_3} y \right\} \quad \tau = \{a_3\} \]

How to prove that \( V \) does not determine \( Q(x, y) = x \xrightarrow{a_4} y \)?
Example: proving non-determinacy

\[ \sigma = \{ a \} \quad V = \left\{ Q_{a_3}(x, y) = x \xrightarrow{a_3} y \right\} \quad \tau = \{ a_3 \} \]

How to prove that \( V \) does not determine \( Q(x, y) = x \xrightarrow{a_4} y \)?
Example: proving non-determinacy

\[ \sigma = \{a\} \quad \mathbf{V} = \left\{ Q_{a_3}(x, y) = x \xrightarrow{a_3} y \right\} \quad \tau = \{a_3\} \]

How to prove that \( \mathbf{V} \) does not determine \( Q(x, y) = x \xrightarrow{a_4} y \)?

\[
\begin{align*}
\mathbf{V}(D) = \mathbf{V}(D') \text{ but } Q(D) &= \{(\bullet, \bullet)\} \text{ and } Q(D') = \emptyset
\end{align*}
\]
Example: proving determinacy

\[ \sigma = \{ a \} \quad \mathbf{V} = \left\{ Q_{a^2}(x, y) = x \xrightarrow{a^2} y \right\} \quad \tau = \{ a_2 \} \]
Example: proving determinacy

\[ \sigma = \{ a \} \]

\[ \mathbf{V} = \left\{ Q_{a_2}(x, y) = x \xrightarrow{a^2} y \right\} \]

\[ \tau = \{ a_2 \} \]

How to prove that \( \mathbf{V} \) determines \( Q(x, y) = x \xrightarrow{a^4} y \)?
Example: proving determinacy

\[ \sigma = \{ a \} \quad \mathbf{V} = \left\{ Q_{a_2}(x, y) = x \xrightarrow{a^2} y \right\} \quad \tau = \{ a_2 \} \]

How to prove that \( \mathbf{V} \) determines \( Q(x, y) = x \xrightarrow{a^4} y \)?

\[ \xrightarrow{\text{By providing a rewriting of } Q \text{ using } \mathbf{V}} \]
Example: proving determinacy

\[ \sigma = \{ a \} \quad \mathbf{V} = \left\{ Q_{a^2}(x, y) = x \xrightarrow{a^2} y \right\} \quad \tau = \{ a_2 \} \]

How to prove that \( \mathbf{V} \) determines \( Q(x, y) = x \xrightarrow{a^4} y \)?

\[ \left\uparrow \right\] By providing a rewriting of \( Q \) using \( \mathbf{V} \)

\[ R(x, y) = \exists z \cdot a_2(x, z) \land a_2(z, y) \]
Example: proving determinacy

\[ \sigma = \{ a \} \quad \text{V} = \left\{ Q_{a^2}(x, y) = x \xrightarrow{a^2} y \right\} \quad \tau = \{ a_2 \} \]

How to prove that V determines \( Q(x, y) = x \xrightarrow{a^4} y \)?

\[ \downarrow \quad \text{By providing a rewriting of Q using V} \]

\[ R(x, y) = \exists z \cdot a_2(x, z) \land a_2(z, y) \]

\[ Q(D) \subseteq R(V(D)) \quad R(V(D)) \subseteq Q(D) \]
Example: proving determinacy

\[ \sigma = \{a\} \]
\[ V = \left\{ Q_{a^2}(x, y) = x \xrightarrow{a^2} y \right\} \]
\[ \tau = \{a^2\} \]

How to prove that \( V \) determines \( Q(x, y) = x \xrightarrow{a^4} y \)?

\[ \Rightarrow \text{By providing a rewriting of } Q \text{ using } V \]

\[ R(x, y) = \exists z \cdot a_2(x, z) \land a_2(z, y) \]

\[ Q(D) \subseteq R(V(D)) \]
\[ R(V(D)) \subseteq Q(D) \]
Example: proving determinacy

\[ \sigma = \{ a \} \]

\[ \mathbf{V} = \left\{ Q_{a_2}(x, y) = x \xrightarrow{a^2} y \right\} \]

\[ \tau = \{ a_2 \} \]

How to prove that \( \mathbf{V} \) determines \( Q(x, y) = x \xrightarrow{a^4} y \)?

\[ \xrightarrow{\rightarrow} \text{By providing a rewriting of } Q \text{ using } \mathbf{V} \]

\[ R(x, y) = \exists z \cdot a_2(x, z) \land a_2(z, y) \]

\[ Q(D) \subseteq R(\mathbf{V}(D)) \]

\[ R(\mathbf{V}(D)) \subseteq Q(D) \]

\[ x \xrightarrow{\rightarrow} y \]
Example: proving determinacy

\[\sigma = \{a\}\]

\[\mathbf{V} = \left\{ Q_{a^2}(x, y) = x \xrightarrow{a^2} y \right\}\]

\[\tau = \{a^2\}\]

How to prove that \(\mathbf{V}\) determines \(Q(x, y) = x \xrightarrow{a^4} y\)?

\[\iff\] By providing a rewriting of \(Q\) using \(\mathbf{V}\)

\[R(x, y) = \exists z \cdot a^2(x, z) \land a^2(z, y)\]

\[Q(D) \subseteq R(\mathbf{V}(D))\]

\[R(\mathbf{V}(D)) \subseteq Q(D)\]
Example: proving determinacy

\[
\sigma = \{ a \} \\
V = \left\{ Q_{a_2}(x, y) = x \xrightarrow{a^2} y \right\} \\
\tau = \{ a_2 \}
\]

How to prove that \( V \) determines \( Q(x, y) = x \xrightarrow{a^4} y \)?

\[ \downarrow \] By providing a rewriting of \( Q \) using \( V \)

\[ R(x, y) = \exists z \cdot a_2(x, z) \land a_2(z, y) \]

\[ Q(D) \subseteq R(V(D)) \checkmark \]

\[ R(V(D)) \subseteq Q(D) \]
Example: proving determinacy

\[ \sigma = \{ a \} \]
\[ \mathbf{V} = \left\{ Q_{a^2}(x, y) = x \xrightarrow{a^2} y \right\} \]
\[ \tau = \{ a_2 \} \]

How to prove that \( \mathbf{V} \) determines \( Q(x, y) = x \xrightarrow{a^4} y \)?

\[ \downarrow \text{By providing a rewriting of } Q \text{ using } \mathbf{V} \]

\[ R(x, y) = \exists z \cdot a_2(x, z) \land a_2(z, y) \]

\[ Q(D) \subseteq R(\mathbf{V}(D)) \quad \checkmark \]
\[ R(\mathbf{V}(D)) \subseteq Q(D) \]
Example: proving determinacy

\[ \sigma = \{ a \} \]
\[ \mathbf{V} = \{ Q_{a^2}(x, y) = x \xrightarrow{a^2} y \} \]
\[ \tau = \{ a^2 \} \]

How to prove that \( \mathbf{V} \) determines \( Q(x, y) = x \xrightarrow{a^4} y \)?

\[ \rightarrow \text{By providing a rewriting of} \ Q \text{using} \ \mathbf{V} \]

\[ R(x, y) = \exists z \cdot a_2(x, z) \land a_2(z, y) \]

\[ Q(D) \subseteq R(\mathbf{V}(D)) \checkmark \]
\[ R(\mathbf{V}(D)) \subseteq Q(D) \]
Example: proving determinacy

\[\sigma = \{a\} \quad \mathbf{V} = \left\{ Q_{a^2}(x, y) = x \xrightarrow{a^2} y \right\} \quad \tau = \{a^2\}\]

How to prove that \(\mathbf{V}\) determines \(Q(x, y) = x \xrightarrow{a^4} y\)?

\[\rightarrow \text{By providing a rewriting of } Q \text{ using } \mathbf{V}
\]

\[R(x, y) = \exists z \cdot a^2(x, z) \wedge a^2(z, y)\]

\[Q(D) \subseteq R(\mathbf{V}(D)) \checkmark \quad \mathbf{R}(\mathbf{V}(D)) \subseteq Q(D)\]
Example: proving determinacy

\[ \sigma = \{ a \} \quad \mathbf{V} = \left\{ Q_{a^2}(x, y) = x \xrightarrow{a^2} y \right\} \quad \tau = \{ a^2 \} \]

How to prove that \( \mathbf{V} \) determines \( Q(x, y) = x \xrightarrow{a^4} y \)?

\[ \rightarrow \text{By providing a rewriting of } Q \text{ using } \mathbf{V} \]

\[ R(x, y) = \exists z \cdot a_2(x, z) \land a_2(z, y) \]

\[ Q(D) \subseteq R(\mathbf{V}(D)) \checkmark \quad R(\mathbf{V}(D)) \subseteq Q(D) \checkmark \]
Example: it’s not always that simple… [Afrati’11]

\[ \sigma = \{ a \} \]

\[ \mathbf{V} = \left\{ \begin{array}{l}
Q_{a_3}(x, y) = x \xrightarrow{a^3} y \\
Q_{a_4}(x, y) = x \xrightarrow{a^4} y
\end{array} \right\} \]

\[ \tau = \{ a_3, a_4 \} \]
\[\sigma = \{a\}\]

\[\mathbf{V} = \left\{ \begin{array}{l}
Q_{a_3}(x, y) = x \xrightarrow{a^3} y \\
Q_{a_4}(x, y) = x \xrightarrow{a^4} y
\end{array} \right\} \]

\[\tau = \{a_3, a_4\}\]

Does \(\mathbf{V}\) determine \(Q(x, y) = x \xrightarrow{a^5} y\)?
Example: it’s not always that simple...

\[ \sigma = \{ a \} \]

\[ \mathbf{V} = \left\{ Q_{a_3}(x, y) = x \xrightarrow{a^3} y, Q_{a_4}(x, y) = x \xrightarrow{a^4} y \right\} \]

\[ \tau = \{ a_3, a_4 \} \]

Does \( \mathbf{V} \) determine \( Q(x, y) = x \xrightarrow{a^5} y \)?

\[ R(x, y) = \exists z \cdot a_4(x, z) \land (\forall z' \cdot a_3(z', z) \implies a_4(z', y)) \]
Example: it’s not always that simple…

\[ \sigma = \{a\} \quad \mathbf{V} = \left\{ \begin{array}{l} Q_{a_3}(x, y) = x \xrightarrow{a^3} y \\ Q_{a_4}(x, y) = x \xrightarrow{a^4} y \end{array} \right\} \quad \tau = \{a_3, a_4\} \]

Does \( \mathbf{V} \) determine \( Q(x, y) = x \xrightarrow{a^5} y \)?

\[ R(x, y) = \exists z \cdot a_4(x, z) \land (\forall z' \cdot a_3(z', z) \Rightarrow a_4(z', y)) \]

\[ Q(D) \subseteq R(\mathbf{V}(D)) \quad R(\mathbf{V}(D)) \subseteq Q(D) \]
Example: it’s not always that simple... [Afrati’11]

\[\sigma = \{a\}\]
\[\mathbf{V} = \left\{ \begin{array}{l}
Q_{a_3}(x, y) = x \xrightarrow{a^3} y \\
Q_{a_4}(x, y) = x \xrightarrow{a^4} y
\end{array} \right\}\]
\[\tau = \{a_3, a_4\}\]

Does \(\mathbf{V}\) determine \(Q(x, y) = x \xrightarrow{a^5} y\)?

\[R(x, y) = \exists z \cdot a_4(x, z) \land (\forall z' \cdot a_3(z', z) \implies a_4(z', y))\]

\(Q(D) \subseteq R(\mathbf{V}(D))\)
\(R(\mathbf{V}(D)) \subseteq Q(D)\)

\(\circ\)
Example: it’s not always that simple... [Afrati’11]

\[ \sigma = \{ a \} \]

\[ \mathbf{v} = \left\{ \begin{array}{l}
Q_{a_3}(x, y) = x \xrightarrow{a^3} y \\
Q_{a_4}(x, y) = x \xrightarrow{a^4} y
\end{array} \right\} \quad \tau = \{ a_3, a_4 \} \]

Does \( \mathbf{v} \) determine \( Q(x, y) = x \xrightarrow{a^5} y \)?

\[ R(x, y) = \exists z \cdot a_4(x, z) \land (\forall z' \cdot a_3(z', z) \Rightarrow a_4(z', y)) \]

\[ Q(D) \subseteq R(\mathbf{v}(D)) \quad R(\mathbf{v}(D)) \subseteq Q(D) \]
Example: it’s not always that simple...

\[\sigma = \{a\}\]

\[\mathbf{V} = \begin{cases} 
Q_{a_3}(x, y) = x \xrightarrow{a^3} y \\
Q_{a_4}(x, y) = x \xrightarrow{a^4} y 
\end{cases}\]

\[\tau = \{a_3, a_4\}\]

Does \(\mathbf{V}\) determine \(Q(x, y) = x \xrightarrow{a^5} y\)?

\[R(x, y) = \exists z \cdot a_4(x, z) \land (\forall z' \cdot a_3(z', z) \Rightarrow a_4(z', y))\]

\[Q(D) \subseteq R(\mathbf{V}(D)) \quad R(\mathbf{V}(D)) \subseteq Q(D)\]
Example: it’s not always that simple...

\[ \sigma = \{ a \} \quad \mathbf{V} = \left\{ \begin{array}{l}
Q_{a_3}(x, y) = x \xrightarrow{a^3} y \\
Q_{a_4}(x, y) = x \xrightarrow{a^4} y
\end{array} \right\} \quad \tau = \{ a_3, a_4 \} \]

Does \( \mathbf{V} \) determine \( Q(x, y) = x \xrightarrow{a^5} y \)?

\[ R(x, y) = \exists z \cdot a_4(x, z) \land (\forall z' \cdot a_3(z', z) \Rightarrow a_4(z', y)) \]

\[ Q(D) \subseteq R(\mathbf{V}(D)) \quad R(\mathbf{V}(D)) \subseteq Q(D) \]
Example: it’s not always that simple... [Afrati’11]

\[ \sigma = \{ a \} \]

\[ \mathbf{V} = \left\{ \begin{array}{l}
Q_{a_3}(x, y) = x \xrightarrow{a^3} y \\
Q_{a_4}(x, y) = x \xrightarrow{a^4} y
\end{array} \right\} \]

\[ \tau = \{ a_3, a_4 \} \]

Does \( \mathbf{V} \) determine \( Q(x, y) = x \xrightarrow{a^5} y \)?

\[ R(x, y) = \exists z \cdot a_4(x, z) \land (\forall z' \cdot a_3(z', z) \Rightarrow a_4(z', y)) \]

\[ Q(D) \subseteq R(\mathbf{V}(D)) \]

\[ R(\mathbf{V}(D)) \subseteq Q(D) \]
Example: it’s not always that simple…

\[ \sigma = \{ a \} \]

\[ V = \begin{cases} Q_{a_3}(x, y) = x \xrightarrow{a_3} y \\ Q_{a_4}(x, y) = x \xrightarrow{a_4} y \end{cases} \]

\[ \tau = \{ a_3, a_4 \} \]

Does \( V \) determine \( Q(x, y) = x \xrightarrow{a_5} y \)?

\[ R(x, y) = \exists z \cdot a_4(x, z) \land (\forall z' \cdot a_3(z', z) \Rightarrow a_4(z', y)) \]

\[ Q(D) \subseteq R(V(D)) \, \checkmark \quad R(V(D)) \subseteq Q(D) \]
Example: it’s not always that simple…

\[
\sigma = \{a\} \quad \text{V} = \left\{ \begin{array}{l}
Q_{a_3}(x, y) = x \xrightarrow{a^3} y \\
Q_{a_4}(x, y) = x \xrightarrow{a^4} y
\end{array} \right. \quad \tau = \{a_3, a_4\}
\]

Does \(\text{V}\) determine \(Q(x, y) = x \xrightarrow{a^5} y\)?

\[
R(x, y) = \exists z \cdot a_4(x, z) \land (\forall z' \cdot a_3(z', z) \Rightarrow a_4(z', y))
\]

\[Q(D) \subseteq R(\text{V}(D)) \checkmark \quad R(\text{V}(D)) \subseteq Q(D)\]
Example: it's not always that simple...

\[
\sigma = \{ a \} \\
\mathbf{V} = \left\{ \begin{array}{l} Q_{a_3}(x, y) = x \xrightarrow{a^3} y \\
Q_{a_4}(x, y) = x \xrightarrow{a^4} y \end{array} \right\} \\
\tau = \{ a_3, a_4 \}
\]

Does \( \mathbf{V} \) determine \( Q(x, y) = x \xrightarrow{a^5} y \)?

\[
R(x, y) = \exists z \cdot a_4(x, z) \land (\forall z' \cdot a_3(z', z) \Rightarrow a_4(z', y))
\]

\( Q(D) \subseteq R(\mathbf{V}(D)) \) ✓

\( R(\mathbf{V}(D)) \subseteq Q(D) \)
Example: it’s not always that simple…

\[ \sigma = \{ a \} \quad \mathbf{V} = \left\{ \begin{array}{ll} Q_{a3}(x, y) = x \xrightarrow{a^3} y \\ Q_{a4}(x, y) = x \xrightarrow{a^4} y \end{array} \right\} \quad \tau = \{ a_3, a_4 \} \]

Does \( \mathbf{V} \) determine \( Q(x, y) = x \xrightarrow{a^5} y \)?

\[ R(x, y) = \exists z \cdot a_4(x, z) \land (\forall z' \cdot a_3(z', z) \Rightarrow a_4(z', y)) \]

\[ Q(D) \subseteq R(\mathbf{V}(D)) \checkmark \quad R(\mathbf{V}(D)) \subseteq Q(D) \]
Example: it's not always that simple... 

\[ \sigma = \{ a \} \]

\[ \mathbf{V} = \left\{ \begin{array}{l}
Q_{a_3}(x, y) = x \xrightarrow{a^3} y \\
Q_{a_4}(x, y) = x \xrightarrow{a^4} y
\end{array} \right\} \]

\[ \tau = \{ a_3, a_4 \} \]

Does \( \mathbf{V} \) determine \( Q(x, y) = x \xrightarrow{a^5} y \)?

\[ R(x, y) = \exists z \cdot a_4(x, z) \land (\forall z' \cdot a_3(z', z) \Rightarrow a_4(z', y)) \]

\[ Q(D) \subseteq R(\mathbf{V}(D)) \checkmark \]

\[ R(\mathbf{V}(D)) \subseteq Q(D) \]
Example: it’s not always that simple…

\[ \sigma = \{ a \} \]
\[ \mathbf{V} = \left\{ \begin{array}{ll} Q_{a_3}(x, y) &= x \xrightarrow{a^3} y \\ Q_{a_4}(x, y) &= x \xrightarrow{a^4} y \end{array} \right\} \]
\[ \tau = \{ a_3, a_4 \} \]

Does \( \mathbf{V} \) determine \( Q(x, y) = x \xrightarrow{a^5} y \)?

\[ R(x, y) = \exists z \cdot a_4(x, z) \land (\forall z' \cdot a_3(z', z) \Rightarrow a_4(z', y)) \]

\[ Q(D) \subseteq R(\mathbf{V}(D)) \checkmark \]

\[ R(\mathbf{V}(D)) \subseteq Q(D) \]
Example: it’s not always that simple...

\[ \sigma = \{a\} \quad \mathbf{V} = \left\{ \begin{array}{ll}
Q_{a_3}(x, y) = x \xrightarrow{a_3} y \\
Q_{a_4}(x, y) = x \xrightarrow{a_4} y
\end{array} \right\} \quad \tau = \{a_3, a_4\} \]

Does \( \mathbf{V} \) determine \( Q(x, y) = x \xrightarrow{a_5} y \)?

\[ R(x, y) = \exists z \cdot a_4(x, z) \land (\forall z' \cdot a_3(z', z) \Rightarrow a_4(z', y)) \]

\[ Q(D) \subseteq R(\mathbf{V}(D)) \checkmark \quad R(\mathbf{V}(D)) \subseteq Q(D) \checkmark \]
Let $V$ be a view from $\sigma$ to $\tau$ and $Q$ be a query over $\sigma$.

**Reminder**: if there exists $R$ such that $Q(D) = R(V(D))$, then $V \rightarrow Q$.

What about the converse?
Let \( V \) be a view from \( \sigma \) to \( \tau \) and \( Q \) be a query over \( \sigma \).

**Reminder**: if there exists \( R \) such that \( Q(D) = R(V(D)) \), then \( V \rightarrow Q \).

What about the converse?

Assume \( \forall D, D' \cdot V(D) = V(D') \implies Q(D) = Q(D') \)
Let $V$ be a view from $\sigma$ to $\tau$ and $Q$ be a query over $\sigma$.

**Reminder**: if there exists $R$ such that $Q(D) = R(V(D))$, then $V \rightarrow Q$.

What about the converse?

Assume $\forall D, D' \cdot V(D) = V(D') \implies Q(D) = Q(D')$

$\rightarrow$ **functional dependency** between $V(D)$ and $Q(D)$

$Q, V \leadsto f$: function induced by $Q$ using $V$
Let $V$ be a view from $\sigma$ to $\tau$ and $Q$ be a query over $\sigma$.

**Reminder**: if there exists $R$ such that $Q(D) = R(V(D))$, then $V \rightarrow Q$.

What about the converse?

Assume $\forall D, D' \cdot V(D) = V(D') \implies Q(D) = Q(D')$

$\rightarrow$ **functional dependency** between $V(D)$ and $Q(D)$

$Q, V \rightsquigarrow f$: function induced by $Q$ using $V$

$f$ is only defined over **view images**.

A **rewriting** is any query over $\tau$ that coincides with $f$ on **view images**.
Let \( V \) be a view from \( \sigma \) to \( \tau \) and \( Q \) be a query over \( \sigma \).

**Reminder:** if there exists \( R \) such that \( Q(D) = R(V(D)) \), then \( V \rightarrow Q \).

What about the converse?

Assume \( \forall D, D' \cdot V(D) = V(D') \implies Q(D) = Q(D') \)

\( \rightarrow \) functional dependency between \( V(D) \) and \( Q(D) \)

\( Q, V \leadsto f \): function induced by \( Q \) using \( V \)

\( f \) is only defined over view images.

A rewriting is any query over \( \tau \) that coincides with \( f \) on view images.

Does there always exist one?
Let $V$ be a view from $\sigma$ to $\tau$ and $Q$ be a query over $\sigma$.

**Reminder:** if there exists $R$ such that $Q(D) = R(V(D))$, then $V \rightarrow Q$. What about the converse?

Assume $\forall D, D' \cdot V(D) = V(D') \implies Q(D) = Q(D')$  
$\rightarrow$ **functional dependency** between $V(D)$ and $Q(D)$

$Q, V \rightsquigarrow f$: function induced by $Q$ using $V$

$f$ is only defined over **view images**.

A **rewriting** is any query over $\tau$ that coincides with $f$ on view images.

Does there always exist one?  
Yes! $\text{cert}_{Q,V}$ is always a rewriting.
Let $V$ be a view from $\sigma$ to $\tau$ and $Q$ be a query over $\sigma$.

**Reminder:** if there exists $R$ such that $Q(D) = R(V(D))$, then $V \rightarrow Q$. What about the converse?

Assume $\forall D, D' \cdot V(D) = V(D') \implies Q(D) = Q(D')$

$\rightarrow$ functional dependency between $V(D)$ and $Q(D)$

$Q, V \leadsto f$: function induced by $Q$ using $V$

$f$ is only defined over view images.

A rewriting is any query over $\tau$ that coincides with $f$ on view images.

Does there always exist one?

Yes! $\text{cert}_{Q,V}$ is always a rewriting.

But there can be more, as we have seen in previous examples.
Let $V$ be a view from $\sigma$ to $\tau$ and $Q$ be a query over $\sigma$.

**Reminder:** if there exists $R$ such that $Q(D) = R(V(D))$, then $V \rightarrow Q$.

What about the converse?

Assume $\forall D, D' \cdot V(D) = V(D') \implies Q(D) = Q(D')$

\[ \rightarrow \text{functional dependency between } V(D) \text{ and } Q(D) \]

$Q, V \rightsquigarrow f$: function induced by $Q$ using $V$

$f$ is only defined over view images.

A rewriting is any query over $\tau$ that coincides with $f$ on view images.

Does there always exist one?

Yes! $\text{cert}_{Q,V}$ is always a rewriting.

But there can be more, as we have seen in previous examples.

**Rewritings can differ in behavior and complexity outside of view images.**
Example: different rewritings of varying complexity

\[ \sigma = \{ c, e, p \} \]

\[ \mathbf{V} = \left\{ \begin{array}{l}
Q_{\text{edge}}(x, y) = e(x, y) \\
Q_{\text{palette}}(x) = p(x) \\
Q_{\text{color}}(x) = \exists z \cdot p(z) \land c(x, z) \\
Q_{\text{error}}(x, y) = \exists z \cdot c(x, z) \land c(y, z) \land e(x, y) 
\end{array} \right\} \]
Example: different rewritings of varying complexity

\[ \sigma = \{c, e, p\} \]

\[ \mathbf{V} = \left\{ \begin{array}{l}
Q_{\text{edge}}(x, y) = e(x, y) \\
Q_{\text{palette}}(x) = p(x) \\
Q_{\text{color}}(x) = \exists z \cdot p(z) \land c(x, z) \\
Q_{\text{error}}(x, y) = \exists z \cdot c(x, z) \land c(y, z) \land e(x, y)
\end{array} \right\} \]

\[ Q(x) = p(x) \land Q_{\text{error}} \]
Example: different rewritings of varying complexity

\( \sigma = \{ c, e, p \} \)

\[ V = \begin{cases} 
Q_{\text{edge}}(x, y) = e(x, y) \\
Q_{\text{palette}}(x) = p(x) \\
Q_{\text{color}}(x) = \exists z \cdot p(z) \land c(x, z) \\
Q_{\text{error}}(x, y) = \exists z \cdot c(x, z) \land c(y, z) \land e(x, y) 
\end{cases} \]

\[ Q(x) = p(x) \land Q_{\text{error}} \]

\[ R_1(x) = x \in \text{cert}_{Q, V}(E) \]

\[ R_2(x) = \text{palette}(x) \land \text{error}() \]
Example: different rewritings of varying complexity

\[ \sigma = \{c, e, p\} \]

\[ V = \begin{cases} 
Q_{\text{edge}}(x, y) = e(x, y) \\
Q_{\text{palette}}(x) = p(x) \\
Q_{\text{color}}(x) = \exists z \cdot p(z) \land c(x, z) \\
Q_{\text{error}}(x, y) = \exists z \cdot c(x, z) \land c(y, z) \land e(x, y) 
\end{cases} \]

\[ Q(x) = p(x) \land Q_{\text{error}} \]

\[ R_1(x) = x \in \text{cert}_{Q, V}(E) \]

(check if the graph is 3-colorable)

\[ R_2(x) = \text{palette}(x) \land \text{error}() \]

(trust the view instance)
### Problem

**Determinacy for languages $\mathcal{L}$ and $\mathcal{L}'$**

**Input:** An $\mathcal{L}$-view $V$ and an $\mathcal{L}'$-query $Q$

**Question:** Does $V \rightarrow Q$?

### Example

1. Is there a rewriting that can be expressed in first-order logic?
2. Is there a rewriting with $\mathcal{PTime}$ evaluation complexity?
3. Is there a rewriting that is monotone?
**Some problems around determinacy and rewritings**

<table>
<thead>
<tr>
<th>Problem</th>
<th>Determinacy for languages $\mathcal{L}$ and $\mathcal{L}'$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>An $\mathcal{L}$-view $\mathbf{V}$ and an $\mathcal{L}'$-query $Q$</td>
</tr>
<tr>
<td>Question</td>
<td>Does $\mathbf{V} \rightarrow Q$?</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Problem</th>
<th>$\mathcal{P}$-Rewriting for languages $\mathcal{L}$ and $\mathcal{L}'$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>An $\mathcal{L}$-view $\mathbf{V}$ and an $\mathcal{L}'$-query $Q$ st $\mathbf{V} \rightarrow Q$</td>
</tr>
<tr>
<td>Question</td>
<td>Is there a rewriting of $Q$ using $\mathbf{V}$ satisfying $\mathcal{P}$?</td>
</tr>
</tbody>
</table>
Some problems around determinacy and rewritings

**Problem:** Determinacy for languages \( \mathcal{L} \) and \( \mathcal{L}' \)

**Input:** An \( \mathcal{L} \)-view \( V \) and an \( \mathcal{L}' \)-query \( Q \)

**Question:** Does \( V \rightarrow Q \)?

---

**Problem:** \( \mathcal{P} \)-Rewriting for languages \( \mathcal{L} \) and \( \mathcal{L}' \)

**Input:** An \( \mathcal{L} \)-view \( V \) and an \( \mathcal{L}' \)-query \( Q \) st \( V \rightarrow Q \)

**Question:** Is there a rewriting of \( Q \) using \( V \) satisfying \( \mathcal{P} \)?

---

**Example:**

- Is there a rewriting that can be expressed in first-order logic?
- Is there a rewriting with \( \text{PTime} \) evaluation complexity?
- Is there a rewriting that is monotone?
How hard is the determinacy problem?

**Short answer:** it’s **hard**, even for simple query and view languages.
Short answer: it’s hard, even for simple query and view languages.

- [Gogacz, Marcinkowski'16]: Determinacy is undecidable for conjunctive queries and views.
How hard is the determinacy problem?

Short answer: it’s hard, even for simple query and view languages.

- [Gogacz, Marcinkowski’16]: Determinacy is **undecidable** for conjunctive queries and views.

- [Nash, Segoufin, Vianu’07]: Monotone rewritings of conjunctive queries using conjunctive views can be expressed in CQ.  
  \[(Existence \text{ is } \text{NP-complete}, \text{ using } [Levy \text{ et al’95})]\]
How hard is the determinacy problem?

**Short answer:** It’s hard, even for simple query and view languages.

- [Gogacz, Marcinkowski’16]: Determinacy is *undecidable* for conjunctive queries and views.

- [Nash, Segoufin, Vianu’07]: Monotone rewritings of conjunctive queries using conjunctive views can be expressed in CQ.
  
  (*Existence is NP-complete, using [Levy et al’95]*)

- [Afrati’11]: Determinacy is decidable for chain queries and chain views. Rewritings can be expressed in FO.
How hard is the determinacy problem?

Short answer: it’s hard, even for simple query and view languages.

- [Gogacz, Marcinkowski’16]: Determinacy is undecidable for conjunctive queries and views.

- [Nash, Segoufin, Vianu’07]: Monotone rewritings of conjunctive queries using conjunctive views can be expressed in CQ.
  
  \( \text{Existence is } \text{NP-complete, using [Levy et al’95]} \) 

- [Afrati’11]: Determinacy is decidable for chain queries and chain views. Rewritings can be expressed in FO.

Some results over graphs:

[Gluch et al’19]: Determinacy is undecidable for finite RPQs.

[F., Segoufin, Sirangelo’15]: Monotone rewritings of RPQ queries using RPQ views can be expressed in Datalog.

\( \text{Existence is } \text{ExpSpace-complete, using [Calvanese et al’02]} \)
How hard is the determinacy problem?

**Short answer**: it’s hard, even for simple query and view languages.

- [Gogacz, Marcinkowski’16]: Determinacy is **undecidable** for conjunctive queries and views.

- [Nash, Segoufin, Vianu’07]: Monotone rewritings of conjunctive queries using conjunctive views can be expressed in CQ.
  
  (*Existence is NP-complete, using [Levy et al’95]*)

- [Afrati’11]: Determinacy is decidable for chain queries and chain views. Rewritings can be expressed in FO.

**Some results over graphs:**

- [Gluch et al’19]: Determinacy is undecidable for finite RPQs.
How hard is the determinacy problem?

Short answer: it’s hard, even for simple query and view languages.

- [Gogacz, Marcinkowski’16]: Determinacy is undecidable for conjunctive queries and views.

- [Nash, Segoufin, Vianu’07]: Monotone rewritings of conjunctive queries using conjunctive views can be expressed in CQ.
  
  *(Existence is NP-complete, using [Levy et al’95])*

- [Afrati’11]: Determinacy is decidable for chain queries and chain views. Rewritings can be expressed in FO.

Some results over graphs:

- [Gluch et al’19]: Determinacy is undecidable for finite RPQs.

- [F., Segoufin, Sirangelo’15]: Monotone rewritings of RPQ queries using RPQ views can be expressed in Datalog.
  
  *(Existence is ExpSpace-complete, using [Calvanese et al’02])*
Some open problems

**Question 1**
In which language can we rewrite CQ queries using CQ views?
Some open problems

**Question 1**
In which language can we rewrite CQ queries using CQ views?

**Question 2**
In which language can we rewrite RPQ queries using RPQ views?
Some open problems

**Question 1**
In which language can we rewrite CQ queries using CQ views?

**Question 2**
In which language can we rewrite RPQ queries using RPQ views?

**Question 3**
Is determinacy decidable for chain queries and disjunctive chain views?
One last example
Example: disjunctive chain queries – the Chase

\[ \begin{align*}
\sigma &= \{ a \} \\
\tau &= \{ (2), (1, 2) \}
\end{align*} \]

\[ V = \begin{cases} 
Q_2(x, y) = x \xrightarrow{a^2} y \\
Q_{1,2}(x, y) = (x \xrightarrow{a} y) \lor (x \xrightarrow{a^2} y)
\end{cases} \]
Example: disjunctive chain queries – the Chase

\[\sigma = \{a\}\]
\[\tau = \{(2), (1, 2)\}\]

\[V = \begin{cases} 
Q_2(x, y) = x \xrightarrow{a^2} y \\
Q_{1,2}(x, y) = (x \xrightarrow{a} y) \lor (x \xrightarrow{a^2} y) 
\end{cases}\]

Does \(V\) determine \(Q(x, y) = x \xrightarrow{a} y\)?
Example: disjunctive chain queries – the Chase

\[ \sigma = \{ a \} \]
\[ \tau = \{(2), (1, 2)\} \]

\[ \mathbf{V} = \left\{ \begin{array}{l} Q_2(x, y) = x \xrightarrow{a^2} y \\ Q_{1,2}(x, y) = (x \xrightarrow{a} y) \lor (x \xrightarrow{a^2} y) \end{array} \right\} \]

Does \( \mathbf{V} \) determine \( Q(x, y) = x \xrightarrow{a} y \)?
Example: disjunctive chain queries – the Chase

\[ \sigma = \{a\} \]
\[ \tau = \{(2), (1, 2)\} \]

\[ \mathbf{V} = \left\{ \begin{array}{l}
Q_2(x, y) = x \xrightarrow{a^2} y \\
Q_{1,2}(x, y) = (x \xrightarrow{a} y) \vee (x \xrightarrow{a^2} y)
\end{array} \right\} \]

Does \( \mathbf{V} \) determine \( Q(x, y) = x \xrightarrow{a} y \)?
Example: disjunctive chain queries – the Chase

\[ \sigma = \{ a \} \]
\[ \tau = \{(2), (1, 2)\} \]

\[ \mathbf{V} = \left\{ \begin{array}{l}
Q_2(x, y) = x \xrightarrow{a^2} y \\
Q_{1,2}(x, y) = (x \xrightarrow{a} y) \lor (x \xrightarrow{a^2} y)
\end{array} \right\} \]

Does \( \mathbf{V} \) determine \( Q(x, y) = x \xrightarrow{a} y \)?
Example: disjunctive chain queries – the Chase

\[ \sigma = \{ a \} \]
\[ \tau = \{(2), (1, 2)\} \]

\[ V = \ \begin{cases} 
Q_2(x, y) = x \xrightarrow{a^2} y \\
Q_{1,2}(x, y) = (x \xrightarrow{a} y) \lor (x \xrightarrow{a^2} y) 
\end{cases} \]

Does \( V \) determine \( Q(x, y) = x \xrightarrow{a} y \)?
Example: disjunctive chain queries – the Chase

\[ \sigma = \{ a \} \]
\[ \tau = \{ (2), (1, 2) \} \]

\[ \mathbf{V} = \left\{ \begin{array}{l}
Q_2(x, y) = x \xrightarrow{a^2} y \\
Q_{1,2}(x, y) = (x \xrightarrow{a} y) \lor (x \xrightarrow{a^2} y)
\end{array} \right\} \]

Does \( \mathbf{V} \) determine \( Q(x, y) = x \xrightarrow{a} y \)?
Example: disjunctive chain queries – a rewriting

\[\sigma = \{a\}\]
\[\tau = \{a_2, a_{1,2}, a_{2,3}\}\]

\[V = \begin{cases} 
Q_2(x, y) &= x \xrightarrow{a^2} y \\
Q_{1,2}(x, y) &= (x \xrightarrow{a} y) \lor (x \xrightarrow{a^2} y) \\
Q_{2,3}(x, y) &= (x \xrightarrow{a^2} y) \lor (x \xrightarrow{a^3} y) 
\end{cases} \]
Example: disjunctive chain queries – a rewriting

\[ \sigma = \{ a \} \]
\[ \tau = \{ a_2, a_{1,2}, a_{2,3} \} \]
\[ V = \begin{cases} 
Q_2(x, y) = x \xrightarrow{a^2} y \\
Q_{1,2}(x, y) = (x \xrightarrow{a} y) \lor (x \xrightarrow{a^2} y) \\
Q_{2,3}(x, y) = (x \xrightarrow{a^2} y) \lor (x \xrightarrow{a^3} y) 
\end{cases} \]

Does \( V \) determine \( Q(x, y) = x \xrightarrow{a^5} y \)?
Example: disjunctive chain queries – a rewriting

\[ \sigma = \{ a \} \]
\[ \tau = \{ a_2, a_{1,2}, a_{2,3} \} \]

\[ V = \left\{ \begin{array}{l}
Q_2(x, y) = x \overset{a^2}{\rightarrow} y \\
Q_{1,2}(x, y) = (x \overset{a}{\rightarrow} y) \lor (x \overset{a^2}{\rightarrow} y) \\
Q_{2,3}(x, y) = (x \overset{a^2}{\rightarrow} y) \lor (x \overset{a^3}{\rightarrow} y)
\end{array} \right\} \]

Does \( V \) determine \( Q(x, y) = x \overset{a^5}{\rightarrow} y \)?

\[ R(x_0, x_5) = \exists x_2, x_3 \cdot a_2(x_0, x_2) \land a_2(x_3, x_5) \]
\[ \cdot a_{2,3}(x_2, x_5) \]
\[ \cdot \forall z \cdot a_{1,2}(z, x_2) \Rightarrow (a_2(z, x_2) \lor a_2(z, x_3)) \]
Example: disjunctive chain queries – a rewriting

\[ \sigma = \{ a \} \]
\[ \tau = \{ a_2, a_{1,2}, a_{2,3} \} \]

\[ V = \left\{ \begin{array}{l}
Q_2(x, y) = x \xrightarrow{a^2} y \\
Q_{1,2}(x, y) = (x \xrightarrow{a} y) \lor (x \xrightarrow{a^2} y) \\
Q_{2,3}(x, y) = (x \xrightarrow{a^2} y) \lor (x \xrightarrow{a^3} y)
\end{array} \right. \]

Does \( V \) determine \( Q(x, y) = x \xrightarrow{a^5} y \)?

\[ R(x_0, x_5) = \exists x_2, x_3 \cdot a_2(x_0, x_2) \land a_2(x_3, x_5) \]

\[ \cdot a_{2,3}(x_2, x_5) \]

\[ \cdot \forall z \cdot a_{1,2}(z, x_2) \Rightarrow (a_2(z, x_2) \lor a_2(z, x_3)) \]
Example: disjunctive chain queries – a rewriting

\[ \sigma = \{a\} \]
\[ \tau = \{a_2, a_{1,2}, a_{2,3}\} \]

\[ \mathbf{V} = \left\{ \begin{array}{l}
Q_2(x, y) = x \xrightarrow{a^2} y \\
Q_{1,2}(x, y) = (x \xrightarrow{a} y) \lor (x \xrightarrow{a^2} y) \\
Q_{2,3}(x, y) = (x \xrightarrow{a^2} y) \lor (x \xrightarrow{a^3} y) \\
\end{array} \right\} \]

Does \( \mathbf{V} \) determine \( Q(x, y) = x \xrightarrow{a^5} y \)?

\[ R(x_0, x_5) = \exists x_2, x_3 \cdot a_2(x_0, x_2) \land a_2(x_3, x_5) \]
\[ \cdot a_{2,3}(x_2, x_5) \]
\[ \cdot \forall z \cdot a_{1,2}(z, x_2) \Rightarrow (a_2(z, x_2) \lor a_2(z, x_3)) \]
Example: disjunctive chain queries – a rewriting

\[
\begin{align*}
\sigma &= \{a\} \\
\tau &= \{a_2, a_{1,2}, a_{2,3}\}
\end{align*}
\]

\[
V = \begin{cases}
Q_2(x, y) = x \xrightarrow{a^2} y \\
Q_{1,2}(x, y) = (x \xrightarrow{a} y) \lor (x \xrightarrow{a^2} y) \\
Q_{2,3}(x, y) = (x \xrightarrow{a^2} y) \lor (x \xrightarrow{a^3} y)
\end{cases}
\]

Does \( V \) determine \( Q(x, y) = x \xrightarrow{a^5} y \)?

\[
R(x_0, x_5) = \exists x_2, x_3 \cdot a_2(x_0, x_2) \land a_2(x_3, x_5) \\
\land a_{2,3}(x_2, x_5) \\
\land \forall z \cdot a_{1,2}(z, x_2) \Rightarrow (a_2(z, x_2) \lor a_2(z, x_3))
\]
Example: disjunctive chain queries – a rewriting

\[ \sigma = \{ a \} \]
\[ \tau = \{ a_2, a_{1,2}, a_{2,3} \} \]

\[ \mathbf{V} = \begin{cases} 
Q_2(x, y) = x \xrightarrow{a^2} y \\
Q_{1,2}(x, y) = (x \xrightarrow{a} y) \lor (x \xrightarrow{a^2} y) \\
Q_{2,3}(x, y) = (x \xrightarrow{a^2} y) \lor (x \xrightarrow{a^3} y) 
\end{cases} \]

Does \( \mathbf{V} \) determine \( Q(x, y) = x \xrightarrow{a^5} y \)?

\[ R(x_0, x_5) = \exists x_2, x_3 \cdot a_2(x_0, x_2) \land a_2(x_3, x_5) \]
\[ \cdot a_{2,3}(x_2, x_5) \]
\[ \cdot \forall z \cdot a_{1,2}(z, x_2) \Rightarrow (a_2(z, x_2) \lor a_2(z, x_3)) \]
Example: disjunctive chain queries – a rewriting

\[\sigma = \{a\}\]
\[\tau = \{a_2, a_{1,2}, a_{2,3}\}\]

\[\mathbf{V} = \left\{\begin{array}{l}
Q_2(x, y) = x \xrightarrow{a^2} y \\
Q_{1,2}(x, y) = (x \xrightarrow{a} y) \lor (x \xrightarrow{a^2} y) \\
Q_{2,3}(x, y) = (x \xrightarrow{a^2} y) \lor (x \xrightarrow{a^3} y)
\end{array}\right\}\]

Does \(\mathbf{V}\) determine \(Q(x, y) = x \xrightarrow{a^5} y\)?

\[R(x_0, x_5) = \exists x_2, x_3 \cdot a_2(x_0, x_2) \land a_2(x_3, x_5)\]
\[\cdot a_{2,3}(x_2, x_5)\]
\[\cdot \forall z \cdot a_{1,2}(z, x_2) \Rightarrow (a_2(z, x_2) \lor a_2(z, x_3))\]
Example: disjunctive chain queries – a rewriting

\[ \sigma = \{ a \} \]
\[ \tau = \{ a_2, a_{1,2}, a_{2,3} \} \]

\[ V = \begin{cases} 
Q_2(x, y) = x \xrightarrow{a^2} y \\
Q_{1,2}(x, y) = (x \xrightarrow{a} y) \lor (x \xrightarrow{a^2} y) \\
Q_{2,3}(x, y) = (x \xrightarrow{a^2} y) \lor (x \xrightarrow{a^3} y) 
\end{cases} \]

Does \( V \) determine \( Q(x, y) = x \xrightarrow{a^5} y \)?

\[ R(x_0, x_5) = \exists x_2, x_3 \cdot a_2(x_0, x_2) \land a_2(x_3, x_5) \]
\[ \cdot a_{2,3}(x_2, x_5) \]
\[ \cdot \forall z \cdot a_{1,2}(z, x_2) \Rightarrow (a_2(z, x_2) \lor a_2(z, x_3)) \]
Example: disjunctive chain queries – a rewriting

\[ \sigma = \{ a \} \]
\[ \tau = \{ a_2, a_{1,2}, a_{2,3} \} \]

\[ V = \begin{cases} 
Q_2(x, y) = x \xrightarrow{a^2} y \\
Q_{1,2}(x, y) = (x \xrightarrow{a} y) \lor (x \xrightarrow{a^2} y) \\
Q_{2,3}(x, y) = (x \xrightarrow{a^2} y) \lor (x \xrightarrow{a^3} y) 
\end{cases} \]

Does \( V \) determine \( Q(x, y) = x \xrightarrow{a^5} y \)?

\[ R(x_0, x_5) = \exists x_2, x_3 \cdot a_2(x_0, x_2) \land a_2(x_3, x_5) \]
\[ \cdot a_{2,3}(x_2, x_5) \]
\[ \cdot \forall z \cdot a_{1,2}(z, x_2) \Rightarrow (a_2(z, x_2) \lor a_2(z, x_3)) \]
Example: disjunctive chain queries – a rewriting

\[ \sigma = \{ a \} \]
\[ \tau = \{ a_2, a_{1,2}, a_{2,3} \} \]
\[ V = \left\{ \begin{array}{l}
Q_2(x, y) = x \xleftarrow{a^2} y \\
Q_{1,2}(x, y) = (x \xleftarrow{a} y) \lor (x \xleftarrow{a^2} y) \\
Q_{2,3}(x, y) = (x \xleftarrow{a^2} y) \lor (x \xleftarrow{a^3} y)
\end{array} \right\} \]

Does \( V \) determine \( Q(x, y) = x \xleftarrow{a^5} y \)?

\[ R(x_0, x_5) = \exists x_2, x_3 \quad \cdot \quad a_2(x_0, x_2) \land a_2(x_3, x_5) \]
\[ \quad \cdot \quad a_{2,3}(x_2, x_5) \]
\[ \quad \cdot \quad \forall z \cdot a_{1,2}(z, x_2) \Rightarrow (a_2(z, x_2) \lor a_2(z, x_3)) \]
Example: disjunctive chain queries – a rewriting

\[
\sigma = \{a\} \\
\tau = \{a_2, a_{1,2}, a_{2,3}\}
\]

\[
V = \begin{cases} 
Q_2(x, y) = x \xrightarrow{a^2} y \\
Q_{1,2}(x, y) = (x \xrightarrow{a} y) \lor (x \xrightarrow{a^2} y) \\
Q_{2,3}(x, y) = (x \xrightarrow{a^2} y) \lor (x \xrightarrow{a^3} y)
\end{cases}
\]

Does \( V \) determine \( Q(x, y) = x \xrightarrow{a^5} y \)?

\[
R(x_0, x_5) = \exists x_2, x_3 \cdot a_2(x_0, x_2) \land a_2(x_3, x_5) \\
\cdot a_{2,3}(x_2, x_5) \\
\cdot \forall z \cdot a_{1,2}(z, x_2) \Rightarrow (a_2(z, x_2) \lor a_2(z, x_3))
\]
Example: disjunctive chain queries – homework

\[ \sigma = \{ a \} \]
\[ \tau = \{ a_2, a_{1,2}, a_{2,5} \} \]
\[ \mathbf{V} = \left\{ \begin{array}{l}
Q_2(x, y) = x \xrightarrow{a^2} y \\
Q_{1,2}(x, y) = (x \xrightarrow{a} y) \lor (x \xrightarrow{a^2} y) \\
Q_{2,5}(x, y) = (x \xrightarrow{a^2} y) \lor (x \xrightarrow{a^5} y)
\end{array} \right\} \]

Can you prove that \( \mathbf{V} \) does not determine \( Q(x, y) = x \xrightarrow{a^9} y \)?
Example: disjunctive chain queries – homework

\[ \sigma = \{ a \} \]
\[ \tau = \{ a_2, a_{1,2}, a_{2,5} \} \]

\[ V = \begin{cases} 
Q_2(x, y) = x \xrightarrow{a^2} y \\
Q_{1,2}(x, y) = (x \xrightarrow{a} y) \lor (x \xrightarrow{a^2} y) \\
Q_{2,5}(x, y) = (x \xrightarrow{a^2} y) \lor (x \xrightarrow{a^5} y) 
\end{cases} \]

Can you prove that \( V \) does not determine \( Q(x, y) = x \xrightarrow{a^9} y \)?

I know a proof...
Example: disjunctive chain queries – homework

\[ \sigma = \{ a \} \]
\[ \tau = \{ a_2, a_{1,2}, a_{2,5} \} \]

\[ V = \begin{cases} 
Q_2(x, y) = x \xrightarrow{a^2} y \\
Q_{1,2}(x, y) = (x \xrightarrow{a} y) \vee (x \xrightarrow{a^2} y) \\
Q_{2,5}(x, y) = (x \xrightarrow{a^2} y) \vee (x \xrightarrow{a^5} y) 
\end{cases} \]

Can you prove that \( V \) does not determine \( Q(x, y) = x \xrightarrow{a^9} y \)?

I know a proof...

...and it’s ugly...
Example: disjunctive chain queries – homework

\[
\begin{align*}
\sigma &= \{ a \} \\
\tau &= \{ a_2, a_{1,2}, a_{2,5} \}
\end{align*}
\]

\[
V = \left\{ \begin{array}{l}
Q_2(x, y) = x \xrightarrow{a^2} y \\
Q_{1,2}(x, y) = (x \xrightarrow{a} y) \lor (x \xrightarrow{a^2} y) \\
Q_{2,5}(x, y) = (x \xrightarrow{a^2} y) \lor (x \xrightarrow{a^5} y)
\end{array} \right\}
\]

Can you prove that \( V \) does not determine \( Q(x, y) = x \xrightarrow{a^9} y \)?

I know a proof...

...and it’s ugly...

If you think you have an elegant proof, come talk to me!
Announcement

If you want to know more...
Announcement

If you want to know more... come work with us!
Announcement

If you want to know more... come work with us!
1-year postdoc funding at Marne-la-Vallée
Announcement

If you want to know more... come work with us!
1-year postdoc funding at Marne-la-Vallée

Thank you!