Equivalences between
Data Dependencies in Nested Databases and
Fragments of Propositional Logic

Sebastian Link

Information Science Research Centre, Dept of Information Systems
Massey University, New Zealand

This research is supported by Marsden Funding, Royal Society of New Zealand.
An Outline of the Talk

(1) Dependencies in Relational Databases

(2) An abstract Data Model for Nested Databases

(3) Identifying Nested Data Elements

(4) Data Dependencies in Nested Databases

(5) Equivalences for FDs, MVDs and Boolean Dependencies

(6) Reusing Relational Tools & Time-Complexity of Implication Problems
1.1 Dependency Implication

- relation schema: finite set $R$ with domains $\text{dom}(A)$ for all $A \in R$
- $\Sigma \cup \{\sigma\}$ finite set of dependencies on $R$:
  \[
  \Sigma \models \sigma \iff \forall r \subseteq \text{dom}(R). \text{ if } \models_r \Sigma, \text{ then } \models_r \sigma
  \]
- $\mathcal{C}$-implication problem:
  For any $R$ and any $\Sigma \cup \{\sigma\}$ in $\mathcal{C}$, does $\Sigma$ imply $\sigma$?
- data dependency implication crucial in database design:
  - $\sigma$ doesn’t need to be specified additionally to $\Sigma$ iff $\Sigma \models \sigma$
  - avoid redundancy in specification and consistency checking routines
  - query optimisation
  - database normalisation (BCNF, 3NF, 4NF, etc.)
- characterising dependency implication in logical terms gives insight and allows to apply tools from artificial intelligence
1.2 FDs and Horn clauses

- **Movie-database:**

<table>
<thead>
<tr>
<th>Title</th>
<th>Year</th>
<th>Director</th>
<th>Writer</th>
<th>Actor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memoirs of a Geisha</td>
<td>2005</td>
<td>Rob Marhall</td>
<td>Arthur Golden</td>
<td>Ken Watanabe</td>
</tr>
</tbody>
</table>

- database satisfies \( Title, \ Year \rightarrow Director, \ Writer \)
- database violates \( Title, \ Year \rightarrow Actor \)
- FD \( X \rightarrow Y \) with \( X, Y \subseteq R \) satisfied by \( R \)-database \( r \) iff
  \[
  \forall t, t' \in r. \text{ if } t[X] = t'[X], \text{ then } t[Y] = t'[Y]
  \]
- Horn clause: finite disjunction of literals (at most one positive)
  \[
  \neg V_1 \lor \cdots \lor \neg V_n \lor V
  \]
- Horn clause implication decidable in linear time
1.3 Linking FDs and Horn Clauses

- consider $\text{LECTURE} = \{\text{Class, Lecturer, Time, Room}\}$ with $\Sigma$ of FDs:
  
  $\text{Class} \rightarrow \text{Lecturer}$, and $\text{Class, Time} \rightarrow \text{Room}$, and 
  $\text{Lecturer, Time} \rightarrow \text{Class}$, and $\text{Room, Time} \rightarrow \text{Class}$

- FD $\sigma$: $\text{Class, Lecturer, Room } \rightarrow \text{Time}$ is not implied by $\Sigma$
  
  $\{(\text{Databases, H. Simpson, 2:30pm, 3.12}), (\text{Databases, H. Simpson, 4:30pm, 3.12})\}$

- $\textit{Fagin’77}$: FD implication is equivalent to Horn clause implication:
  
  $\neg \text{Class} \lor \neg \text{Lecturer} \lor \neg \text{Room} \lor \text{Time}$  or  
  $(\text{Class} \land \text{Lecturer} \land \text{Room}) \Rightarrow \text{Time}$

- define $\theta$ by $\theta(V) = true$ iff $V \in \{\text{Class, Lecturer, Room}\}$

- $\theta$ satisfies all Horn clauses in $\Sigma$ but violates $\sigma$
1.4 Multivalued Dependencies

- **DVD-database:**

<table>
<thead>
<tr>
<th>Movie</th>
<th>Role</th>
<th>Actor</th>
<th>Feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memoirs of a Geisha</td>
<td>Sayuri</td>
<td>Ziyi Zhang</td>
<td>Deleted Scene</td>
</tr>
<tr>
<td>Memoirs of a Geisha</td>
<td>Chairman</td>
<td>Ken Watanabe</td>
<td>Making of</td>
</tr>
<tr>
<td>Memoirs of a Geisha</td>
<td>Sayuri</td>
<td>Ziyi Zhang</td>
<td>Production Note</td>
</tr>
<tr>
<td>Memoirs of a Geisha</td>
<td>Chairman</td>
<td>Ken Watanabe</td>
<td>Deleted Scene</td>
</tr>
<tr>
<td>Memoirs of a Geisha</td>
<td>Sayuri</td>
<td>Ziyi Zhang</td>
<td>Making of</td>
</tr>
<tr>
<td>Memoirs of a Geisha</td>
<td>Chairman</td>
<td>Ken Watanabe</td>
<td>Production Note</td>
</tr>
</tbody>
</table>

- database violates 
  \[ Movie \to Role \text{ and } Movie \to Actor \text{ and } Movie \to Feature \]

- database satisfies \[ Movie \to Role, Actor \]

- MVD \( X \to Y \) with \( X, Y \subseteq R \) satisfied by \( R \)-database \( r \) iff

\[
r = r[XY] \bowtie r[X(R - Y)]
\]
1.5 MVDs and Normalisation

- Movie $\rightarrow$ Role, Actor suggests to decompose

<table>
<thead>
<tr>
<th>Movie</th>
<th>Role</th>
<th>Actor</th>
<th>Feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memoirs of a Geisha</td>
<td>Sayuri</td>
<td>Ziyi Zhang</td>
<td>Deleted Scene</td>
</tr>
<tr>
<td>Memoirs of a Geisha</td>
<td>Chairman</td>
<td>Ken Watanabe</td>
<td>Making of</td>
</tr>
<tr>
<td>Memoirs of a Geisha</td>
<td>Sayuri</td>
<td>Ziyi Zhang</td>
<td>Production Note</td>
</tr>
<tr>
<td>Memoirs of a Geisha</td>
<td>Chairman</td>
<td>Ken Watanabe</td>
<td>Deleted Scene</td>
</tr>
<tr>
<td>Memoirs of a Geisha</td>
<td>Sayuri</td>
<td>Ziyi Zhang</td>
<td>Making of</td>
</tr>
<tr>
<td>Memoirs of a Geisha</td>
<td>Chairman</td>
<td>Ken Watanabe</td>
<td>Production Note</td>
</tr>
</tbody>
</table>

into

<table>
<thead>
<tr>
<th>Movie</th>
<th>Role</th>
<th>Actor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memoirs of a Geisha</td>
<td>Sayuri</td>
<td>Ziyi Zhang</td>
</tr>
<tr>
<td>Memoirs of a Geisha</td>
<td>Chairman</td>
<td>Ken Watanabe</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Movie</th>
<th>Feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memoirs of a Geisha</td>
<td>Deleted Scene</td>
</tr>
<tr>
<td>Memoirs of a Geisha</td>
<td>Making of</td>
</tr>
<tr>
<td>Memoirs of a Geisha</td>
<td>Production Note</td>
</tr>
</tbody>
</table>
1.6 MVDs and a Fragment of Propositional Logic

- view *Movie* $\rightarrow$ *Role, Actor* on DVD=$\{\text{Title, Role, Actor, Feature}\}$
- *Movie* $\rightarrow$ *Actor, Feature* not implied by *Movie* $\rightarrow$ *Role, Actor*

<table>
<thead>
<tr>
<th>Movie</th>
<th>Role</th>
<th>Actor</th>
<th>Feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memoirs of a Geisha</td>
<td>Sayuri</td>
<td>Ziyi Zhang</td>
<td>Deleted Scene</td>
</tr>
<tr>
<td>Memoirs of a Geisha</td>
<td>Chairman</td>
<td>Ken Watanabe</td>
<td>Deleted Scene</td>
</tr>
</tbody>
</table>

- **Sagiv, Delobel, Parker, Fagin’81**: MVD-Implication equivalent to

$$
(U_1 \land \cdots \land U_k) \Rightarrow ((V_1 \land \cdots \land V_l) \lor (W_1 \land \cdots \land W_m))
$$

- define $\theta$ by $\theta(V) = \text{true}$ iff $V \in \{\text{Movie, Feature}\}$
- $\theta$ satisfies *Movie* $\Rightarrow ((\text{Role} \land \text{Actor}) \lor \text{Feature})$ but violates *Movie* $\Rightarrow ((\text{Actor} \land \text{Feature}) \lor \text{Role})$
1.7 Nested Databases

- schema: $\text{SHOP}(\text{Customer}, \text{BAG}(\text{ITEM}(\text{Article}, \text{Price})), \text{Discount})$

- instance:
  
  (Homer, $\langle(\text{Donut},1.5), (\text{Donut},1.5), (\text{Chocolate},2), (\text{Chocolate},2)\rangle$, 0) 
  
  (Bart, $\langle(\text{Donut},2), (\text{Donut},2), (\text{Chocolate},1.5), (\text{Chocolate},1.5)\rangle$, 1) 

- customers with the same bag of items should receive the same discount

  $\text{SHOP}(\text{BAG}(\text{ITEM}(\text{Article}, \text{Price})))) \rightarrow \text{SHOP}(\text{Discount})$

- fundamental problem again:
  What other dependencies are implied by those specified?

- major objective:
  Logical characterisation of dependency implication in nested databases
2.1 Database Schemata: Nested Attributes

- capture characteristics of objects in target database by attributes

\[ N := A | \lambda | L(N, \ldots, N) | L[N] | L\{N\} | L\langle N \rangle \]

- examples:
  - \text{SHOP}(\text{Customer}, \text{BAG}\langle \text{ITEM}(\text{Article}, \text{Price})\rangle, \text{Discount})
  - \text{HALFTONING}(\text{Brightness}, \text{INPUT}[\text{Level}], \text{OUTPUT}[\text{Bit}])
  - \text{SOCCER}\{\text{MATCH}(\text{Winner}, \text{Loser})\}
  - \text{NUMBERS}[\text{REPRESENTATION}(\text{Prime})]
  - \text{LECTURE}(\text{Class}, \text{Lecturer}, \text{Time}, \text{Room})
2.2 Database Instances: Domain Assignment

- extend \( \textit{dom} \) from flat to nested attributes (\( \textit{dom}(\lambda) = \{\text{ok}\} \))
- examples for nested tuples:
  - \( \text{SHOP}(\text{Customer}, \text{BAG}(\text{ITEM}(\text{Article}, \text{Price})), \text{Discount}) \):
    - \((\text{Homer}, \langle (\text{Donut}, 1.5), (\text{Donut}, 1.5), (\text{Chocolate}, 2), (\text{Chocolate}, 2) \rangle, 0) \)
    - \((\text{Bart}, \langle (\text{Donut}, 2), (\text{Donut}, 2), (\text{Chocolate}, 1.5), (\text{Chocolate}, 1.5) \rangle, 1) \)
  - \( \text{SOCCE}\{\text{MATCH}(\text{Winner}, \text{Loser})\} \):
    - \((\text{Denmark, Sweden}), (\text{New Zealand, Australia})\) 
    - \((\text{Mexico, USA}), (\text{Brazil, Argentina}), (\text{Brazil, USA})\) 

- \textbf{RDM}: single application of record constructor
- \textbf{Nested Relational Data Model}: record and set constructor
- \textbf{Object-oriented Data Models}: record, set, multiset and list constructor
2.3 Subschemata: Subattributes

- recursively replacing attributes by $\lambda$ gives different layers of info:
- some *subattributes* of
  \[ \text{SHOP}(\text{Customer}, \text{Bag}\langle \text{Item}(\text{Article}, \text{Price})\rangle, \text{Discount}) : \]
  - \[ \text{SHOP}(\lambda, \text{Bag}\langle \text{Item}(\text{Article}, \text{Price})\rangle, \text{Discount}) \]
  - \[ \text{SHOP}(\text{Customer}, \text{Bag}\langle \text{Item}(\lambda, \lambda)\rangle, \text{Discount}) \]
  - \[ \text{SHOP}(\lambda, \text{Bag}\langle \text{Item}(\text{Article}, \lambda)\rangle, \lambda) \]
  - \[ \text{SHOP}(\text{Customer}, \lambda, \text{Discount}) \]
- subattributes of \text{HALFTONING}(\text{Brightness}, \text{INPUT}[\text{Level}], \text{OUPTUT}[\text{Bit}]):
2.4 Subattributes: Formal Definition

- formally:
  define subattribute relation $\leq$ on nested attributes (partial order) by
  
  - $N \leq N$ for all nested attributes $N \in \mathcal{N}A$,
  - $\lambda \leq A$ for all flat attributes $A \in \mathcal{U}$,
  - $\lambda \leq N$ for all set-, multiset- and list-valued attributes $N \in \mathcal{N}A$,
  - $L(N_1, \ldots, N_k) \leq L(M_1, \ldots, M_k)$, if $N_i \leq M_i$ for all $i = 1, \ldots, k$,
  - $L\{N\} \leq L\{M\}$, if $N \leq M$,
  - $L\langle N\rangle \leq L\langle M\rangle$, if $N \leq M$,
  - $L[N] \leq L[M]$, if $N \leq M$

- $\leq$ induces Brouwerian algebra $\langle Sub(N), \leq, \sqcup, \sqcap, \sqsetminus, \lambda_N \rangle$ on set $Sub(N)$ of subattributes on $N$
2.5 Database Transformations: Projection Function

- subattributes represent at most as much info as their superattributes
- formally: for $M \leq N$ there is projection $\pi^N_M : \text{dom}(N) \rightarrow \text{dom}(M)$
- $N = \text{SHOP}(\text{Customer}, \text{Bag}\langle \text{ITEM}(\text{Article}, \text{Price})\rangle, \text{Discount})$ with
  
  $t = (\text{Bart}, \langle (\text{Donut}, 2), (\text{Donut}, 2), (\text{Chocolate}, 1.5), (\text{Chocolate}, 1.5) \rangle, 1)$

- $M = \text{SHOP}(\text{Customer}, \text{Bag}\langle \text{ITEM}(\lambda, \text{Price})\rangle, \text{Discount})$
  
  $\pi^N_M(t) = (\text{Bart}, \langle (\text{ok}, 2), (\text{ok}, 2), (\text{ok}, 1.5), (\text{ok}, 1.5) \rangle, 1)$

- $M = \text{SHOP}(\lambda, \text{Bag}\langle \text{ITEM}(\lambda, \lambda)\rangle, \text{Discount})$
  
  $\pi^N_M(t) = (\text{ok}, \langle (\text{ok}, \text{ok}), (\text{ok}, \text{ok}), (\text{ok}, \text{ok}), (\text{ok}, \text{ok}) \rangle, 1)$
2.6 Projections on Halftoning

(Brightness, Input[Level], Output[Bit])
3.1 Join-Irreducibles

- to store tuples in relational database we store their values on attributes
- $a \in L$ of lattice $(L, \sqsubseteq, \sqcup, \sqcap, 0)$ is *join-irreducible* iff $a \neq 0$ and if $a = b \sqcup c$ holds for any $b, c \in L$, then $a = b$ or $a = c$
- let $\mathcal{B}(N)$ denote the join-irreducibles of $(\text{Sub}(N), \leq, \sqcup, \sqcap, \lambda_N)$
- What subattributes identify nested data elements in presence of type constructors?
- $\text{Bag}\langle \text{Item}(\text{Article, Price}) \rangle$:
  \[
  \langle (\text{Donut}, 1.5), (\text{Donut}, 1.5), (\text{Chocolate}, 2), (\text{Chocolate}, 2) \rangle
  \]
  \[
  \langle (\text{Donut}, 2), (\text{Donut}, 2), (\text{Chocolate}, 1.5), (\text{Chocolate}, 1.5) \rangle
  \]
- join-irreducibles are sufficient in presence of records and lists
- what is needed in presence of sets or multisets?
3.2 Reconcilability

- $X, Y \in \text{Sub}(N)$ reconcilable iff one of the following holds:
  - $Y \leq X$ or $X \leq Y$,
  - $N = L(N_1, \ldots, N_k)$, $X = L(X_1, \ldots, X_k)$, $Y = L(Y_1, \ldots, Y_k)$ where $X_i$ and $Y_i$ are reconcilable for all $i = 1, \ldots, k$,
  - $N = L[N']$, $X = L[X']$, $Y = L[Y']$ where $X'$ and $Y'$ reconcilable

- Theorem:
  $\forall N. \forall X, Y \in \text{Sub}(N). X$ and $Y$ reconcilable if and only if
  $\forall t, t' \in \text{dom}(N). (\pi_X^N(t) = \pi_X^N(t') \land \pi_Y^N(t) = \pi_Y^N(t')) \Rightarrow \pi_{X \cup Y}^N(t) = \pi_{X \cup Y}^N(t')$
3.3 Extended Join-Irreducibles

- not reconcilable are:
  \( \text{SHOP}(\lambda, \text{BAG}(\text{ITEM}(\text{Article}, \lambda)), \lambda) \), \( \text{SHOP}(\lambda, \text{BAG}(\text{ITEM}(\lambda, \text{Price})), \lambda) \)

- extended join-irreducibles form smallest \( \mathcal{E}(N) \subseteq \text{Sub}(N) \) such that
  1. \( \mathcal{B}(N) \subseteq \mathcal{E}(N) \), and
  2. for all \( X, Y \in \mathcal{E}(N) \) which are not reconcilable also \( X \sqcup Y \in \mathcal{E}(N) \)

- \( \text{SHOP}(\text{Customer}, \text{BAG}(\text{ITEM}(\text{Article}, \text{Price}))) \):

\[ \text{Diagram showing a lattice structure.} \]
4.1 FDs and MVDs in the Presence of Records and Lists

- **an FD on** \( N \) **is expression**

  \[ X \rightarrow Y \text{ where } X, Y \in Sub(N) \]

- **\( r \subseteq \text{dom}(N) \) satisfies** \( X \rightarrow Y \) **on** \( N \) \( (|=r X \rightarrow Y) \) **iff**

  \[ \forall t_1, t_2 \in \text{dom}(r) . \text{if } \pi_X^N(t_1) = \pi_X^N(t_2), \text{ then } \pi_Y^N(t_1) = \pi_Y^N(t_2) \]

- **an MVD on** \( N \) **is expression**

  \[ X \rightarrow Y \text{ where } X, Y \in Sub(N) \]

- **\( r \subseteq \text{dom}(N) \) satisfies** \( X \rightarrow Y \) **on** \( N \) \( (|=r X \rightarrow Y) \) **iff**

  \[ r = \pi_{X \sqcup Y}(r) \Join \pi_{X \sqcup Y}c(r) \]

- **note that**

  - \( \pi_X(r) = \{ \pi_X^N(t) \mid t \in r \} \) for \( r \subseteq \text{dom}(N) \)
  - \( r_1 \Join r_2 = \{ t \in \text{dom}(X \sqcup Y) \mid \exists t_1 \in r_1, t_2 \in r_2 . \pi_X^{X \sqcup Y}(t) = t_1 \text{ and } \pi_Y^{X \sqcup Y}(t) = t_2 \} \)

  for \( r_1 \subseteq \text{dom}(X) \), \( r_2 \subseteq \text{dom}(Y) \) and \( X, Y \in Sub(N) \)
4.2 An Example from Digital Halftoning

- convert continuous-tone image into binary one that looks similar
- input matrix $A$ represents digital (gray) image, where $a_{ij}$ represents brightness level of $(i, j)$-pixel in $N \times N$ pixel grid
- replace $A$ by $\{0, 1\}$-matrix $B$ that is good approximation of $A$
- typical family of regions $\mathcal{R}$:

  \[
  \begin{array}{|c|c|}
  \hline
  a & b \\
  \hline
  c & d \\
  \hline
  \end{array}
  \quad
  \begin{array}{|c|c|}
  a & b \\
  \hline
  b & d \\
  \hline
  \end{array}
  ,
  \]

- $B$ should minimise

  \[
  \left| \sum_{(i,j) \in R} a_{i,j} - \sum_{(i,j) \in R} b_{i,j} \right| \quad \text{for all } R \in \mathcal{R}
  \]
simple example: $\{0, \frac{1}{2}, 1\}$-input matrix

all inputs with overall brightness $\frac{1}{2}$ and length two, i.e. $[0, \frac{1}{2}]$ or $[\frac{1}{2}, 0]$, could be mapped to any of $[0,1]$, $[1,0]$ or $[0,0]$

all inputs with overall brightness $\frac{3}{2}$ and length four such as $[0,0,1,\frac{1}{2}]$ can be mapped to any of $[0,0,0,1]$, $[0,0,1,0]$, $[0,1,0,0]$, $[1,0,0,0]$, $[0,0,1,1]$, $[0,1,0,1]$, $[1,0,0,1]$, $[0,1,1,0]$, $[1,0,1,0]$, $[1,1,0,0]$

the set of input regions ($\{[0, \frac{1}{2}], [\frac{1}{2}, 0]\}$) is determined by the overall brightness of the input region ($\frac{1}{2}$) and the length of the input region (2), independently of the set of output regions ($\{[0, 1], [1, 0], [0, 0]\}$)
• database stores input and output regions as lists together with the overall brightness of the input region

• find \{0,1\}-matrix \( B \) that has for every regions of \( A \) a corresponding output region in database

\[
A = \begin{pmatrix}
  0 & 0 \\
  \frac{1}{2} & \frac{1}{2}
\end{pmatrix}
\]
has approximation \( B = \begin{pmatrix}
  0 & 0 \\
  0 & 1
\end{pmatrix} \)

• every \( 2 \times 2 \) matrix has 5 input regions; mappings producing \( B \) from \( A \) are: \([0,0] \leftrightarrow [0,0] \), \([\frac{1}{2}, \frac{1}{2}] \leftrightarrow [0,1] \), \([0, \frac{1}{2}] \leftrightarrow [0,0] \) (left column), \([0, \frac{1}{2}] \leftrightarrow [0,1] \) (right column) and \([0,0,\frac{1}{2},\frac{1}{2}] \leftrightarrow [0,0,0,1] \).

\[
\begin{pmatrix}
  0 & 1 \\
  0 & 0
\end{pmatrix}
\]
not a good approximation of \( A \) since \([\frac{1}{2}, \frac{1}{2}] \) should not be mapped to \([0,0]\)
• constraints that a database designer may choose to specify for this application are the following:

(1) the length of the input region determines the length of the output region, and vice versa

(2) the overall brightness and length of the input region together determine the set of all input regions independently from the set of the output regions
4.3 Formal Constraints for Digital Halftoning

- FDs the every legal database should satisfy are
  - \( \text{HALFTONING}(\lambda, \text{Input}[\lambda], \lambda) \rightarrow \text{HALFTONING}(\lambda, \lambda, \text{Output}[\lambda]) \)
  - \( \text{HALFTONING}(\lambda, \lambda, \text{Output}[\lambda]) \rightarrow \text{HALFTONING}(\lambda, \text{Input}[\lambda], \lambda) \)

- an FD which will not be satisfied by every legal database is
  - \( \text{HALFTONING}(\text{Brightness}, \text{Input}[\lambda], \lambda) \rightarrow \text{HALFTONING}(\lambda, \text{Input}[\text{Level}], \lambda) \)

- an MVD that every legal database should satisfy is
  - \( \text{HALFTONING}(\text{Brightness}, \text{Input}[\lambda], \lambda) \rightarrow \text{HALFTONING}(\lambda, \text{Input}[\text{Level}], \lambda) \)
4.4 FDs in the General Case

- an FD on $N$ is expression
  $$\mathcal{X} \rightarrow \mathcal{Y}$$ with non-empty $\mathcal{X}, \mathcal{Y} \subseteq Sub(N)$

- $r \subseteq dom(N)$ satisfies $\mathcal{X} \rightarrow \mathcal{Y}$ ($\models_r \mathcal{X} \rightarrow \mathcal{Y}$) iff $\forall t_1, t_2 \in dom(r)$.
  if $\forall X \in \mathcal{X}. \pi_X^N(t_1) = \pi_X^N(t_2)$, then $\forall Y \in \mathcal{Y}. \pi_Y^N(t_1) = \pi_Y^N(t_2)$

- for $\mathcal{X} \subseteq Sub(N)$ let
  $$\vartheta(\mathcal{X}) = \max\{Y \in \mathcal{E}(N) \mid Y \leq X \text{ for some } X \in \mathcal{X}\}$$

- $\models_r \mathcal{X} \rightarrow \mathcal{Y}$ if and only if $\models_r \vartheta(\mathcal{X}) \rightarrow \vartheta(\mathcal{Y})$

- examples:
  - $\text{SHOP}(\text{BAG}(\text{ITEM}(\text{Article, Price}))) \rightarrow \text{SHOP}(\text{Discount})$
  - $\text{SHOP}(\text{BAG}(\text{ITEM}(\text{Article}))), \text{SHOP}(\text{BAG}(\text{ITEM}(\text{Price}))) \rightarrow \text{SHOP}(\text{Discount})$


4.5 Boolean Dependencies

- **Boolean dependencies** on $N$ form smallest set $Bd(N)$ with:
  1. $\mathcal{E}(N) \subseteq Bd(N)$,
  2. if $X \in Bd(N)$, then $\neg X \in Bd(N)$,
  3. if $X, Y \in Bd(N)$, then $(X \land Y), (X \lor Y), (X \Rightarrow Y) \in Bd(N)$.

- $\sigma \in Bd(N), t_1, t_2 \in \text{dom}(N)$ distinct: $\models \{t_1, t_2\} \sigma$ iff
  1. if $\sigma = X \in \mathcal{E}(N)$, then $\models \{t_1, t_2\} \sigma$ iff $\pi^N_X(t_1) = \pi^N_X(t_2)$,
  2. if $\sigma = \neg \tau$ for $\tau \in Bd(N)$, then $\models \{t_1, t_2\} \sigma$ iff not $\models \{t_1, t_2\} \tau$,
  3. if $\sigma = \sigma_1 \land \sigma_2$ for $\sigma_1, \sigma_2 \in Bd(N)$, then $\models \{t_1, t_2\} \sigma$ if and only if $\models \{t_1, t_2\} \sigma_1$ and $\models \{t_1, t_2\} \sigma_2$.

- $\models_r \sigma$ iff $\forall t_1, t_2 \in r. \text{ if } t_1 \neq t_2, \text{ then } \models \{t_1, t_2\} \sigma$

- trivial $\text{Bd } \phi_N = \neg X_1 \lor \cdots \lor \neg X_k$ where $\max_{\leq} \mathcal{E}(N) = \{X_1, \ldots, X_k\}$
5.1 Example of Equivalence for FDs and MVDs - The Dependency Point of View

- consider $\text{HALFTONING}(\text{Brightness, INPUT}[\text{Level}], \text{OUTPUT}[\text{Bit}])$ together with the set $\Sigma$ of FDs and MVDs specified before, i.e.,

$$
\text{HALFTONING}(\text{Brightness, INPUT}[\lambda]) \rightarrow \text{HALFTONING}(\text{INPUT}[\text{Level}]),
$$

$$
\text{HALFTONING}(\text{INPUT}[\lambda]) \rightarrow \text{HALFTONING}(\text{OUTPUT}[\lambda]), \text{ and}
$$

$$
\text{HALFTONING}(\text{OUTPUT}[\lambda]) \rightarrow \text{HALFTONING}(\text{INPUT}[\lambda])
$$

- not implied by $\Sigma$ is the FD $\sigma$

$$
\text{HALFTONING}(\text{Brightness, INPUT}[\text{Level}]) \rightarrow \text{HALFTONING}(\text{OUTPUT}[\text{Bit}])
$$

- in fact,

$$
\{ \left( \frac{3}{2}, \left[ \frac{1}{2}, 0, 1, 0 \right], \left[ 1, 0, 1, 0 \right] \right), \left( \frac{3}{2}, \left[ \frac{1}{2}, 0, 1, 0 \right], \left[ 0, 1, 1, 0 \right] \right) \}
$$

satisfies all dependencies in $\Sigma$, but it does not satisfy $\sigma$
5.2 Example of Equivalence for FDs and MVDs - The Logical Point of View

- associated join-irreducibles with propositional variables:
  - \textsc{Halftoning}(Brightness) is \( V_1 \), \textsc{Halftoning}(\textsc{Input}[/\lambda]) is \( V_2 \),
  - \textsc{Halftoning}(\textsc{Input}[\text{Level}]) is \( V_3 \), \textsc{Halftoning}(\textsc{Output}[/\lambda]) is \( V_4 \), and
  - \textsc{Halftoning}(\textsc{Output}[\text{Bit}]) is \( V_5 \)

- truth assignments to \( V_1, \ldots, V_5 \) cannot be independent from one another:

\[ \Pi_N = \{ V_3 \Rightarrow V_2, V_5 \Rightarrow V_4 \} \]

- \( \Sigma \) mapped to \( \Pi = \{ V_1 \land V_2 \Rightarrow V_3 \lor V_5, V_2 \Rightarrow V_4, V_4 \Rightarrow V_2 \} \)

- \( \sigma \) mapped to \( \sigma' = V_1 \land V_3 \Rightarrow V_5 \)

- \( \theta(V_i) = \text{true} \) iff \( i \in \{1, 2, 3, 4\} \) satisfies \( \Pi \cup \Pi_N \), but violates \( \sigma' \)
5.3 Example of Equivalence for FDs in General Case -
The Dependency Point of View

- consider $\text{DANCE}(\text{Time}, \text{PART}\{\text{Name}\}, \text{DUO}\{\text{PAIR}(\text{Girl,Boy})\}, \text{Rating})$
together with the set $\Sigma$ of FDs:
  - $\text{DANCE}(\text{Time}) \rightarrow \text{DANCE}(\text{PART}\{\text{Name}\}, \text{DUO}\{\text{PAIR}(\text{Girl,Boy})\}, \text{Rating}),$
  - $\text{DANCE}(\text{PART}\{\text{Name}\}) \rightarrow \text{DANCE}(\text{DUO}\{\text{PAIR}(\text{Girl})\}), \text{DANCE}(\text{DUO}\{\text{PAIR}(\text{Boy})\})$
  - $\text{DANCE}(\text{DUO}\{\text{PAIR}(\text{Girl})\}), \text{DANCE}(\text{DUO}\{\text{PAIR}(\text{Boy})\}) \rightarrow \text{DANCE}(\text{PART}\{\text{Name}\})$
  - $\text{DANCE}(\text{DUO}\{\text{PAIR}(\text{Girl,Boy})\}) \rightarrow \text{DANCE}(\text{Rating})$

- not implied by $\Sigma$ is the FD $\sigma$
  
  $\text{DANCE}(\text{PART}\{\text{Name}\}) \rightarrow \text{DANCE}(\text{Rating})$

- in fact,
  
  $\{(24.12.05, \{\text{Lisa, Marge, Bart, Homer}\}, \{(\text{Lisa, Bart}), (\text{Marge, Homer})\}, 5),$
  $(25.12.05, \{\text{Lisa, Marge, Bart, Homer}\}, \{(\text{Lisa, Homer}), (\text{Marge, Bart})\}, 9)\}$

  satisfies all dependencies in $\Sigma$, but it does not satisfy $\sigma$
5.4 Example of Equivalence for FDs in General Case -  
The Logical Point of View

- extended join-irreducibles of \( N \) are mapped to propositional variables:
  - \( \text{DANCE} \text{(Time)} \) is \( V_1 \), \( \text{DANCE} \text{(PART\{Name\})} \) is \( V_2 \),
  - \( \text{DANCE} \text{(PART\{\lambda\})} \) is \( V_3 \), \( \text{DANCE} \text{(DUO\{PAIR(Girl,Boy)\})} \) is \( V_4 \),
  - \( \text{DANCE} \text{(DUO\{PAIR(Girl)\})} \) is \( V_5 \), \( \text{DANCE} \text{(DUO\{PAIR(Boy)\})} \) is \( V_6 \),
  - \( \text{DANCE} \text{(DUO\{\lambda\})} \) is \( V_7 \), and \( \text{DANCE} \text{(Rating)} \) is \( V_8 \)

- Horn clauses encoding the structure of \( N \) are

\[
\Pi_N = \{ V_2 \Rightarrow V_3, V_4 \Rightarrow V_5, V_4 \Rightarrow V_6, V_5 \Rightarrow V_7, V_6 \Rightarrow V_7 \}
\]

- \( \Sigma \) results in following Horn clauses

\[
\Pi = \{ V_1 \Rightarrow V_2, V_1 \Rightarrow V_4, V_1 \Rightarrow V_8, V_2 \Rightarrow V_5, V_2 \Rightarrow V_6, V_5 \land V_6 \Rightarrow V_2, V_4 \Rightarrow V_8 \}
\]

- \( \sigma \) corresponds to \( \sigma' = V_2 \Rightarrow V_8 \)

- \( \theta(V_i) = \text{true} \) iff \( i \in \{2, 3, 5, 6, 7\} \) satisfies \( \Pi \cup \Pi_N \), but violates \( \sigma' \)
5.5 Using Horn clauses to encode FDs and Structures

- Fagin’s idea: interpret attributes as propositional variables
- interpret extended join-irreducibles as variables via $\psi : E(N) \rightarrow V$
- the FD $\sigma = \mathcal{X} \rightarrow \mathcal{Y}$ where $\psi(\mathcal{X}) = \{X_1, \ldots, X_k\}$, and $\psi(\mathcal{Y}) = \{Y_1, \ldots, Y_m\}$ results in set $\Phi(\sigma)$ of $m$ Horn clauses
  \[
  \bigwedge_{i=1}^{k} \psi(X_i) \Rightarrow \psi(Y_1), \ldots, \bigwedge_{i=1}^{k} \psi(X_i) \Rightarrow \psi(Y_m)
  \]
- Horn clauses can also encode the structure of $N$
  \[
  \Pi_N = \{\psi(U) \Rightarrow \psi(V) \mid U, V \in E(N), U \text{ covers } V\}
  \]
5.6 Mapping MVDs and Boolean Dependencies

- consider the MVD $\sigma: X \rightarrow Y$ on $N$ where
  - $\vartheta(X) = \{X_1, \ldots, X_k\}$, $\vartheta(Y) = \{Y_1, \ldots, Y_m\}$, $\vartheta(Y^C - X) = \{Z_1, \ldots, Z_n\}$

- define $\Phi(\sigma)$ to be the Boolean propositional formula
  $$\psi(X_1) \land \cdots \land \psi(X_k) \Rightarrow (\psi(Y_1) \land \cdots \land \psi(Y_m)) \lor (\psi(Z_1) \land \cdots \land \psi(Z_n))$$

- Boolean dependency $\sigma$ mapped to propositional formula $\Phi(\sigma)$ by
  - if $\sigma = X \in \mathcal{E}(N)$, then let $\Phi(\sigma) = \psi(X)$,
  - if $\sigma = \neg \phi$, then $\Phi(\sigma) = \neg \Phi(\phi)$,
  - if $\sigma = \phi_1 \lor \phi_2$, then $\Phi(\sigma) = (\Phi(\phi_1) \lor \Phi(\phi_2))$,
  - if $\sigma = \phi_1 \land \phi_2$, then $\Phi(\sigma) = (\Phi(\phi_1) \land \Phi(\phi_2))$, and
  - if $\sigma = \phi_1 \Rightarrow \phi_2$, then $\Phi(\sigma) = (\Phi(\phi_1) \Rightarrow \Phi(\phi_2))$
5.7 Two-Element Instances and Truth Assignments

- Lemma:
  Let $\sigma$ be a Boolean dependency on the nested attribute $N$, and $r = \{t_1, t_2\} \subseteq \text{dom}(N)$ such that $t_1 \neq t_2$. Then $\models \sigma$ if and only if $\models_{\theta_r} \Phi(\sigma)$ where
  \[
  \theta_r(V) = \begin{cases} 
  \text{true}, & \text{if } \pi_{\psi^{-1}(V)}^N(t_1) = \pi_{\psi^{-1}(V)}^N(t_2) \\
  \text{false}, & \text{else}
  \end{cases}
  \]
  for all $V \in \psi(\mathcal{E}(N))$.

- Lemma:
  Assume that $r \subseteq \text{dom}(N)$ is some finite instance over $N$, $\Sigma$ a set of FDs and MVDs on $N$, and $\sigma$ a single FD or MVD on $N$. Suppose that $r$ satisfies all FDs and MVDs in $\Sigma$, but does not satisfy $\sigma$. Then there is some $r' = \{t_1, t_2\} \subseteq r$ such that $r'$ satisfies all FDs and MVDs in $\Sigma$, but does not satisfy $\sigma$. 

Dortmund, Germany, 02 March 2006
5.8 Equivalence for FDs and MVDs

- Theorem:
  Let \( N \) be a nested attribute, and \( \Sigma \cup \{\sigma\} \) either a set of FDs and MVDs on \( N \) or a set of FDs on \( N \). Let \( \Pi_N \) denote the propositional formulae which encode the structure of \( N \), and \( \Pi \) denote the corresponding set of propositional formulae for \( \Sigma \). Then

  1. \( \Sigma \) implies \( \sigma \),
  2. \( \Sigma \) implies \( \sigma \) in the world of two-element instances, and
  3. \( \Pi \cup \Pi_N \) logically implies \( \Phi(\sigma) \)

are equivalent.

- if \( \Sigma \cup \{\sigma\} \) is a set of FDs and MVDs on \( N \), then we implicitly assume that \( N \) is generated from flat attributes by applications of record and list constructor only.
5.9 Equivalence for Boolean Dependencies

• Theorem:
  Let $N$ be a nested attribute, and $\Sigma \cup \{\sigma\}$ a set of Boolean dependencies on $N$. Let $\Pi_N$ denote the propositional formulae which encode the structure of $N$, and $\Pi$ denote the corresponding set of propositional formulae for $\Sigma$. Then
  
  (1) $\Sigma$ implies $\sigma$,
  (2) $\Sigma$ implies $\sigma$ in the world of two-element instances, and
  (3) $\Pi \cup \Pi_N \cup \{\phi_N\}$ logically implies $\Phi(\sigma)$

  are equivalent. □

• extends well-known results from the relational model of data, where
  • only single application of record constructor allowed,
  • join-irreducibles form anti-chain, and
  • join-irreducibles (attributes) suffice
5.10 Example for Boolean Dependencies - The Dependency Point of View

- consider \( \text{PROFILE}(\text{Customer}, \text{Bag} \langle \text{Item}(\text{Article,Price}) \rangle, \text{Discount}) \) together with the functional dependency \( \sigma \)
  \[
  \text{PROFILE}(\text{Bag} \langle \text{Item}(\text{Article,Price}) \rangle) \rightarrow \text{PROFILE}(\text{Discount})
  \]
  \( \sigma \) implies the Boolean dependency \( \tau \)
  \[
  \neg \text{PROFILE}(\text{Customer}) \lor \neg \text{PROFILE}(\text{Bag} \langle \text{Item}(\text{Article,Price}) \rangle)
  \]
- two distinct elements that agree on \( \text{PROFILE}(\text{Customer}) \) differ on \( \text{PROFILE}(\text{Bag} \langle \text{Item}(\text{Article,Price}) \rangle) \) since they would not be distinct otherwise by \( \sigma \)
- distinct elements agreeing on \( \text{PROFILE}(\text{Bag} \langle \text{Item}(\text{Article,Price}) \rangle) \) (and therefore on \( \text{PROFILE}(\text{Discount}) \)), differ on \( \text{PROFILE}(\text{Customer}) \)
5.11 Example for Boolean Dependencies - The Dependency Point of View

- Extended join-irreducibles are mapped as follows:
  - PROFILE(Customer) is $V_1$, PROFILE(BAG(Item($\lambda$, $\lambda$))) is mapped to $V_2$,
  - PROFILE(BAG(Item(Article, $\lambda$))) is $V_3$, PROFILE(BAG(Item($\lambda$, Price))) is $V_4$,
  - PROFILE(BAG(Item(Article, Price))) is $V_5$, PROFILE(Discount) is $V_6$

- $\sigma'$ is $V_5 \Rightarrow V_6$, and $\tau'$ is $\neg V_1 \lor \neg V_5$, and $\phi_N$ is $\neg V_1 \lor \neg V_5 \lor \neg V_6$

- $\Pi_N = \{ V_5 \Rightarrow V_3, V_5 \Rightarrow V_4, V_3 \Rightarrow V_2, V_4 \Rightarrow V_2 \}$

- $\theta(V_i) = true$ for $1 \leq i \leq 6$ shows that $\tau'$ is not implied by $\Pi_N \cup \{ \sigma' \}$

- However, $\Pi_N \cup \{ \sigma', \phi_N \}$ implies $\tau'$
5.12 Another Example

• bijection $\psi$:

\[
\begin{align*}
\text{SHOP}(\text{Customer}) & \leftrightarrow V_1, \\
\text{SHOP}(\text{Bag}\langle \text{Item}(\text{Article},\text{Price})\rangle) & \leftrightarrow V_2, \\
\text{SHOP}(\text{Bag}\langle \text{Item}(\text{Article})\rangle) & \leftrightarrow V_3, \\
\text{SHOP}(\text{Bag}\langle \text{Item}(\text{Price})\rangle) & \leftrightarrow V_4, \\
\text{SHOP}(\text{Bag}\langle \text{Item}(\lambda,\lambda)\rangle) & \leftrightarrow V_5, \\
\text{SHOP}(\text{Discount}) & \leftrightarrow V_6
\end{align*}
\]

• $\text{SHOP}(\text{Bag}\langle \text{Item}(\text{Article},\text{Price})\rangle) \rightarrow \text{SHOP}(\text{Discount})$ doesn’t imply $\text{SHOP}(\text{Bag}\langle \text{Item}(\text{Article})\rangle), \text{SHOP}(\text{Bag}\langle \text{Item}(\text{Price})\rangle) \rightarrow \text{SHOP}(\text{Discount})$

  (Homer, $\langle (\text{Donut}, 1.5), (\text{Donut}, 1.5), (\text{Chocolate}, 2), (\text{Chocolate}, 2) \rangle$, 0)
  (Bart, $\langle (\text{Donut}, 2), (\text{Donut}, 2), (\text{Chocolate}, 1.5), (\text{Chocolate}, 1.5) \rangle$, 1)

• $\{V_2 \Rightarrow V_6, V_2 \Rightarrow V_3, V_2 \Rightarrow V_4, V_3 \Rightarrow V_5, V_4 \Rightarrow V_5\}$ doesn’t imply $V_3 \land V_4 \Rightarrow V_6$

\[
\theta(V_i) = \text{true} \quad \text{iff} \quad i \in \{3, 4, 5\}
\]
6.1 Mappings to Relation Schemata

- idea: interpret (extended) join-irreducibles as flat attributes
- FDs $\sigma: X \rightarrow Y$ mapped to $\sigma': \vartheta(X) \rightarrow \vartheta(Y)$, and MVD $\sigma: X \rightarrow Y$ mapped to MVD $\sigma': \vartheta(X) \rightarrow \vartheta(Y)$
- Theorem: $R_N = \mathcal{B}(N)$, $\Sigma \cup \{\sigma\}$ set of FDs and MVDs on $N$
  \[
  \Sigma' = \{\tau' \mid \tau \in \Sigma\} \cup \{U \rightarrow V \mid U, V \in \mathcal{B}(N), U \text{ covers } V\}
  \]
  Then $\sigma$ is implied by $\Sigma$ if and only if $\sigma'$ is implied by $\Sigma'$ on $R_N$. $\square$
- for $\sigma: \mathcal{X} \rightarrow \mathcal{Y}$ let $\sigma': \vartheta(\mathcal{X}) \rightarrow \vartheta(\mathcal{Y})$
- Theorem: $\Sigma \cup \{\sigma\}$ set of FDs, $R_N = \mathcal{E}(N)$ and
  \[
  \Sigma' = \{\tau' \mid \tau \in \Sigma\} \cup \{U \rightarrow V \mid U, V \in \mathcal{E}(N), U \text{ covers } V\}
  \]
  Then $\sigma$ is implied by $\Sigma$ if and only if $\sigma'$ is implied by $\Sigma'$ on $R_N$. $\square$
6.2 An Example

- join-irreducibles mapped to attribute names as follows:
  - \text{HALFTONING}(\text{Brightness}) \text{ is } A, \text{HALFTONING}(\text{INPUT}[\lambda]) \text{ is } B,
  - \text{HALFTONING}(\text{INPUT}[\text{Level}]) \text{ is } C, \text{HALFTONING}(\text{OUTPUT}[\lambda]) \text{ is } D,
  - \text{HALFTONING}(\text{OUTPUT}[\text{Bit}]) \text{ is } E

- set $\Sigma'$ of FDs and MVDs on $R_N = \{A, B, C, D, E\}$ is
  $$\Sigma' = \{AB \to C, B \to D, D \to B, C \to B, E \to D\}$$

- $\Sigma \models \text{HALFTONING}(\text{Brightness, OUTPUT}[\lambda]) \rightarrow \text{HALFTONING}(\text{OUTPUT}[\text{Bit}])$?
  $\Sigma \models \text{HALFTONING}(\text{Brightness, OUTPUT}[\lambda]) \rightarrow \text{HALFTONING}(\text{OUTPUT}[\text{Bit}])$?

- equivalent: Are $A, D \rightarrow E$ and $A, D \rightarrow E$ implied by $\Sigma'$?

- $DepB(AD) = \{A, B, C, D, E\}$ and $(AD)^+ = \{A, D, E\}$ wrt $\Sigma'$

- $E \notin A^+$ and $E \in DepB(AD): \Sigma' \not\models AD \rightarrow E \text{ and } \Sigma' \models AD \rightarrow E$
6.3 The Complexity of Implication Problems

- for FDs in the presence of records and lists: $\mathcal{O}(n)$ where $n$ denotes the total number of join-irreducibles of $N$ that occur in FDs $\vartheta(X) \rightarrow \vartheta(Y)$ in $\Sigma$.

- for FDs in the presence of records, lists, sets and multisets: $\mathcal{O}(n)$ where $n$ denotes the total number of extended join-irreducibles of $N$ that occur in FDs $\vartheta(X) \rightarrow \vartheta(Y)$ in $\Sigma$.

- for FDs and MVDs in the presence of records and lists: $\mathcal{O}(n \log n)$ where $n$ denotes the total number of join-irreducibles of $N$ that occur in FDs $\vartheta(X) \rightarrow \vartheta(Y)$ and MVDs $\vartheta(X) \rightarrow \vartheta(Y)$ in $\Sigma$.

- for a fragment of Boolean dependencies: can be solved in the same time as the corresponding implication problem for the associated class of Boolean propositional formulae, i.e., it is NP-complete in the most general case.
7 Conclusion and Future Work

- framework of nested attributes allows to capture data models by including corresponding type constructors
- theory of Brouwerian algebras can be used to extend many achievements from relational databases
- allows to study direct impact of type constructor on design problem without considering peculiarities of specific data model
- logical characterisation of dependency implication gives further insight, extends relational theory, and allows to cover many new application domains
- study different classes of dependencies in different combinations of constructors
- increase expressiveness by studying embedded dependencies (allowing several Brouwerian algebras simultaneously)
- normal forms