Presentation on Functional Dependencies CS x265

This presentation assumes that you have previously viewed
Watch videos (90 min) and answer questions from DB8
Relational Design Theory
(https://class.stanford.edu/courses/DB/RD/SelfPaced/courseware/ch-relational_design_theory/)
and/or read Chapter 3 of Ullman and Widom, Introduction to Database Management Systems
Assume the following relational schema covering vehicle ownership data (forgive lack of key, or assume that all attributes form the key, for now).

```
Name    Addr     SSN      VRN     Type      Make       Model
(N)       (A)       (S)         (V)        (T)        (Ma)         (Mo)
```

A “mega” relation Owners

Individual persons, uniquely identified by SSN, are stored with their Name and Addr(ess), and are stored with information of the vehicles they own, where each vehicle is uniquely identified by a Vehicle Registration Number (VRN), its Type (auto, truck, motorcycle), Manufacturer (aka Make), and Model. A sample of a database fitting this schema is below.

<table>
<thead>
<tr>
<th>N</th>
<th>A</th>
<th>S</th>
<th>V</th>
<th>T</th>
<th>Ma</th>
<th>Mo</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fred</td>
<td>Nashville</td>
<td>123</td>
<td>987</td>
<td>Truck</td>
<td>Ford</td>
<td>Ranger</td>
</tr>
<tr>
<td>Sri</td>
<td>NewYork</td>
<td>234</td>
<td>876</td>
<td>Car</td>
<td>Toyota</td>
<td>Camry</td>
</tr>
<tr>
<td>Gabriel</td>
<td>Nashville</td>
<td>345</td>
<td>765</td>
<td>MotorCy</td>
<td>Harley</td>
<td>Hog</td>
</tr>
<tr>
<td>Fred</td>
<td>Nashville</td>
<td>123</td>
<td>654</td>
<td>Car</td>
<td>VW</td>
<td>Bug</td>
</tr>
</tbody>
</table>

Given your domain knowledge of vehicle ownership relationships in the real world, list functional dependencies that you believe should be asserted as true/required of this relational schema (and be enforced in the database).
Assume the following functional dependencies apply to the schema

\[
S \rightarrow N, \quad S \rightarrow A, \quad V \rightarrow T, \quad V \rightarrow Ma, \quad V \rightarrow Mo, \quad Mo \rightarrow Ma, \quad Mo \rightarrow T, \quad V \rightarrow S
\]

<table>
<thead>
<tr>
<th>Name</th>
<th>Addr</th>
<th>SSN</th>
<th>VRN</th>
<th>Type</th>
<th>Make</th>
<th>Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fred</td>
<td>Nashville</td>
<td>123</td>
<td>987</td>
<td>Truck</td>
<td>Ford</td>
<td>Ranger</td>
</tr>
<tr>
<td>Sri</td>
<td>NewYork</td>
<td>234</td>
<td>876</td>
<td>Car</td>
<td>Toyota</td>
<td>Camry</td>
</tr>
<tr>
<td>Gabriel</td>
<td>Nashville</td>
<td>345</td>
<td>765</td>
<td>MotorCy</td>
<td>Harley</td>
<td>Hog</td>
</tr>
<tr>
<td>Fred</td>
<td>Nashville</td>
<td>123</td>
<td>654</td>
<td>Car</td>
<td>VW</td>
<td>Bug</td>
</tr>
<tr>
<td>Sri</td>
<td>NewYork</td>
<td>234</td>
<td>654</td>
<td>Car</td>
<td>VW</td>
<td>Bug</td>
</tr>
<tr>
<td>Mary</td>
<td>LosAngeles</td>
<td>456</td>
<td>876</td>
<td>Car</td>
<td>Toyota</td>
<td>Corolla</td>
</tr>
<tr>
<td>Fred</td>
<td>Nshville</td>
<td>123</td>
<td>543</td>
<td>Car</td>
<td>Honda</td>
<td>Accord</td>
</tr>
</tbody>
</table>

Each of the final three rows cause a violation of at least one FD (given the earlier rows). Identify the FDs that are violated in each case.
Assume the following functional dependencies apply to the schema

\[
S \rightarrow N, \quad S \rightarrow A, \quad V \rightarrow T, \quad V \rightarrow Ma, \quad V \rightarrow Mo, \quad Mo \rightarrow Ma, \quad Mo \rightarrow T, \quad V \rightarrow S
\]

<table>
<thead>
<tr>
<th>Name</th>
<th>Addr</th>
<th>SSN</th>
<th>VRN</th>
<th>Type</th>
<th>Make</th>
<th>Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fred</td>
<td>Nashville</td>
<td>123</td>
<td>987</td>
<td>Truck</td>
<td>Ford</td>
<td>Ranger</td>
</tr>
<tr>
<td>Sri</td>
<td>NewYork</td>
<td>234</td>
<td>876</td>
<td>Car</td>
<td>Toyota</td>
<td>Camry</td>
</tr>
<tr>
<td>Gabriel</td>
<td>Nashville</td>
<td>345</td>
<td>765</td>
<td>MotorCy</td>
<td>Harley</td>
<td>Hog</td>
</tr>
<tr>
<td>Fred</td>
<td>Nashville</td>
<td>123</td>
<td>654</td>
<td>Car</td>
<td>VW</td>
<td>Bug</td>
</tr>
<tr>
<td>Sri</td>
<td>NewYork</td>
<td>234</td>
<td>654</td>
<td>Car</td>
<td>VW</td>
<td>Bug</td>
</tr>
<tr>
<td>Mary</td>
<td>LosAngeles</td>
<td>456</td>
<td>876</td>
<td>Car</td>
<td>Toyota</td>
<td>Corolla</td>
</tr>
<tr>
<td>Fred</td>
<td>Nshville</td>
<td>123</td>
<td>543</td>
<td>Car</td>
<td>Honda</td>
<td>Accord</td>
</tr>
</tbody>
</table>

Each vehicle associated with one owner (1..1) or at most one (0..1). Nothing about these FDs implies that a Person need be an owner (0..*)

Of the FDs given, which are redundant (and not needed because the can be inferred from the remaining FDs)?

Give the key(s) of the relation above, as dictated by the FDs.
Are the FDs minimal? Can we infer an FD from the other FDs?

\[ S \rightarrow N, \; S \rightarrow A, \; V \rightarrow T, \; V \rightarrow Ma, \; V \rightarrow Mo, \; Mo \rightarrow Ma, \; Mo \rightarrow T, \; V \rightarrow S \]

From \( V \rightarrow Mo \) and \( Mo \rightarrow Ma \) we know \( V \rightarrow Ma \) (\( V \rightarrow Ma \) not needed)
From \( V \rightarrow Mo \) and \( Mo \rightarrow T \) we know \( V \rightarrow T \) (\( V \rightarrow T \) not needed)

A minimal set of FDs (not necessarily unique set):

\[ S \rightarrow N, \; S \rightarrow A, \; V \rightarrow T, \; V \rightarrow Ma, \; V \rightarrow Mo, \; Mo \rightarrow Ma, \; Mo \rightarrow T, \; V \rightarrow S \]

Systematic algorithm for determining minimal set of FDs (and …) uses the \textit{Attribute Closure algorithm} (if you know attribute A, what other attributes, B, can be determined)

Attribute closure of \( N \) is \( \{N\} \) \( N \) not on LHS of any FD
Attribute closure of \( A \) is \( \{A\} \)
Attribute closure of \( S \) is \( \{S\} \rightarrow \{S, N\} \rightarrow \{S, N, A\} \)
Attribute closure of \( T \) is \( \{T\} \)
Attribute closure of \( Ma \) is \( \{Ma\} \rightarrow \{Mo, T\} \rightarrow \{Mo, T, Ma\} \)
Attribute closure of \( Mo \) is \( \{Mo\} \rightarrow \{Mo, T\} \rightarrow \{Mo, T, Ma\} \)

\textbf{Extend the set using any FD with a LHS that is member of current set}
Note that the attribute closure of V includes ALL attributes. **V is a key.**

In general, a key is any minimal set of attributes with attribute closures whose union includes all attributes. Only the attribute closure of V contains V, so V is the only key (though there are many super keys).

If V were not an attribute (and all FDs involving V were removed), then **S, Mo** would be the only key. In general, however, there may be more than one Key.
\[ S \rightarrow N, \quad S \rightarrow A, \quad V \rightarrow T, \quad V \rightarrow Ma, \quad V \rightarrow Mo, \quad Mo \rightarrow Ma, \quad Mo \rightarrow T, \quad V \rightarrow S \]

Attribute closure of \( V \) is
\[ \{ V \} \quad \rightarrow \quad \{ V, Mo \} \quad \rightarrow \quad \{ V, Mo, Ma \} \]
\[ Mo \rightarrow T \quad \rightarrow \quad \{ V, Mo, Ma, T \} \quad \rightarrow \quad \{ V, Mo, Ma, T, S \} \]

\[ V \rightarrow Mo \quad \rightarrow \quad \{ V, Mo, Ma, T, S \} \]

\[ Mo \rightarrow T \quad \rightarrow \quad \{ V, Mo, Ma, T, S \} \]

\[ S \rightarrow N \quad \rightarrow \quad \{ V, Mo, Ma, T, S, N \} \]

\[ S \rightarrow A \quad \rightarrow \quad \{ V, Mo, Ma, T, S, N, A \} \]

**Incomplete** Alg. for determining minimal FD set

Give an ordering of FDs (different orderings may lead to different minimal FD sets)

For each FD, \( A \rightarrow B \),

\[ \text{does attribute closure of } A \text{ include } B \text{ when } A \rightarrow B \text{ is excluded from derivation?} \]

if so, eliminate \( A \rightarrow B \) and continue
S → N,  S → A,  V → T,  V → Ma,  V → Mo,  Mo → Ma,  Mo → T,  V → S

*Incomplete* Alg. for determining minimal FD set

Give an ordering of FDs (different orderings may lead to different minimal FD sets)

For each FD, A → B,

- does attribute closure of A include B when A → B is excluded from derivation?

  if so, eliminate A → B and continue

S → N,  S → A,  V → T,  V → Ma,  V → Mo,  Mo → Ma,  Mo → T,  V → S

Attribute closure of S without S→N is \{S, A\}: have to keep S→N
Attribute closure of S without S→A is \{S, N\}: have to keep S→A
Attribute closure of V without V→T is still \{V, Mo, Ma, T, S, N, A\}: can remove V→T
Attribute closure of V without V→Ma (and without V→T) is still \{V, Mo, Ma, T, S, N, A\}: can remove V→Ma
Confirm that we would have to keep V → Mo,  Mo → Ma,  Mo → T,  V → S
Consider FDs of form $X \rightarrow B$, where $X$ is a set of attributes and $B$ is a single attribute. For example, we might assert $N,A \rightarrow S$ and $V, Mo \rightarrow Ma$ in addition to the other FDs asserted previously (but excluding $V \rightarrow Ma$ and $Mo \rightarrow Ma$).

$$S \rightarrow N, \ S \rightarrow A, \ V \rightarrow T, \ V \rightarrow Ma, \ V \rightarrow Mo, \ Mo \rightarrow Ma, \ Mo \rightarrow T, \ V \rightarrow S, \ N,A \rightarrow S, \ V, Mo \rightarrow Ma$$

**Complete Alg.** For determining minimal FD set

For each FD, $X \rightarrow B$, with non-unit LHS, $X$

if any proper subset of $X$, $Y$, determines remainder of $X$ ($X - Y$), then replace $X$ by $Y$ (replace $X \rightarrow B$ by $Y \rightarrow B$)

$V, Mo \rightarrow Ma$ replaced by $V \rightarrow Ma$

Give an ordering of the FDs (different orderings may result in different minimal FD sets)

For each FD, $Z \rightarrow B$,

does generalized attribute closure of $Z$ include $B$ when $Z \rightarrow B$ is excluded from derivation? (e.g., generalized attribute closure of $\{N,A\}$ is $\{N,A, S\}$)

if so, eliminate $Z \rightarrow B$ from set of FDs and continue ($V \rightarrow T$ eliminated; $V \rightarrow Ma$ not)
S \rightarrow N, \ S \rightarrow A, \ V \rightarrow Mo, \ Mo \rightarrow Ma, \ Mo \rightarrow T, \ V \rightarrow S \ (or \ S \rightarrow N, A ; \ Mo \rightarrow Ma, T ; \ V \rightarrow Mo, S)

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>S</th>
<th>V</th>
<th>T</th>
<th>Ma</th>
<th>Mo</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fred</td>
<td>Nashville</td>
<td>123</td>
<td>987</td>
<td>Truck</td>
<td>Ford</td>
<td>Ranger</td>
</tr>
<tr>
<td>Sri</td>
<td>NewYork</td>
<td>234</td>
<td>876</td>
<td>Car</td>
<td>Toyota</td>
<td>Camry</td>
</tr>
<tr>
<td>Gabriel</td>
<td>Nashville</td>
<td>345</td>
<td>765</td>
<td>MotorCy</td>
<td>Harley</td>
<td>Hog</td>
</tr>
<tr>
<td>Fred</td>
<td>Nashville</td>
<td>123</td>
<td>654</td>
<td>Car</td>
<td>VW</td>
<td>Bug</td>
</tr>
</tbody>
</table>

Show a decomposition of the mega relation into BCNF relations using the FDs above

Give the key(s) of the relation above, as dictated by the FDs.

V is NOT on the right-hand side (RHS) of any FD. Thus, V must be part of any key (i.e., the only way to infer V is to be given V).

\[
\begin{align*}
V \rightarrow Mo & \\
Mo \rightarrow Ma
\end{align*}
\]

Is V alone a key? Yes. Attribute closure of V is \( \{V\} \rightarrow \{V, Mo\} \rightarrow \{V, Mo, Ma\} \)

Are there any other (minimal) keys?

\[
\begin{align*}
Mo \rightarrow T & \\
V \rightarrow S \\
\rightarrow \{V, Mo, Ma, T\} & \\
\rightarrow \{V, Mo, Ma, T, S\}
\end{align*}
\]

\[
\begin{align*}
S \rightarrow N & \\
S \rightarrow A \\
\rightarrow \{V, Mo, Ma, T, S, N\} & \\
\rightarrow \{V, Mo, Ma, T, S, N, A\}
\end{align*}
\]
Importance of lossless decomposition: obvious, don’t want to lose information

Importance of dependency preservation: each FD constraint can be checked by looking within a single table/relation (i.e., efficiency)

(Probably) Preferred decomposition

A dependency preserving, lossless decomposition into BCNF relations

Write CREATE TABLE statements for each of the three relations in the preferred decomposition
CREATE TABLE Person (
    Name VARCHAR(60) NOT NULL,
    Address VARCHAR(120) NOT NULL,
    SSN INTEGER PRIMARY KEY
);

CREATE TABLE Description (
    Model CHAR(20) PRIMARY KEY,
    Manufacturer CHAR(20) NOT NULL,
    Type CHAR(10)
);

CREATE TABLE Vehicle (
    SSN INTEGER,  /* NOT NULL? */
    VRN INTEGER PRIMARY KEY,
    Model CHAR(10) NOT NULL,
    FOREIGN KEY (SSN) REFERENCES Person ON DELETE NO ACTION ON UPDATE CASCADE
    FOREIGN KEY (Model) REFERENCES Description ON DELETE NO ACTION ON UPDATE CASCADE
);
CREATE TABLE Person (
    Name VARCHAR(60) NOT NULL,
    Address VARCHAR(120) NOT NULL,
    SSN INTEGER PRIMARY KEY
);

CREATE TABLE Description (
    Model CHAR(20) PRIMARY KEY,
    Manufacturer CHAR(20) NOT NULL,
    Type CHAR(10)
);

CREATE TABLE Vehicle (
    SSN INTEGER, /* NOT NULL? */
    VRN INTEGER PRIMARY KEY,
    Model CHAR(10) NOT NULL,
    FOREIGN KEY (SSN) REFERENCES Person ON DELETE NO ACTION ON UPDATE CASCADE
    FOREIGN KEY (Model) REFERENCES Description ON DELETE NO ACTION ON UPDATE CASCADE);