A functional dependency is a relationship between two attributes. Typically between the PK and other non-key attributes with in the table.  For any relation R, attribute Y is functionally dependent on attribute X (usually the PK), if for every valid instance of X, that value of X uniquely determines the value of Y.

X ———–>      Y

The left-hand side of the FD is called the determinant, and the right-hand side is the dependent.

Examples:

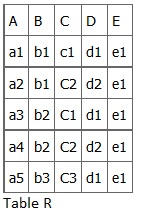
SId   ———->    Name, Address, Birthdate  
SId determines names and address and birthdays.  Given SIN, we can determine any of the other attributes within the table.

Sid, Course  ———>     DateCompleted  
Sin and Course determine date completed.  This must also work for a composite PK.

ISBN  ———–>  Title  
ISBN determines title.

*Rules of Functional Dependencies*

Considser the following instance r(R) of the relation schema R(ABCDE):

[](http://bccampus.pressbooks.com/dbdesign/files/2012/07/Table-R-Functional-Dependency-example.jpg)

What kind of dependencies can

we observe among the attributes in Table R?

* Since the values of A are unique, it follows from the FD definition that:

A →B,    A →C,    A →D,    A →E

* It also follows that A →BC   (or any other subset of ABCDE).
* This can be summarized as   A →BCDE
* From our understanding of primary keys, A is a Primary Key.

Since the values of E are always the same, it follows that:

A →E,    B →E,    C →E,    D →E  
However, we cannot generally summarized above by

ABCD →E  
In general,    A →E,   B →E      AB →E

Other observations:

* combinations of BC are unique, therefore BC →ADE
* combinations of BD are unique, therefore   BD →ACE
* if C values match, so do D values, therefore   C →D  however, D values don’t determine C values, so   C does not determine D, and D does not determine C.

When looking at the data, it makes a lot more sense in terms of which attributes are dependent and which are determinants.

**Closure of a Set of Functional Dependencies**

1. We need to consider *all* functional dependencies that hold. Given a set http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1149.gif of functional dependencies, we can prove that certain other ones also hold. We say these ones are **logically implied** by http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1149.gif.
2. Suppose we are given a relation scheme http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1015.gif, and the set of functional dependencies:

http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tabbing582.gif

Then the functional dependency http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1027.gif is logically implied.

1. The **closure** of a set http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1149.gif of functional dependencies is the set of all functional dependencies logically implied by http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1149.gif.
2. We denote the closure of http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1149.gif by http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1057.gif.
3. To compute http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1057.gif, we can use some rules of inference called **Armstrong's Axioms**:
   * **Reflexivity rule:** if http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1059.gif is a set of attributes and http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1061.gif, then http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1063.gif holds.
   * **Augmentation rule:** if http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1065.gif holds, and http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1067.gif is a set of attributes, then http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1069.gif holds.
   * **Transitivity rule:** if http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1071.gif holds, and http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1073.gif holds, then http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1075.gif holds.
4. These rules are **sound** because they do not generate any incorrect functional dependencies. They are also **complete** as they generate all of http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1057.gif.
5. To make life easier we can use some additional rules, derivable from Armstrong's Axioms:
   * **Union rule:** if http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1079.gif and http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1081.gif, then http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1083.gif holds.
   * **Decomposition rule:** if http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1085.gif holds, then http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1087.gif and http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1089.gif both hold.
     + **Pseudotransitivity rule:** if http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1091.gif holds, and http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1093.gif holds, then http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chap5/_10426_tex2html_wrap1095.gif holds.