ISYS2120 21s2 Week 06 Tut

Relational Design, Functional Dependencies and Schema Normalisation

Dependency Format

- $X \rightarrow Y$
  - X functionally determines Y
  - Y is functionally dependant on X

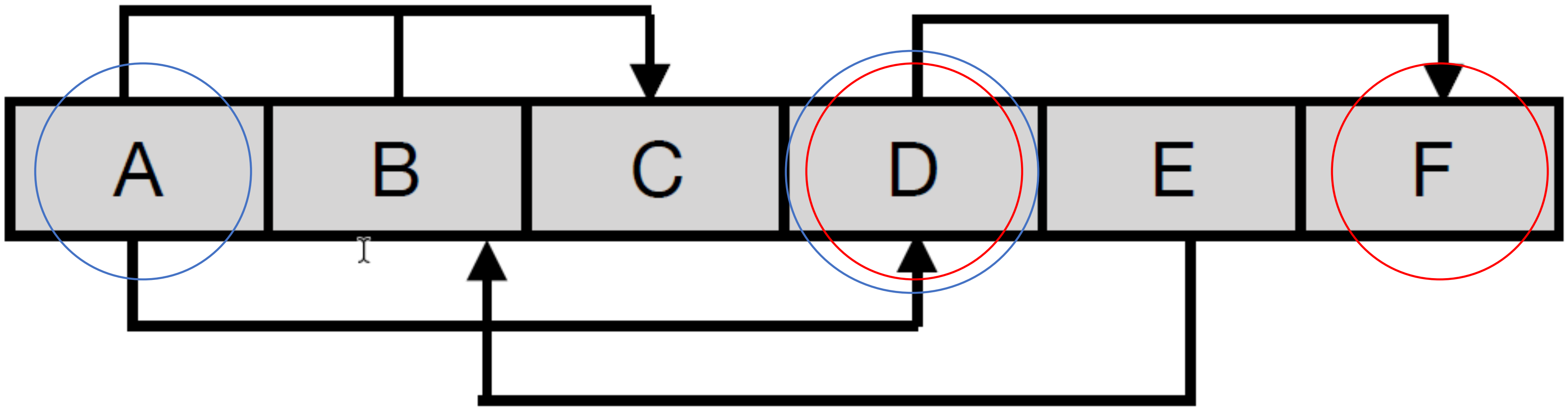
Attribute X is like a candidate key value for Attribute Y.

i.e. Each value of X will map to exactly one value of Y in any given row in a relation.

Transitive Dependency

- If $X \rightarrow Y$ and $Y \rightarrow Z$
- Then $X \rightarrow Z$

Functional Dependency Representation

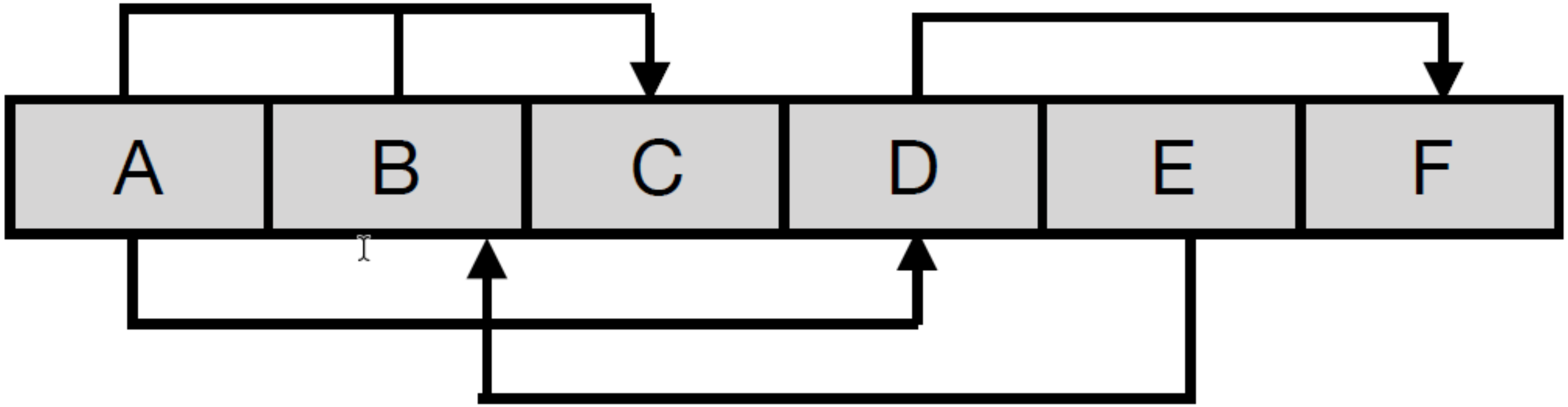


The attribute(s) to where the arrow is pointing is the DEPENDANT ATTRIBUTE(s)

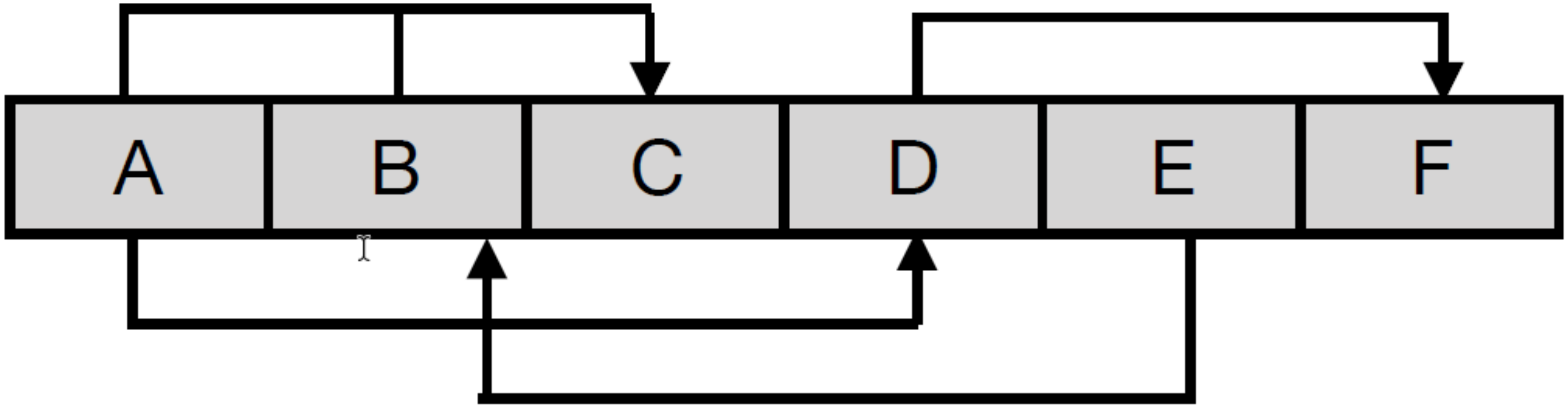
The attribute(s) from which the line originates are the DETERMINING ATTRIBUTE(s)

$D \rightarrow F$

$A \rightarrow D$



- A, B → C
- A → D
- D → F
- E → B



- A, B → C
- A → D
- D → F
- E → B

A → F

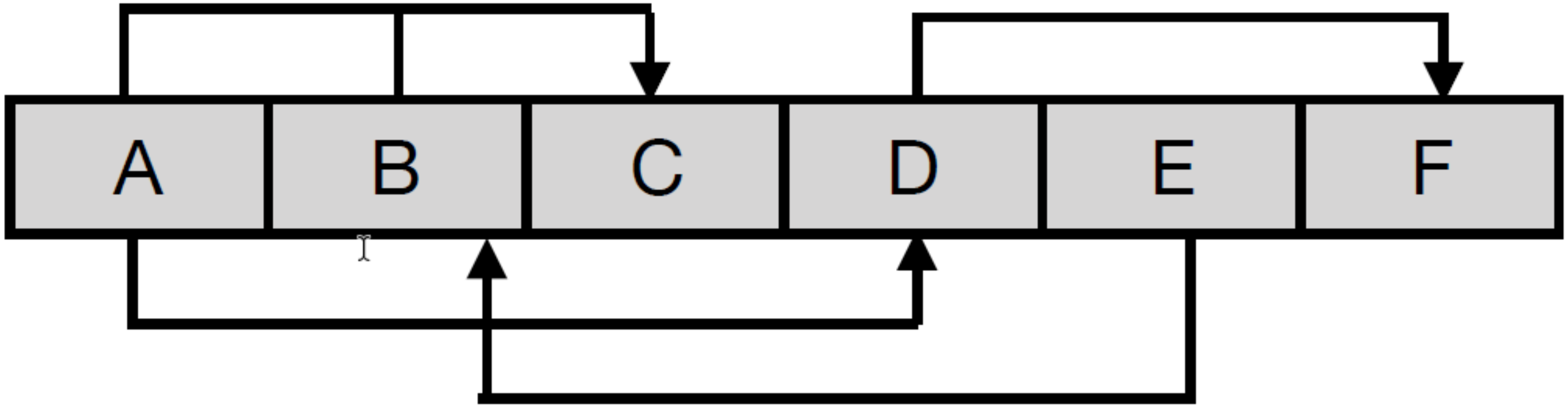
Deduced FD (transitivity rule)

Transitive / Attribute Closure

- Transitive Closure of a set S of attributes is shown as: (S)+
- Transitive Closure is the application of Functional Dependency rules until the resultant set of attributes represented does not change:
- If
  - FD1: X → Y
  - FD2: Y → Z
  - FD3: M → N
- Then, (X)+:
  - Apply FD1 (X → Y), Attributes represented are X and Y
  - (XY)+, Apply FD2 (Y → Z), Attributes represented are X, Y and Z
  - (XYZ)+, Apply FD3 (M → N), Attributes represented are X, Y and Z – unchanged!
- Hence the transitive closure of X given FD1, FD2, FD3 is (XYZ)+

Finding Candidate keys

- A superkey is a set of attributes whose transitive closure contains EVERY attribute in a relation
- A candidate key is a superkey for which no smaller subset is a superkey



(A)+ A → D

(AD)+ D → F

(ADF)+ No more FD

Try adding the next attribute B

(AB)+ A → D

AB → C

(ABCD)+ D → F

(ABCDF)+ No more FD, E is missing

Try adding the missing attribute E

(ABE)+ A → D

AB → C

E → B

(ABCDE)+ D → F

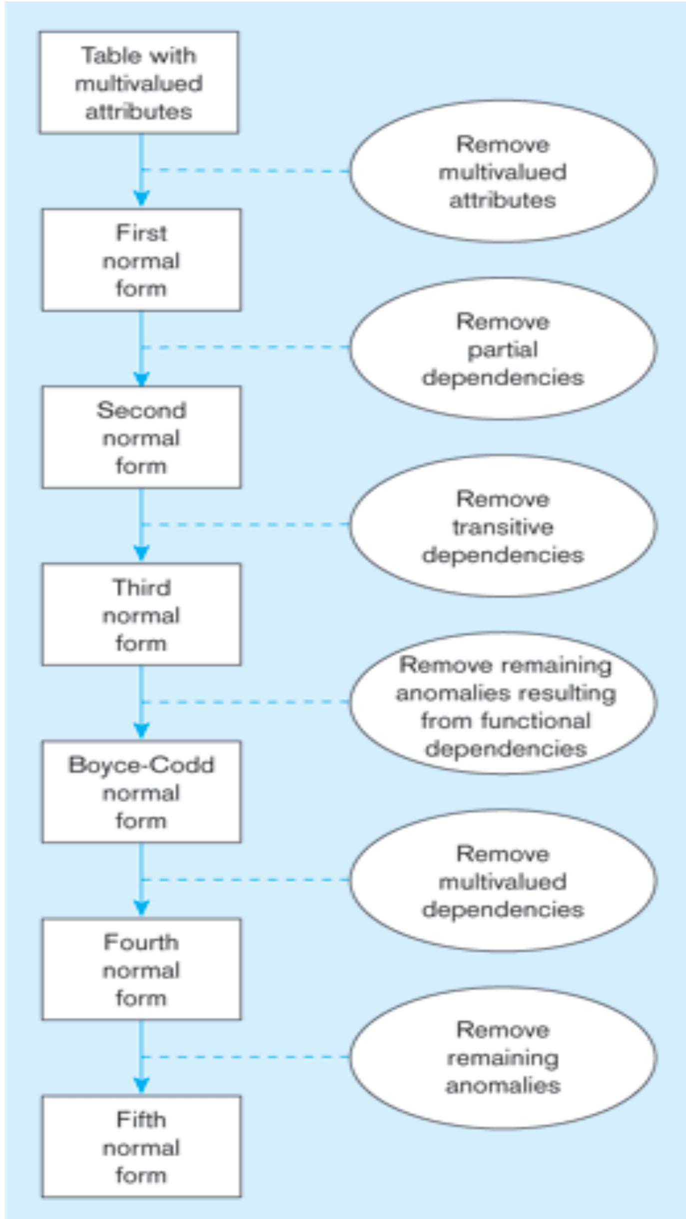
(ADCDEF)+ Complete Relation!

Other Dependencies

- Trivial dependencies:
  - XY also implies XY → X
- Partial dependencies
  - If XY → Z
  - Then Z has a partial dependency on X, and
  - A partial dependency on Y

Normal Forms

- 1NF
  - All attributes are atomic (no multi values attributes)
- 2NF
  - 1NF conditions holds true
  - No Partial Dependencies
- 3NF
  - 1NF, 2NF conditions holds true
  - No Transitive Dependencies
- Boyce-Codd Normal Form (BCNF)
  - In any *non-trivial* $X \rightarrow Y$: *X is a superkey*



A(i)

- $BrokerName \rightarrow Office$

  each broker works in at most one office,

- $StockName \rightarrow Dividend$

  each stock pays at most one dividend,

- $InvestorId \rightarrow BrokerName$

  each investor deals with at most one broker,

- $InvestorID, StockName \rightarrow Quantity$

  an investor can own at most one quantity of a given stock (ie all the holdings of a given investor in a given stock are in one parcel).

A(ii)

Violation of BrokerName → Office

- BrokerName → Office
- StockName → Dividend
- InvestorId → BrokerName
- InvestorID, StockName → Quantity

BrokerName	Office	InvestorId	StockName	Dividend	Quantity
JBWere	Melbourne	4069	BHP	1.25	35
CommSec	Sydney	5035	BHP	1.25	20
JBWere	Sydney	3772	ANZ	0.60	50
ETrade	Canberra	4069	BHP	0.60	15
CommSec	Sydney	5035	RIO	1.25	20
JBWere	Melbourne	3772	ANZ	0.60	100

A(ii)

Violation of StockName → Dividend

- BrokerName → Office
- StockName → Dividend
- InvestorId → BrokerName
- InvestorID, StockName → Quantity

BrokerName	Office	InvestorId	StockName	Dividend	Quantity
JBWere	Melbourne	4069	BHP	1.25	35
CommSec	Sydney	5035	BHP	1.25	20
JBWere	Sydney	3772	ANZ	0.60	50
ETrade	Canberra	4069	BHP	0.60	15
CommSec	Sydney	5035	RIO	1.25	20
JBWere	Melbourne	3772	ANZ	0.60	100

A(ii)

- BrokerName → Office
- StockName → Dividend
- InvestorId → BrokerName
- InvestorID, StockName → Quantity

BrokerName	Office	InvestorId	StockName	Dividend	Quantity
JBWere	Melbourne	4069	BHP	1.25	35
CommSec	Sydney	5035	BHP	1.25	20
JBWere	Sydney	3772	ANZ	0.60	50
ETrade	Canberra	4069	BHP	0.60	15
CommSec	Sydney	5035	RIO	1.25	20
JBWere	Melbourne	3772	ANZ	0.60	100

A(ii)

$\boxed{\text{Violation of } InvestorID, StockName \rightarrow Quantity}$

- $BrokerName \rightarrow Office$
- $StockName \rightarrow Dividend$
- $InvestorId \rightarrow BrokerName$
- $InvestorID, StockName \rightarrow Quantity$

BrokerName	Office	InvestorId	StockName	Dividend	Quantity
JBWere	Melbourne	4069	BHP	1.25	35
CommSec	Sydney	5035	BHP	1.25	20
JBWere	Sydney	3772	ANZ	0.60	50
ETrade	Canberra	4069	BHP	0.60	15
CommSec	Sydney	5035	RIO	1.25	20
JBWere	Melbourne	3772	ANZ	0.60	100

A(iii)

- BrokerName → Office
- StockName → Dividend
- InvestorId → BrokerName
- InvestorID, StockName → Quantity

BrokerName	Office	InvestorId	StockName	Dividend	Quantity
~~JBWere~~ ETrade	~~Melbourne~~ Canberra	4069	BHP	1.25	35
CommSec	Sydney	5035	BHP	1.25	20
JBWere	~~Sydney~~ Melbourne	3772	ANZ	0.60	50
ETrade	Canberra	4069	~~BHP~~ RIO	~~0.60~~ 1.25	15
CommSec	Sydney	5035	RIO	1.25	20
JBWere	Melbourne	3772	~~ANZ~~ RIO	~~0.60~~ 1.25	100

There are many different instances that do meet the FDs!

A(iv)

- BrokerName → Office
- StockName → Dividend
- InvestorId → BrokerName
- InvestorID, StockName → Quantity

- (BrokerName, StockName)+
- Apply FD1: (BrokerName, StockName, Office)+
- Apply FD2: (BrokerName, StockName, Office, Dividend)+
- Can’t apply FD3 – No InvestorID
- Can’t apply FD4 – No InvestorID
- Closure is: (BrokerName, StockName, Office, Dividend)
- (BrokerName, StockName) is not a candidate key, as we can not determine InvestorId or Quantity

A(v)

- BrokerName → Office
- StockName → Dividend
- InvestorId → BrokerName
- InvestorID, StockName → Quantity

- Recall that (BrokerName, StockName)+ is:

  (BrokerName, StockName, Office, Dividend)
- Try (BrokerName, StockName, InvestorID)+
  - Apply FD3, (BrokerName, StockName, Office, Dividend, InvestorID)
  - Apply FD4, (BrokerName, StockName, Office, Dividend, InvestorID, Quantity)
- So Attribute Closure (BrokerName, StockName, InvestorID)+ is a set of all attributes of the whole relation
  - therefore (BrokerName, StockName, InvestorID) is a superkey

Try subsets; we find that (StockName, InvestorID)+ is all attributes,

so (StockName, InvestorID) can be a suitable candidate key

A(vi)

- BrokerName → Office
- StockName → Dividend
- InvestorId → BrokerName
- InvestorID, StockName → Quantity

- Is Investments(BrokerName;Office; InvestorId; StockName;Dividend; Quantity) in BCNF?
  - For InvestorID, StockName → Quantity
    - Left hand side (InvestorID, StockName) is a superkey : this does not contradict BCNF
  - For BrokerName → Office
    - Left hand side (BrokerName) is NOT a superkey: this contradicts BCNF
  - Also for other FDs
  - As long as there is a single FD, where lhs is not a superkey, then we are not in BCNF!

A(vii)

R1(InvestorID, StockName, Dividend, Quantity)

InvestorId	StockName	Dividend	Quantity
4069	BHP	1.25	35
5035	BHP	1.25	20
3772	ANZ	0.60	50
4069	~~BHP~~ RIO	~~0.60~~ 1.25	15
5035	RIO	1.25	20
3772	~~ANZ~~ RIO	~~0.60~~ 1.25	100

Project the contents from A(iii) into just the columns of R1

- ~~BrokerName → Office~~
- StockName → Dividend
- ~~InvestorId → BrokerName~~
- InvestorID, StockName → Quantity

Keep FDs which apply to R1

R2(InvestorID, BrokerName, Office)

BrokerName	Office	InvestorId
ETrade	Canberra	4069
CommSec	Sydney	5035
JBWere	Melbourne	3772

- BrokerName → Office
- ~~StockName → Dividend~~
- InvestorId → BrokerName
- ~~InvestorID, StockName → Quantity~~

A(viii)

- Is decomposition into R1, R2 lossless?
  - Yes; when you join tables back, you get original state
  - Use theorem: Deduce $\text{InvestorID, StockName} \rightarrow \text{Dividend, Quantity}$
    - R1 is the attributes of this deduced FD
    - And R2 is all attributes except rhs $\text{Dividend, Quantity}$
- Are FDs preserved?
  - Yes – R1 has FD2, FD4; R2 has FD1, FD3
- Is it in BCNF?
  - No, in R2 we have $\text{BrokerName} \rightarrow \text{Office}$ but BrokerName is not superkey
  - No, in R1 we have $\text{StockName} \rightarrow \text{Dividend}$ but StockName is not superkey

A(ix)

- $\text{BrokerName} \rightarrow \text{Office}$
- $\text{StockName} \rightarrow \text{Dividend}$
- $\text{InvestorId} \rightarrow \text{BrokerName}$
- $\text{InvestorID, StockName} \rightarrow \text{Quantity}$

- R3(BrokerName,Office)
- R4(Office,InvestorId),
- R5(InvestorId, Stockname),
- R6(Stockname, Dividend), and
- R7(Dividend, Quantity).

1. Is this a lossless decomp?
    1. No – We can not match which investorid's have bought how much of each stock
2. Is this a dependency preserving decomp?
    1. No – FD3 and FD4 are notd educed from what fits into the decomposed relations
3. BCNF does hold (in fact, a relation with two columns is always in BCNF!)
4. Problems with the design – No way to match which investors bought what stock (or how much of it)

A(x)

- R8(BrokerName,Office)
- R9(Stockname, Dividend)
- R10(InvestorId, BrokerName),
- R11(InvestorID, StockName, Quantity)

Initial state of wide table

BrokerName	Office	InvestorId	StockName	Dividend	Quantity
~~JBWere~~ ETrade	~~Melbourne~~ Canberra	4069	BHP	1.25	35
CommSec	Sydney	5035	BHP	1.25	20
JBWere	~~Sydney~~ Melbourne	3772	ANZ	0.60	50
ETrade	Canberra	4069	~~BHP~~ RIO	~~0.60~~ 1.25	15
CommSec	Sydney	5035	RIO	1.25	20
JBWere	Melbourne	3772	~~ANZ~~ RIO	~~0.60~~ 1.25	100

R8(BrokerName,Office)

BrokerName	Office
CommSec	Sydney
ETrade	Canberra
JBWere	Melbourne

R9(Stockname, Dividend)

StockName	Dividend
BHP	1.25
ANZ	0.60
RIO	1.25

R10(InvestorId, BrokerName)

BrokerName	InvestorId
ETrade	4069
CommSec	5035
JBWere	3772

R11(InvestorID, StockName, Quantity)

InvestorId	StockName	Quantity
4069	BHP	35
5035	BHP	20
3772	ANZ	50
4069	~~BHP~~ RIO	15
5035	RIO	20
3772	~~ANZ~~ RIO	100