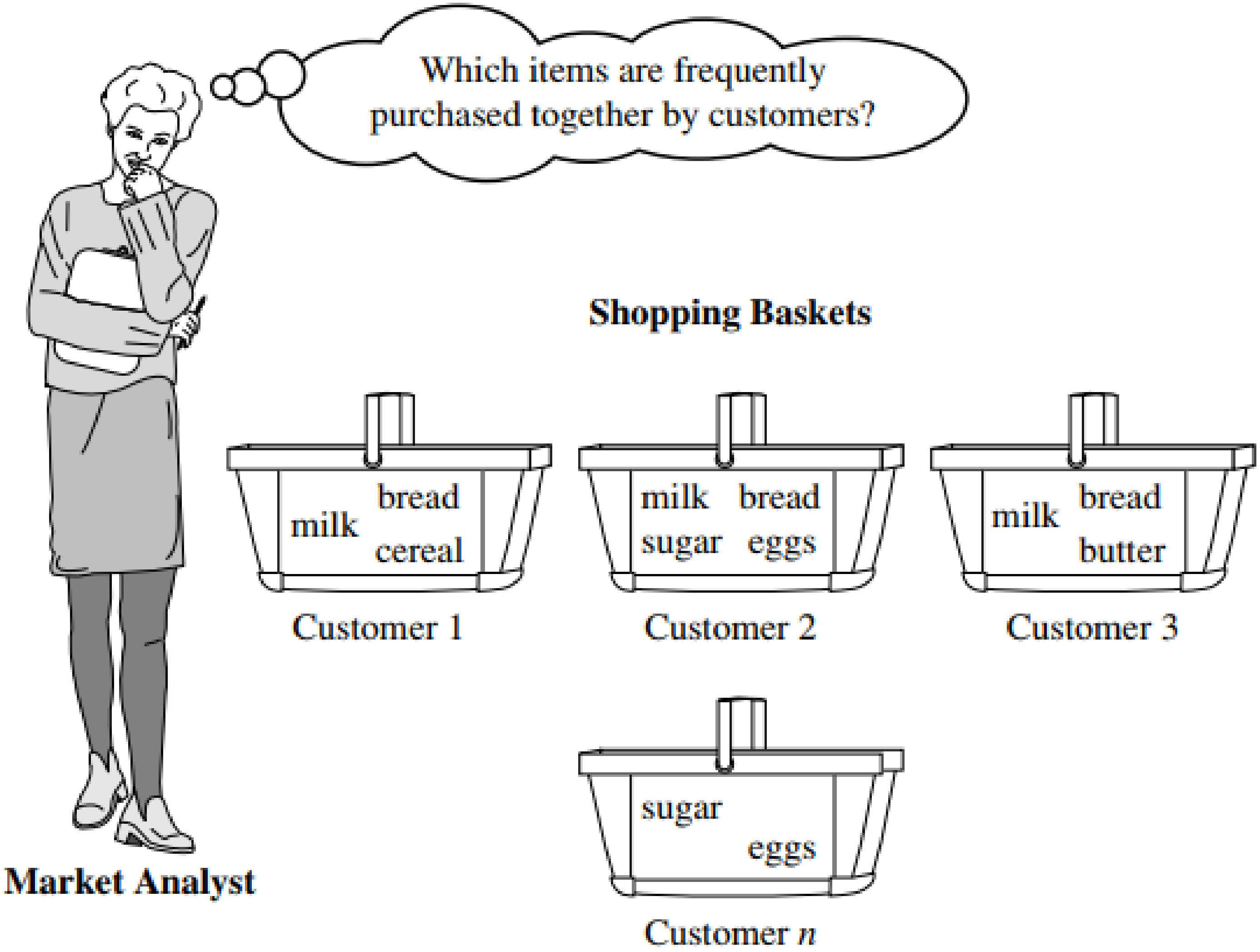
# Chapter 03
# Frequent Patterns Mining



# Market Basket Analysis

# Applications

- **Market Basket Analysis:** given a database of customer transactions, where each transaction is a set of items the goal is to find groups of items which are frequently purchased together.
- **Telecommunication** (each customer is a transaction containing the set of phone calls)
- **Credit Cards/ Banking Services** (each card/account is a transaction containing the set of customer's payments)
- **Medical Treatments** (each patient is represented as a transaction containing the ordered set of diseases)
- **Basketball-Game Analysis** (each game is represented as a transaction containing the ordered set of ball passes)

# Market Basket Analysis

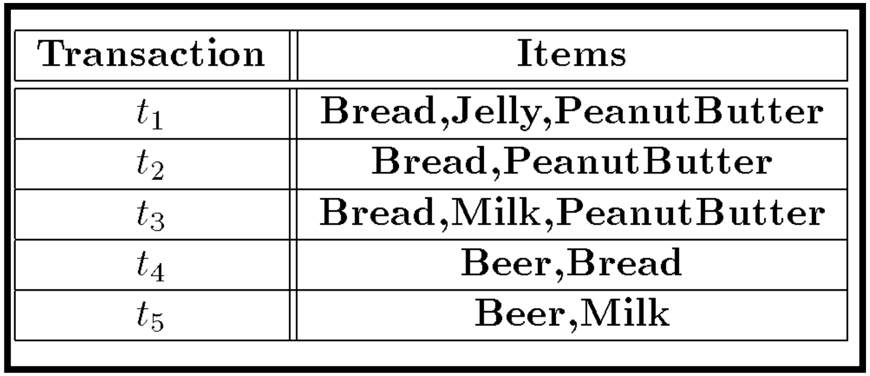
- Market Basket Analysis (Association Analysis) is a mathematical modeling technique based upon the theory that if you buy a certain group of items, you are likely to buy another group of items.

- It is used to analyze the customer purchasing behavior and helps in increasing the sales and maintain inventory by focusing on the point of sale transaction data.

- Given a dataset, the Apriori Algorithm trains and identifies product baskets and product association rules.

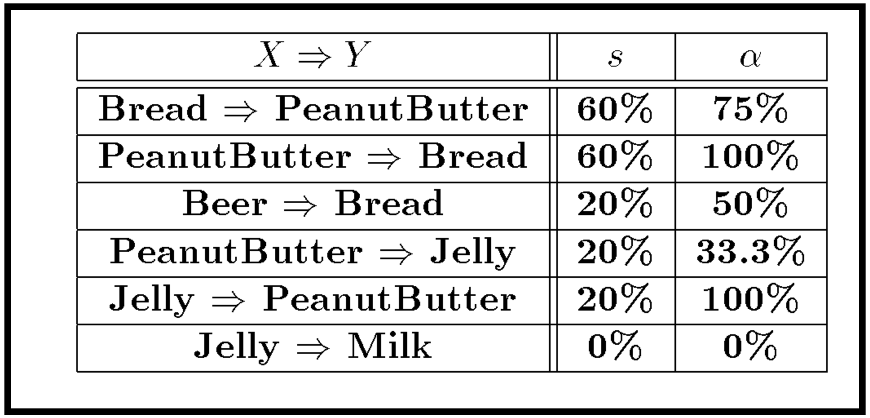
# Association Rule Problem

- Given a database of transactions:



| Transaction | Items |
|---|---|
| $t_1$ | Bread,Jelly,PeanutButter |
| $t_2$ | Bread,PeanutButter |
| $t_3$ | Bread,Milk,PeanutButter |
| $t_4$ | Beer,Bread |
| $t_5$ | Beer,Milk |

- Find all the association rules:



| $X \Rightarrow Y$ | $s$ | $\alpha$ |
|---|---|---|
| Bread $\Rightarrow$ PeanutButter | 60% | 75% |
| PeanutButter $\Rightarrow$ Bread | 60% | 100% |
| Beer $\Rightarrow$ Bread | 20% | 50% |
| PeanutButter $\Rightarrow$ Jelly | 20% | 33.3% |
| Jelly $\Rightarrow$ PeanutButter | 20% | 100% |
| Jelly $\Rightarrow$ Milk | 0% | 0% |

# Association Rule Definitions

- $I=\{i_1, i_2, \ldots, i_n\}$: a set of all the items
- Transaction $T$: a set of items such that $T \subseteq I$
- Transaction Database $D$: a set of transactions
- A transaction $T \subseteq I$ contains a set $X \subseteq I$ of some items, if $X \subseteq T$
- ***An Association Rule***: is an implication of the form $X \Rightarrow Y$, where $X, Y \subseteq I$

# Association Rule Definitions

**Support:**

This measurement technique measures how often multiple items are purchased and compared it to the overall dataset.

**(Item A + Item B) / (Entire dataset)**

**Confidence:**

This measurement technique measures how often item B is purchased when item A is purchased as well.

**(Item A + Item B)/ (Item A)**

# Association Rule Definitions

Frequent pattern:
A pattern (a set of items, subsequences, substructures, etc.) that occurs frequently in a data set

- The **support** s of an itemset X is the percentage of transactions in the transaction database D that contain X.

- The support of the rule $X \Rightarrow Y$ in the transaction database D is the support of the items set $\mathbf{X} \cup \mathbf{Y}$ in D.

- The **confidence** of the rule $X \Rightarrow Y$ in the transaction database D is the ratio of the number of transactions in D that contain $\mathbf{X} \cup \mathbf{Y}$ to the number of transactions that contain **X** in D.

# Association Rule Problem

- Given:
  - — a set $I$ of all the items;
  - — a database $D$ of transactions;
  - — minimum support $s$;
  - — minimum confidence $c$;
- Find:
  - — all association rules $X \Rightarrow Y$ with a minimum support $s$ and confidence $c$.

# Problem Decomposition

| Transaction ID | Items Bought |
|---|---|
| 1 | Shoes, Shirt, Jacket |
| 2 | Shoes,Jacket |
| 3 | Shoes, Jeans |
| 4 | Shirt, Sweatshirt |

If the *minimum support* is 50%, then {Shoes,Jacket} is the only 2-itemset that satisfies the minimum support.

| Frequent Itemset | Support |
|---|---|
| {Shoes} | 75% |
| {Shirt} | 50% |
| {Jacket} | 50% |
| {Shoes, Jacket} | 50% |

If the *minimum confidence* is 50%, then the only two rules generated from this 2-itemset, that have confidence greater than 50%, are:

Shoes ⇒ Jacket Support=50%, Confidence= (2/3)=66%
Jacket ⇒ Shoes Support=50%, Confidence= (2/2)=100%

# The Apriori Algorithm

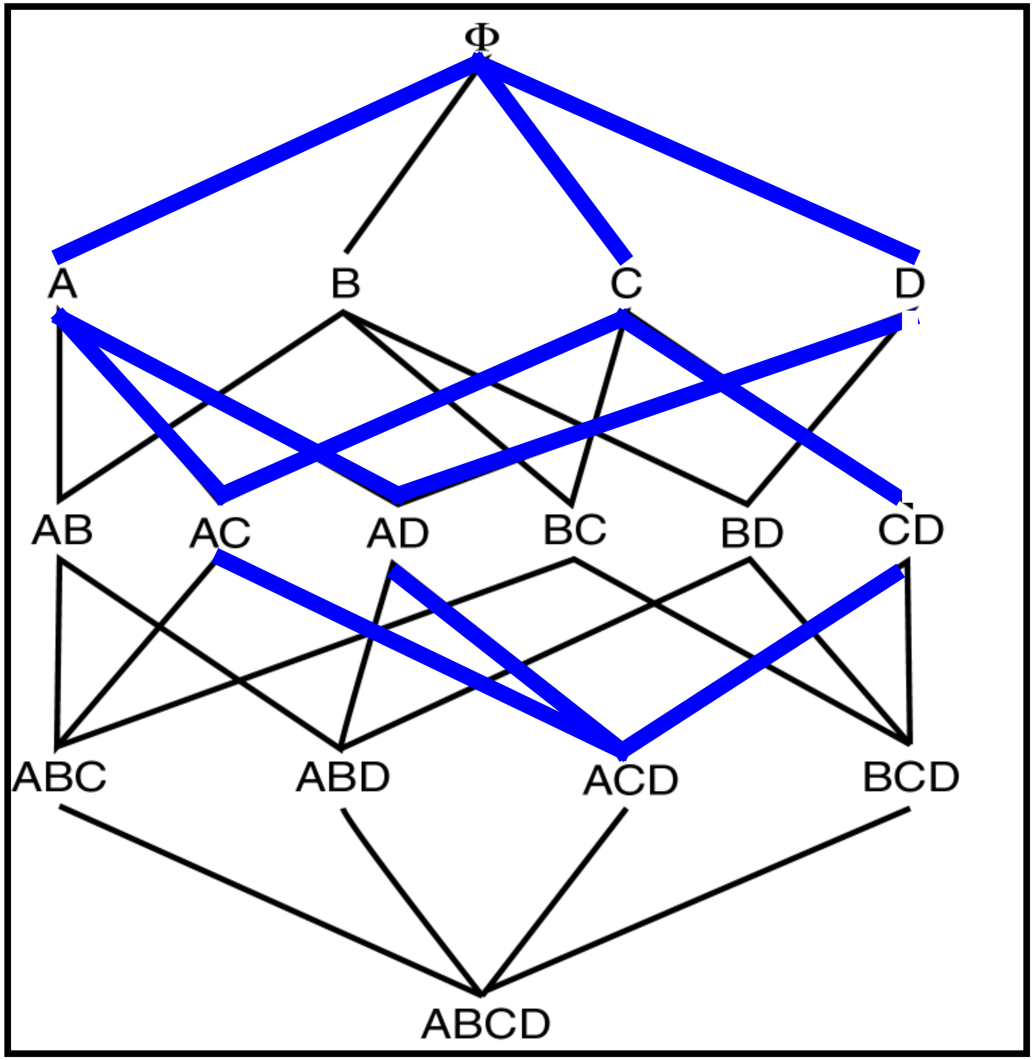
- ***Frequent Itemset Property:***

*Any subset of a frequent itemset is frequent.*

- ***Contrapositive:***

*If an itemset is not frequent, none of its supersets are frequent.*

# Frequent Itemset Property



# The Apriori Algorithm

- $L_k$: Set of frequent itemsets of size $k$ (with min support)
- $C_k$: Set of candidate itemset of size $k$ (potentially frequent itemsets)

```
$L_1$ = {frequent items};
**for** ($k$ = 1; $L_k$ !=$\emptyset$; $k$++) **do**
    $C_{k+1}$ = candidates generated from $L_k$;
    **for each** transaction $t$ in database do
        increment the count of all candidates in
        $C_{k+1}$ that are contained in $t$
    $L_{k+1}$ = candidates in $C_{k+1}$ with min_support
**return** $\cup_k L_k$;
```

# The Apriori Algorithm — Example 1

A dataset D has 4 transactions.

Let the minimum support be 50% and minimum confidence be 80%.

Find all the frequent item set and also generate association rule using Apriori algorithm

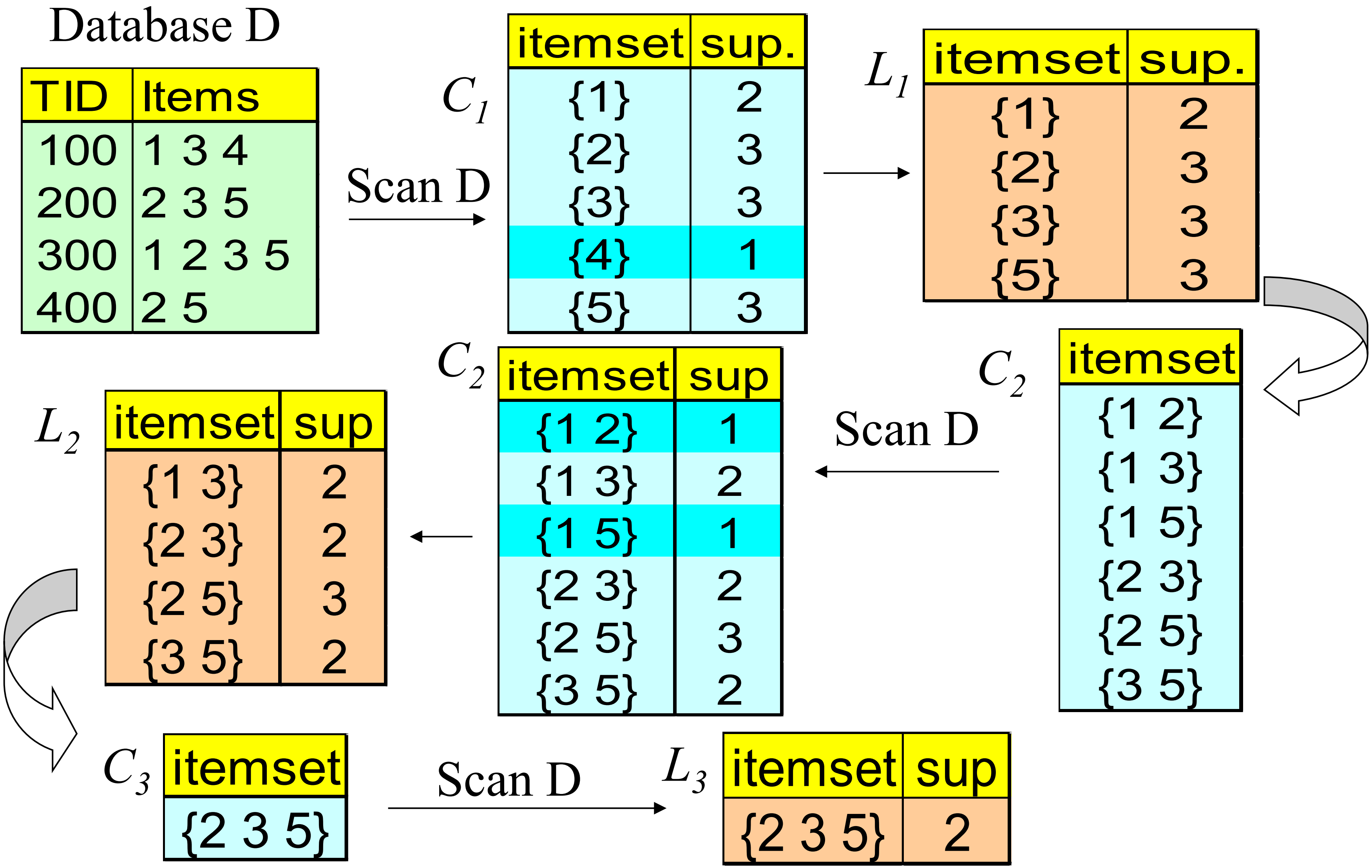
Database D

| TID | Items |
|---|---|
| 100 | 1 3 4 |
| 200 | 2 3 5 |
| 300 | 1 2 3 5 |
| 400 | 2 5 |

Min support count = min support threshold * total no. of transactions
= (50/100) * 4
= 2

# The Apriori Algorithm — Example

Min support =50%



Database D

| TID | Items |
|---|---|
| 100 | 1 3 4 |
| 200 | 2 3 5 |
| 300 | 1 2 3 5 |
| 400 | 2 5 |

Scan D

$C_1$

| itemset | sup. |
|---|---|
| {1} | 2 |
| {2} | 3 |
| {3} | 3 |
| {4} | 1 |
| {5} | 3 |

$L_1$

| itemset | sup. |
|---|---|
| {1} | 2 |
| {2} | 3 |
| {3} | 3 |
| {5} | 3 |

$C_2$

| itemset |
|---|
| {1 2} |
| {1 3} |
| {1 5} |
| {2 3} |
| {2 5} |
| {3 5} |

Scan D

$C_2$

| itemset | sup |
|---|---|
| {1 2} | 1 |
| {1 3} | 2 |
| {1 5} | 1 |
| {2 3} | 2 |
| {2 5} | 3 |
| {3 5} | 2 |

$L_2$

| itemset | sup |
|---|---|
| {1 3} | 2 |
| {2 3} | 2 |
| {2 5} | 3 |
| {3 5} | 2 |

$C_3$

| itemset |
|---|
| {2 3 5} |

Scan D

$L_3$

| itemset | sup |
|---|---|
| {2 3 5} | 2 |

Therefore frequent item sets are

**L= {2,3,5}**

Now, for strong association rule:

Generate non-empty subset for L

**S={2},{3},{5},{2,3},{2,5},{3,5}**

Find S→(L-S)

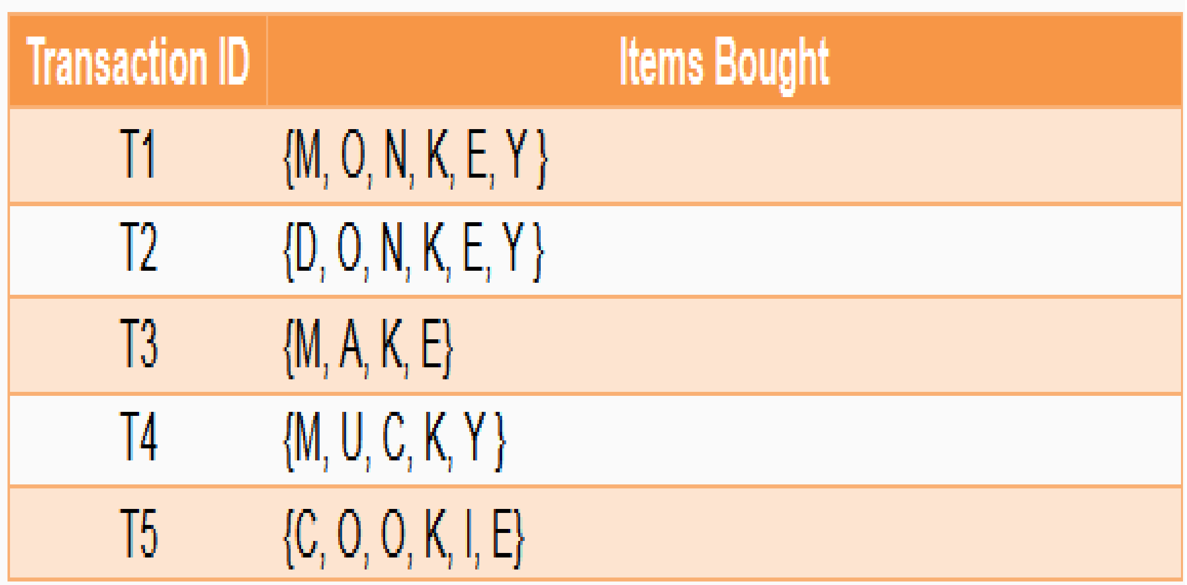
For example: {2}→{3,5}

| S→(L-S) | Support | Confidence | Confidence(%) |
|---|---|---|---|
| {2}→{3,5} | 2 | 2/3 | 66.66 |
| {3}→{2,5} | 2 | 2/3 | 66.66 |
| {5}→{2,3} | 2 | 2/3 | 66.66 |
| {2,3}→{5} | 2 | 2/2 | 100 |
| {2,5}→{3} | 2 | 2/3 | 66.66 |
| {3,5}→{2} | 2 | 2/2 | 100 |

**Minimum confidence threshold=80%**
Therefore strong association rules are_
{2,3}→{5}
{3,5}→{2}

# Apriori Example 2

- Consider the following database with minimum support count=60%.Find all the frequent itemset using apriori algorithm and also generate strong association rules if minimum confidences= 50%.



| Transaction ID | Items Bought |
|---|---|
| T1 | {M, O, N, K, E, Y } |
| T2 | {D, O, N, K, E, Y } |
| T3 | {M, A, K, E} |
| T4 | {M, U, C, K, Y } |
| T5 | {C, O, O, K, I, E} |

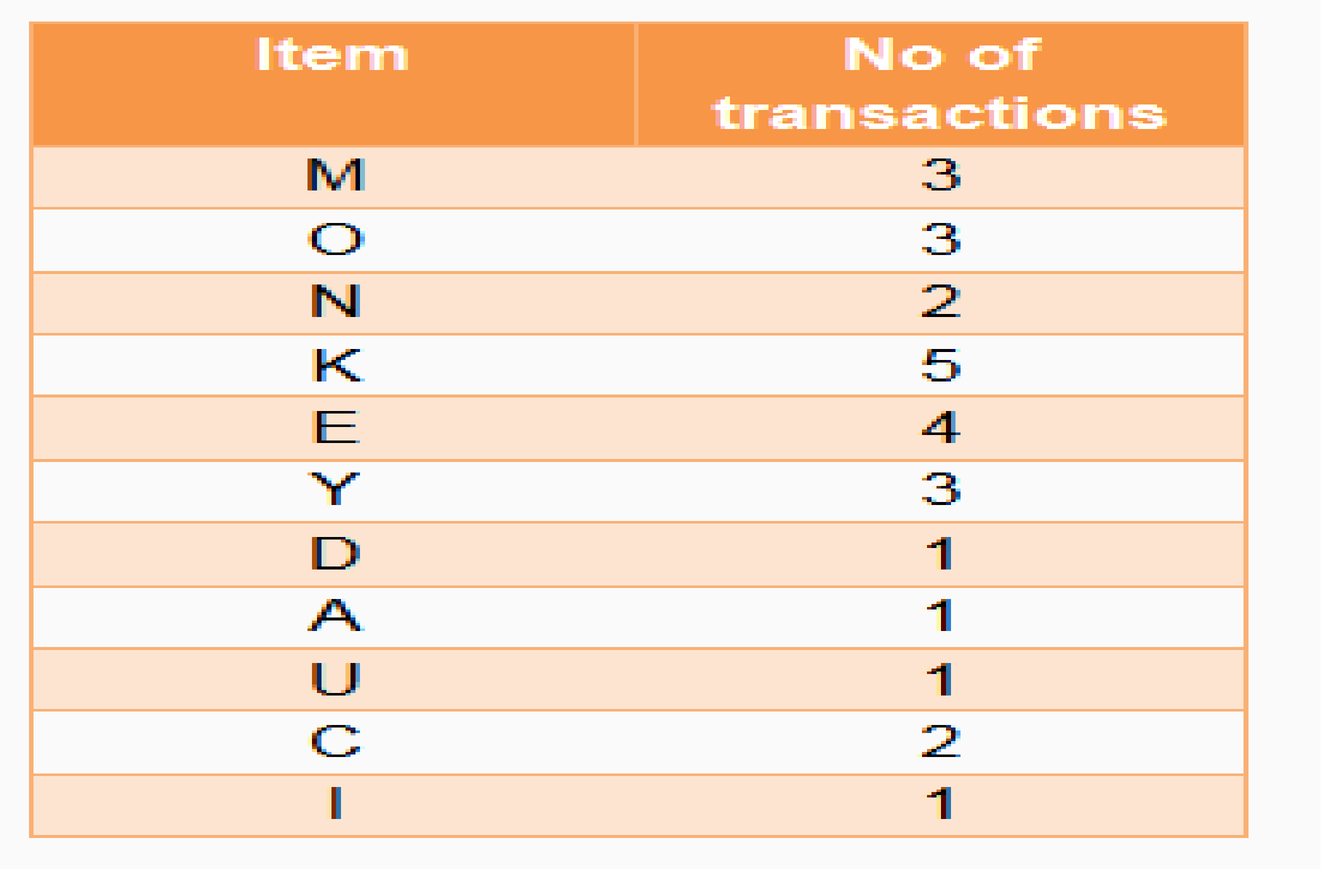
Hint : O is bought 4 times in total, but, it occurs in just 3 transactions.

Min support count = min support threshold * total no. of transactions

Min support count = (60/100) * 5

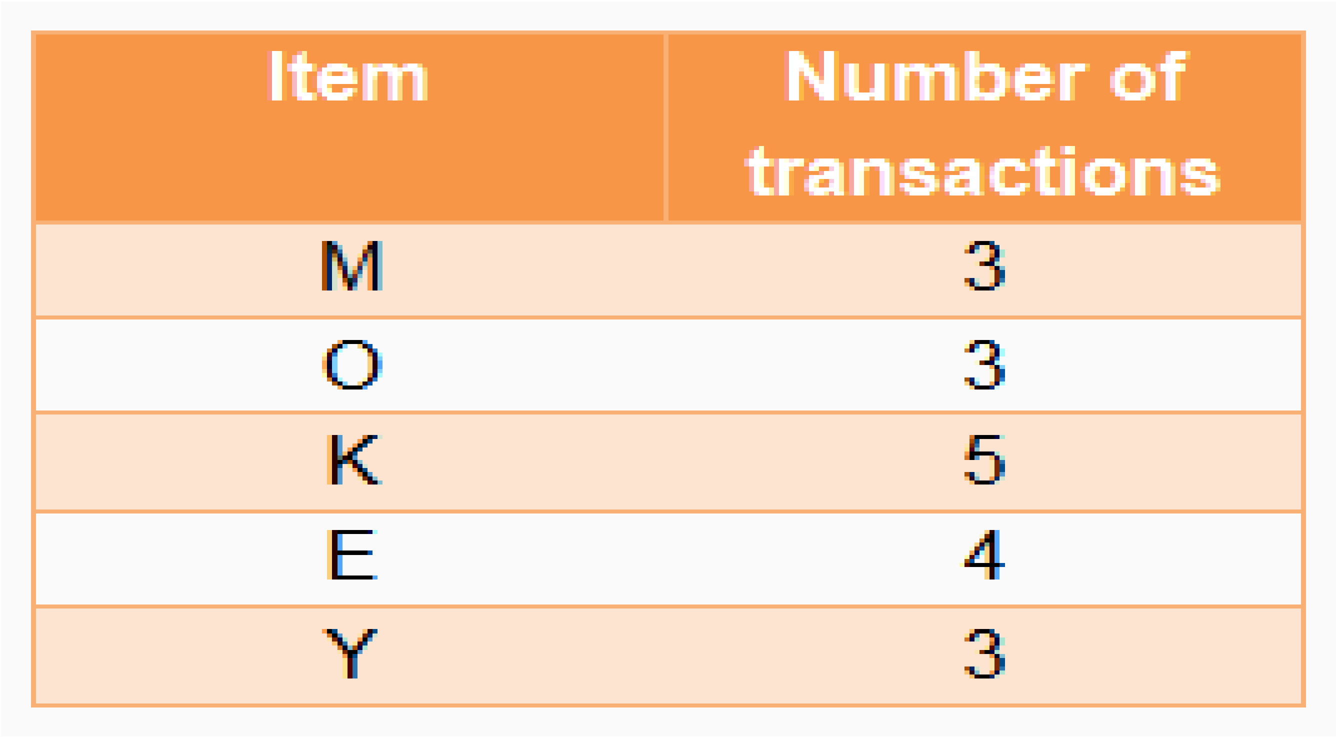
Min support count = 3

# Step 1: Generate C1



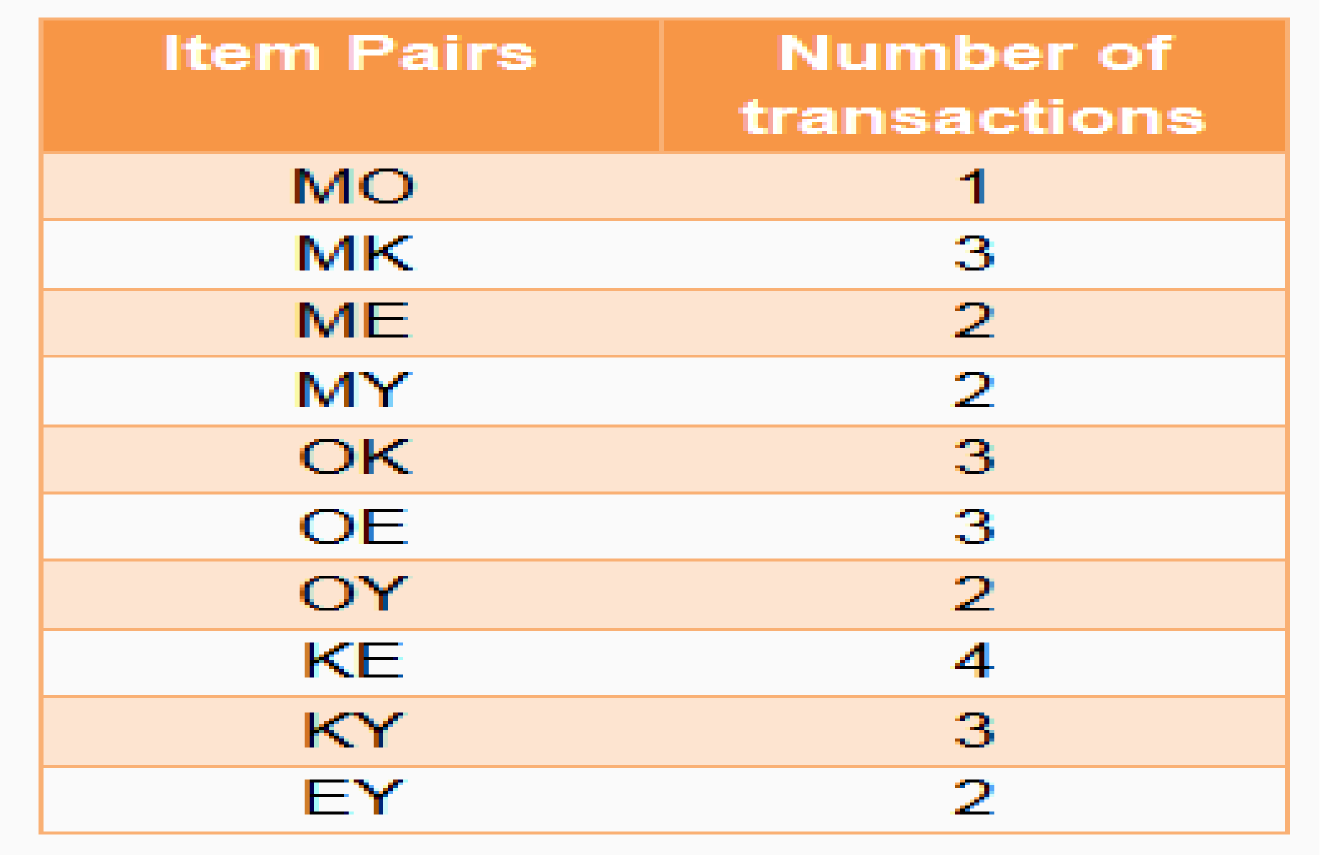
| Item | No of transactions |
|---|---|
| M | 3 |
| O | 3 |
| N | 2 |
| K | 5 |
| E | 4 |
| Y | 3 |
| D | 1 |
| A | 1 |
| U | 1 |
| C | 2 |
| I | 1 |

# Step 2: Generate L1 from C1



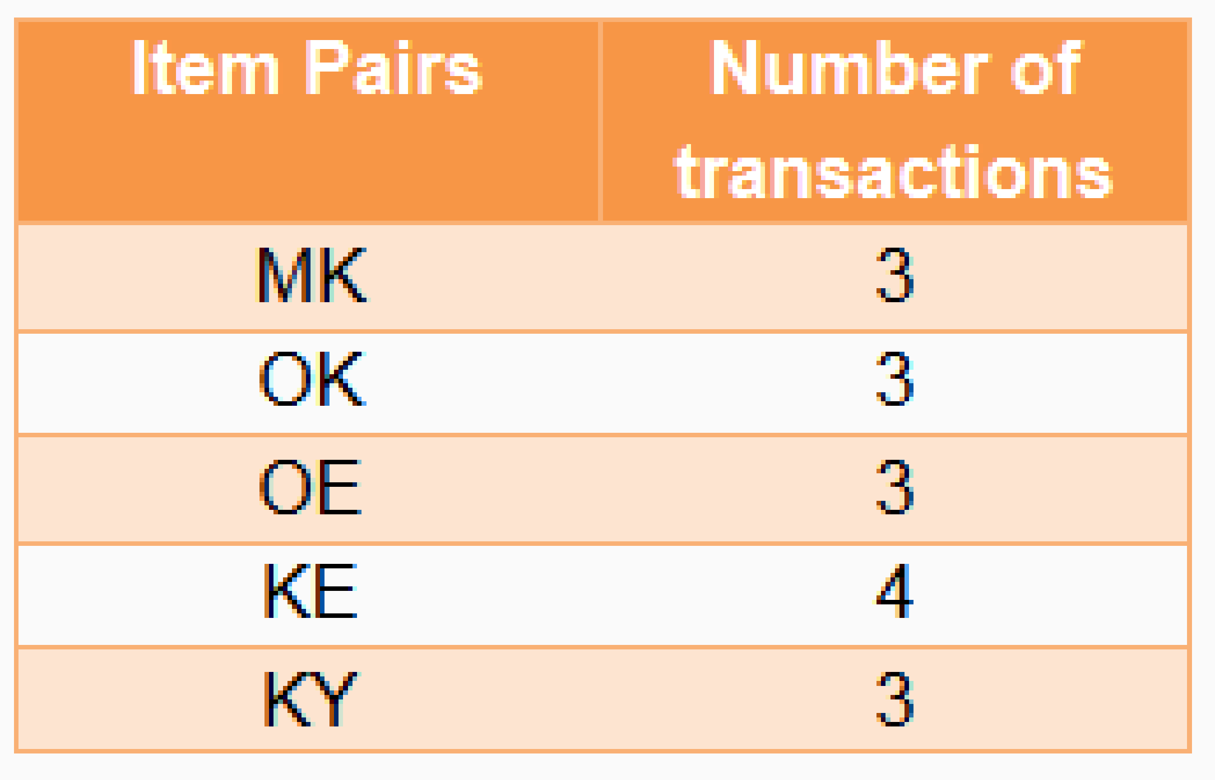
| Item | Number of transactions |
|---|---|
| M | 3 |
| O | 3 |
| K | 5 |
| E | 4 |
| Y | 3 |

# Step 3: Generate C2 from L1 by Join Step



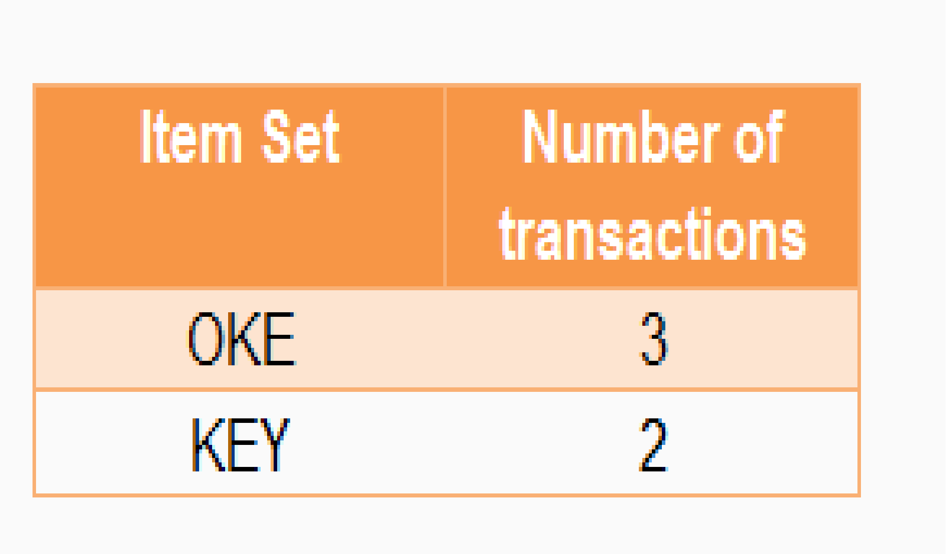
| Item Pairs | Number of transactions |
|---|---|
| MO | 1 |
| MK | 3 |
| ME | 2 |
| MY | 2 |
| OK | 3 |
| OE | 3 |
| OY | 2 |
| KE | 4 |
| KY | 3 |
| EY | 2 |

# Step 4: Generate L2 from C2 by prune Step



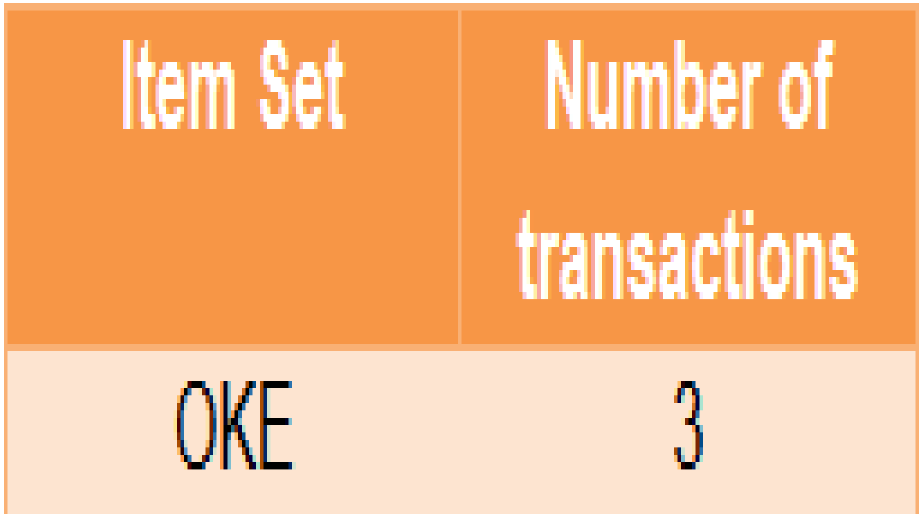
| Item Pairs | Number of transactions |
|---|---|
| MK | 3 |
| OK | 3 |
| OE | 3 |
| KE | 4 |
| KY | 3 |

# Step 5: Generate C3 from L2 by Join step



| Item Set | Number of transactions |
|---|---|
| OKE | 3 |
| KEY | 2 |

# Step 6: Generate L3 from C3 by Prune step



| Item Set | Number of transactions |
|---|---|
| OKE | 3 |

For strong association rule_

L = { O, K, E}

Generate non empty subset of L

S = {O}, {K}, {E}, {OK}, {OE}, {KE}

Generate association rule S→(L-S)

# Step 7: Finding association Rules with min confidences

| Association Rule | Support | Confidence | Confidence(100 %) |
|---|---|---|---|
| O , K → E | **3** | 3/3 | 100% |
| O , E → K | 3 | 3/3 | 100% |
| K , E → O | 3 | 3/4 | 75% |
| O → K , E | 3 | 3/3 | 100% |
| K → O , E | 3 | 3/5 | 60% |
| E → O , K | 3 | 3/4 | 75% |

All the association rules are having confidence more than 50%.

Therefore all rules are strong association rule

# Apriori Example 3

- Consider the following database with minimum support count=50%.Find all the frequent itemset using apriori algorithm and also generate strong association rules if minimum confidences= 50%.

| T id | Items Bought |
|---|---|
| 1 | A,B,D |
| 2 | A,D |
| 3 | A,C |
| 4 | B,D,E,F |

# Apriori Example 4

| Tid | items |
|---|---|
| 1 | A B D |
| 2 | B C D |
| 3 | A B |
| 4 | B D |
| 5 | A B C |

**Find frequent item set and strong association rule**

- min support = 30%
- min confidence = 75%

# How to Generate Candidates

```
Input: $L_{i-1}$ : set of frequent itemsets of size $i-1$
Output: $C_i$ : set of candidate itemsets of size $i$
$C_i$ = empty set;
for each itemset $J$ in $L_{i-1}$ do
          for each itemset $K$ in $L_{i-1}$ s.t. $K <> J$ do
                    if $i-2$ of the elements in $J$ and $K$ are equal then
                              if all subsets of $\{K \cup J\}$ are in $L_{i-1}$ then
                                        $C_i = C_i \cup \{K \cup J\}$
return $C_i$;
```

# Example of Generating Candidates

- $L_3=\{abc, abd, acd, ace, bcd\}$
- Generating $C_4$ from $L_3$
  - *abcd* from *abc* and *abd*
  - *acde* from *acd* and *ace*
- Pruning:
  - *acde* is removed because *ade* is not in $L_3$
- $C_4=\{abcd\}$

# Example of Discovering Rules

Let use consider the 3-itemset {I1, I2, I5}:

$I1 \wedge I2 \Rightarrow I5$

$I1 \wedge I5 \Rightarrow I2$

$I2 \wedge I5 \Rightarrow I1$

$I1 \Rightarrow I2 \wedge I5$

$I2 \Rightarrow I1 \wedge I5$

$I5 \Rightarrow I1 \wedge I2$

# Discovering Rules

```
for each frequent itemset $I$ do
  for each subset $C$ of $I$ do
    if (support($I$) / support($I - C$) >= minconf) then
      output the rule $(I - C) \Rightarrow C$,
      with confidence = support($I$) / support ($I - C$)
        and support = support($I$)
```

# Example of Discovering Rules

| **TID** | **List of Item_IDs** |
|---|---|
| T100 | I1, I2, I5 |
| T200 | I2, I4 |
| T300 | I2, I3 |
| T400 | I1, I2, I4 |
| T500 | I1, I3 |
| T600 | I2, I3 |
| T700 | I1, I3 |
| T800 | I1, I2, I3, I5 |
| T900 | I1, I2, I3 |

Let use consider the 3-itemset {I1, I2, I5} with support of 0.22(2)%. Let generate all the association rules from this itemset:

$I1 \wedge I2 \Rightarrow I5$ *confidence*= 2/4 = 50%

$I1 \wedge I5 \Rightarrow I2$ *confidence*= 2/2 = 100%

$I2 \wedge I5 \Rightarrow I1$ *confidence*= 2/2 = 100%

$I1 \Rightarrow I2 \wedge I5$ *confidence*= 2/6 = 33%

$I2 \Rightarrow I1 \wedge I5$ *confidence*= 2/7 = 29%

$I5 \Rightarrow I1 \wedge I2$ *confidence*= 2/2 = 100%

Frequent Itemset with support count 2 is {I1,I2,I3} and {I1,I2,I5}

| Association rule | support | confidence | Confidence % |
|---|---|---|---|
| 1,5→2 | 2 | 2/2 | 100 |
| 2,5→1 | 2 | 2/2 | 100 |
| 5→1,2 | 2 | 2/2 | 100 |
| | | | |

# Apriori Advantages/Disadvantages

- ***Advantages:***
  - Uses large itemset property.
  - Easily parallelized
  - Easy to implement.
- ***Disadvantages:***
  - Assumes transaction database is memory resident.
  - Requires many database scans.

# What is FP Growth?

- FP Growth Stands for frequent pattern growth
- It is a scalable technique for mining frequent pattern in a database

# FP Growth

- FP growth improves Apriority to a big extent
- Frequent Item set Mining is possible without candidate generation
- Only “two scan” to the database is needed

BUT HOW?

# FP Growth

- Simply a two step procedure
  - Step 1: Build a compact data structure called the FP-tree
    - Built using 2 passes over the data-set.
  - Step 2: Extracts frequent item sets directly from the FP-tree

Note: Assume min support = 2

| **TID** | **List of Item_IDs** |
|---|---|
| T100 | I1, I2, I5 |
| T200 | I2, I4 |
| T300 | I2, I3 |
| T400 | I1, I2, I4 |
| T500 | I1, I3 |
| T600 | I2, I3 |
| T700 | I1, I3 |
| T800 | I1, I2, I3, I5 |
| T900 | I1, I2, I3 |

# FP Growth

- Now Lets Consider the following transaction table

| **TID** | **List of Item_IDs** |
|---|---|
| T100 | I1, I2, I5 |
| T200 | I2, I4 |
| T300 | I2, I3 |
| T400 | I1, I2, I4 |
| T500 | I1, I3 |
| T600 | I2, I3 |
| T700 | I1, I3 |
| T800 | I1, I2, I3, I5 |
| T900 | I1, I2, I3 |

# FP Growth

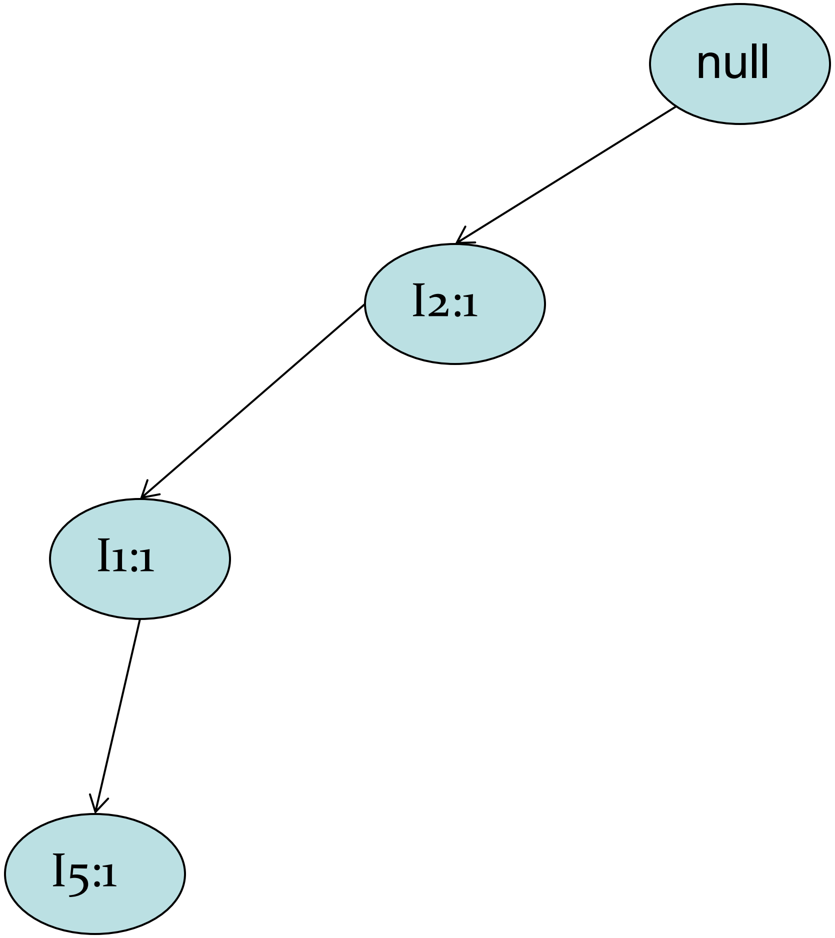
- Now we will build a FP tree of that database
- Item sets are considered in order of their descending value of support count.

# FP Growth

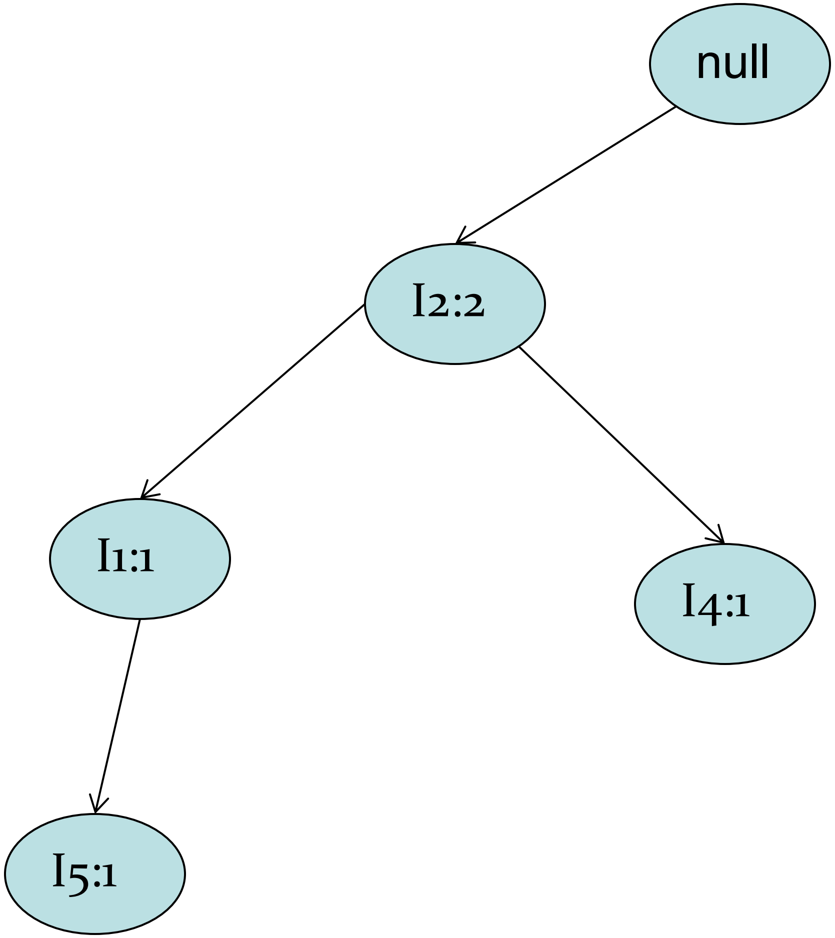
| Items | Support count |
|---|---|
| I1 | 6 |
| I2 | 7 |
| I3 | 6 |
| I4 | 2 |
| I5 | 2 |

| TID | List of Item_IDs |
|---|---|
| T100 | I2, I1, I5 |
| T200 | I2, I4 |
| T300 | I2, I3 |
| T400 | I2, I1, I4 |
| T500 | I1, I3 |
| T600 | I2, I3 |
| T700 | I1, I3 |
| T800 | I2, I1, I3, I5 |
| T900 | I2, I1, I3 |

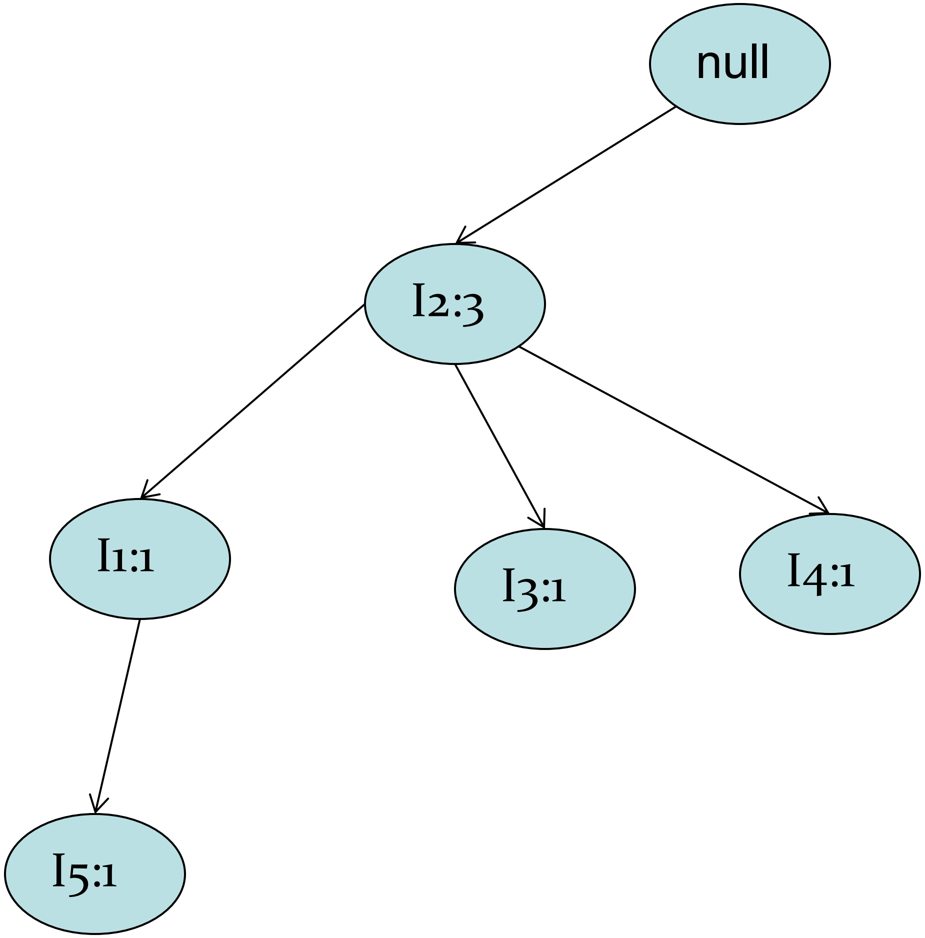
## For Transaction:
## I2,I1,I5



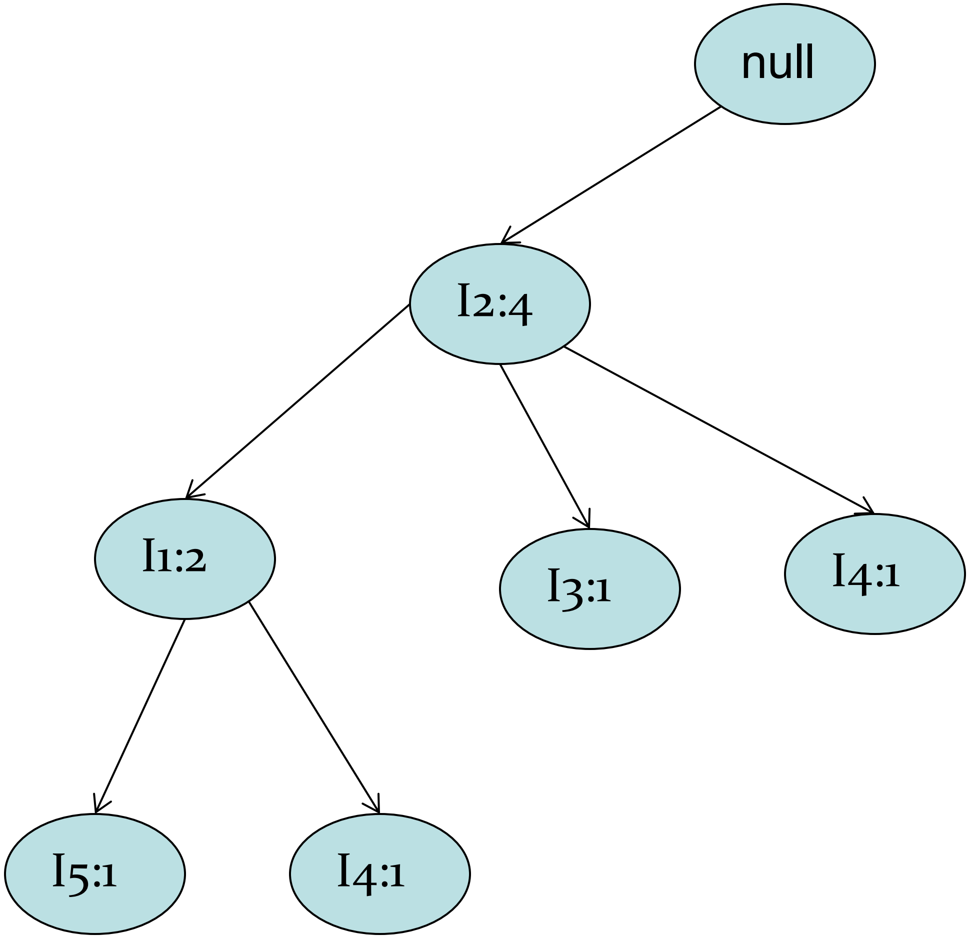
## For Transaction:
## I2,I4



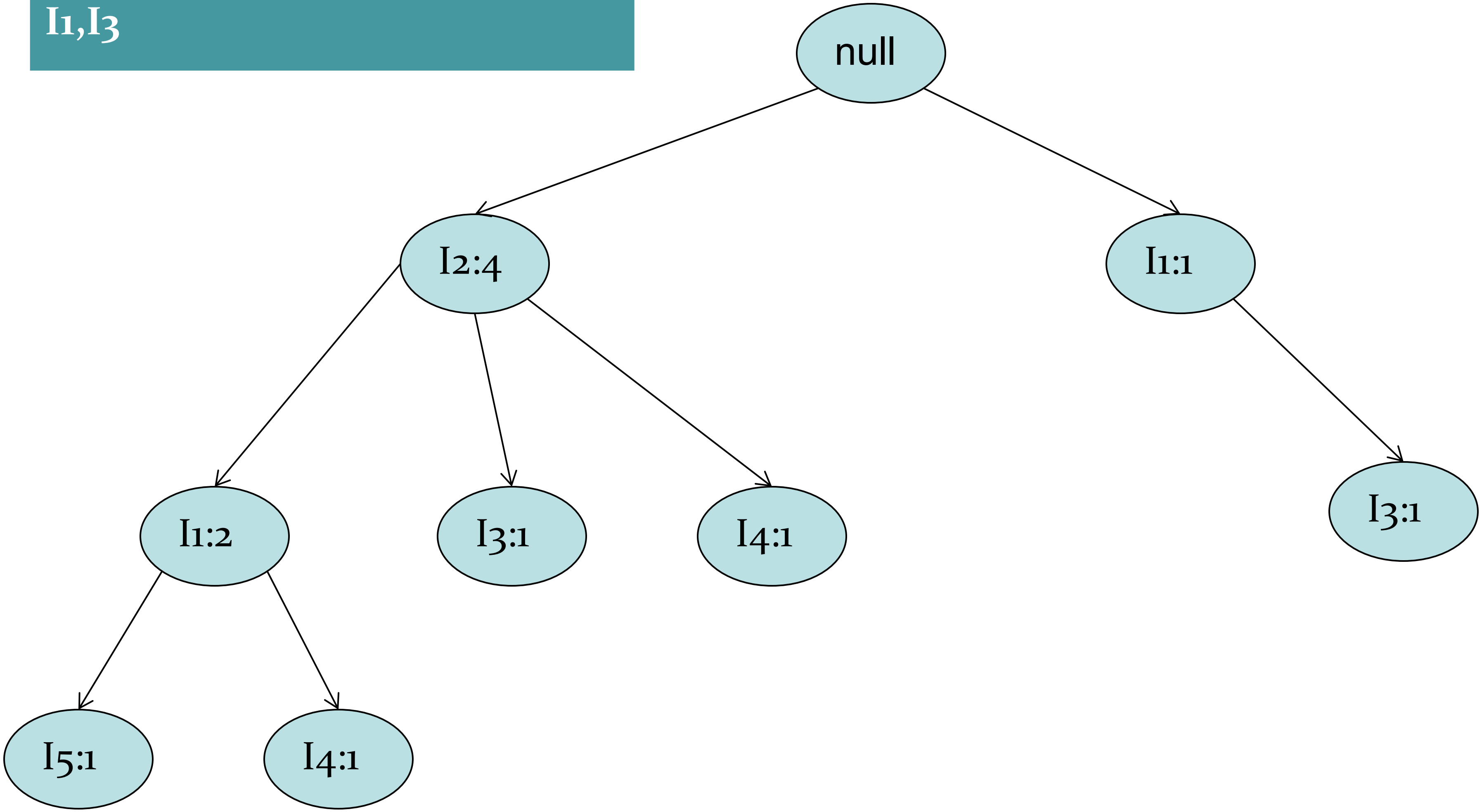
## For Transaction:
## I2,I3



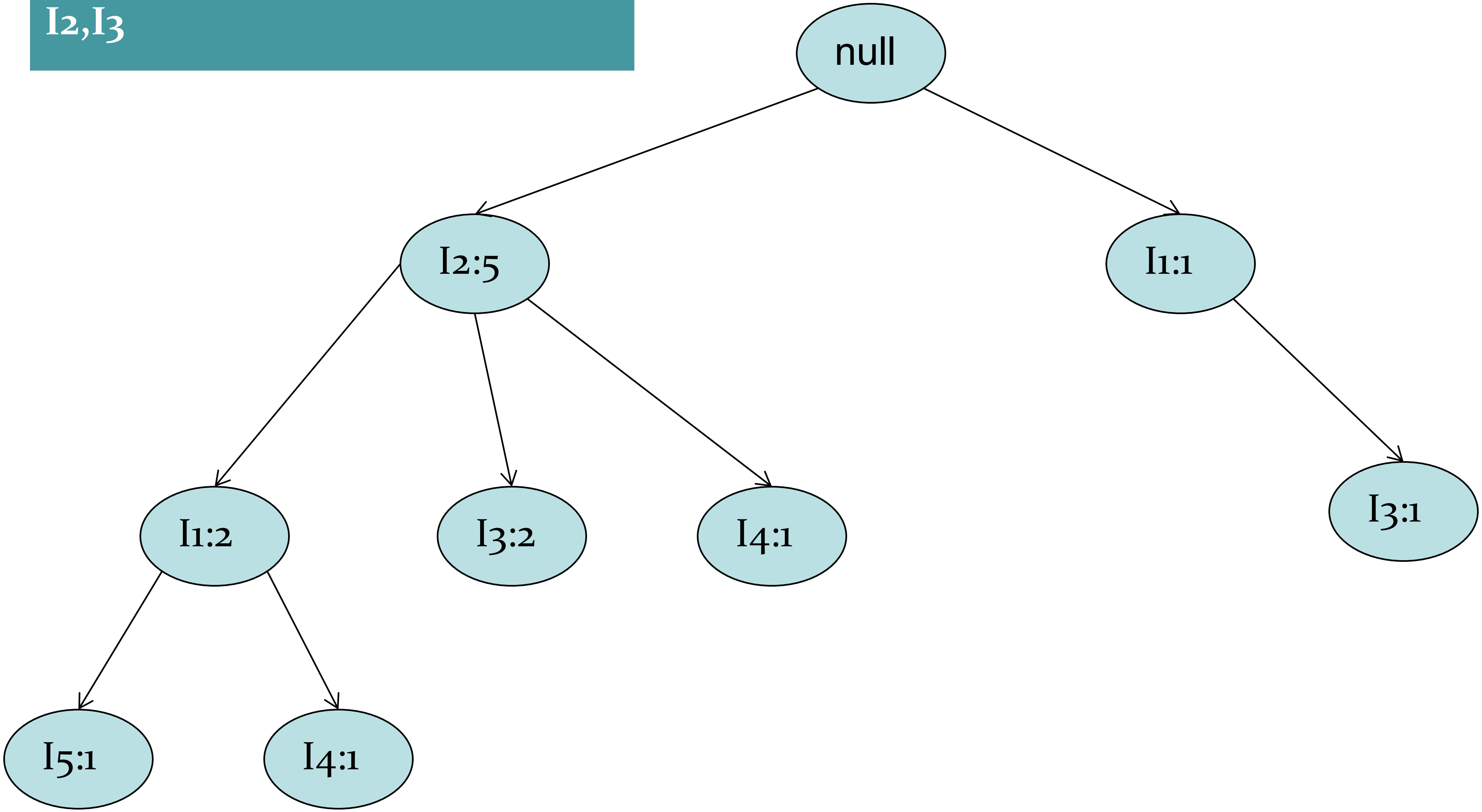
## For Transaction:
## I2,I1,I4



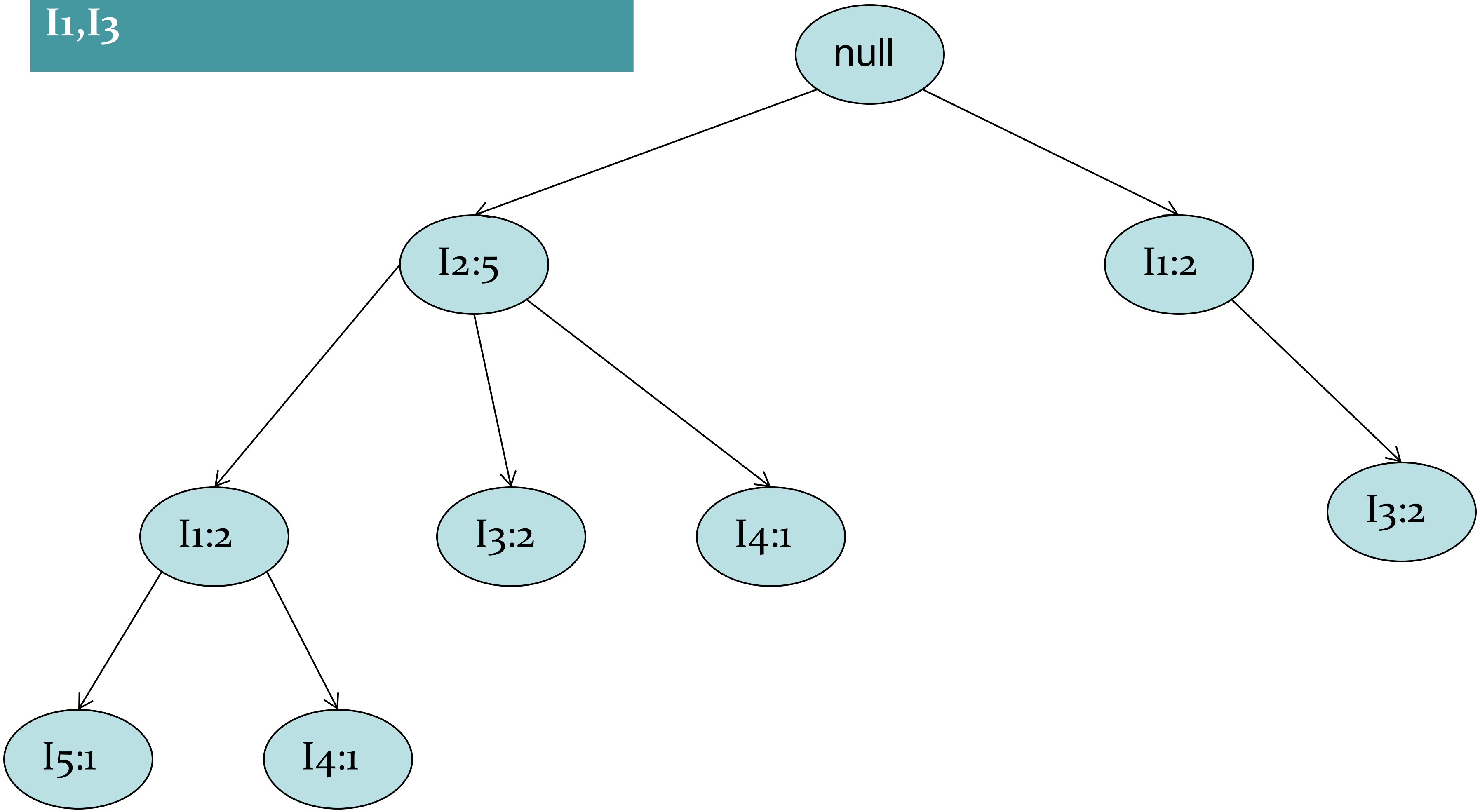
## For Transaction:
## I1,I3



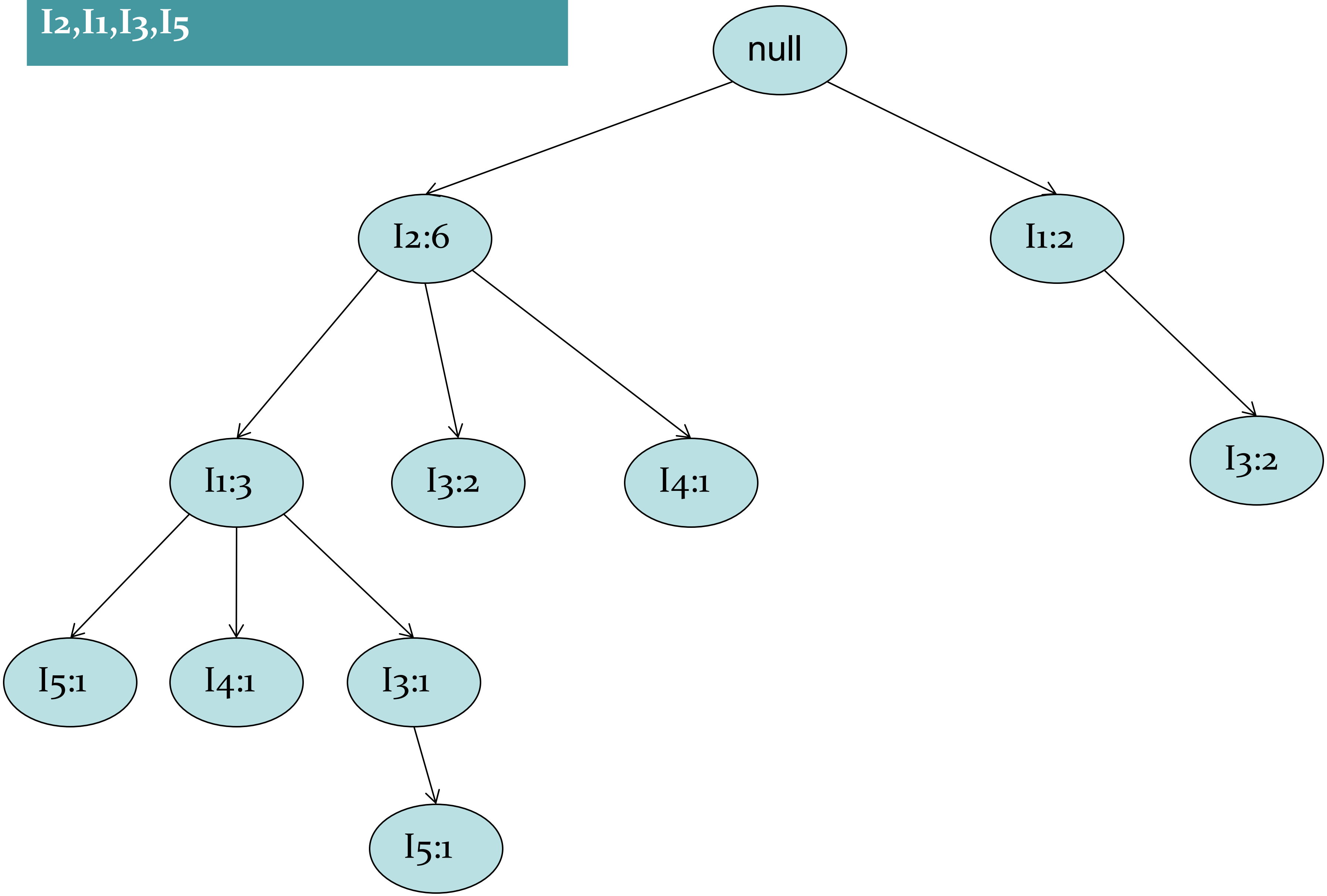
For Transaction:
I2,I3



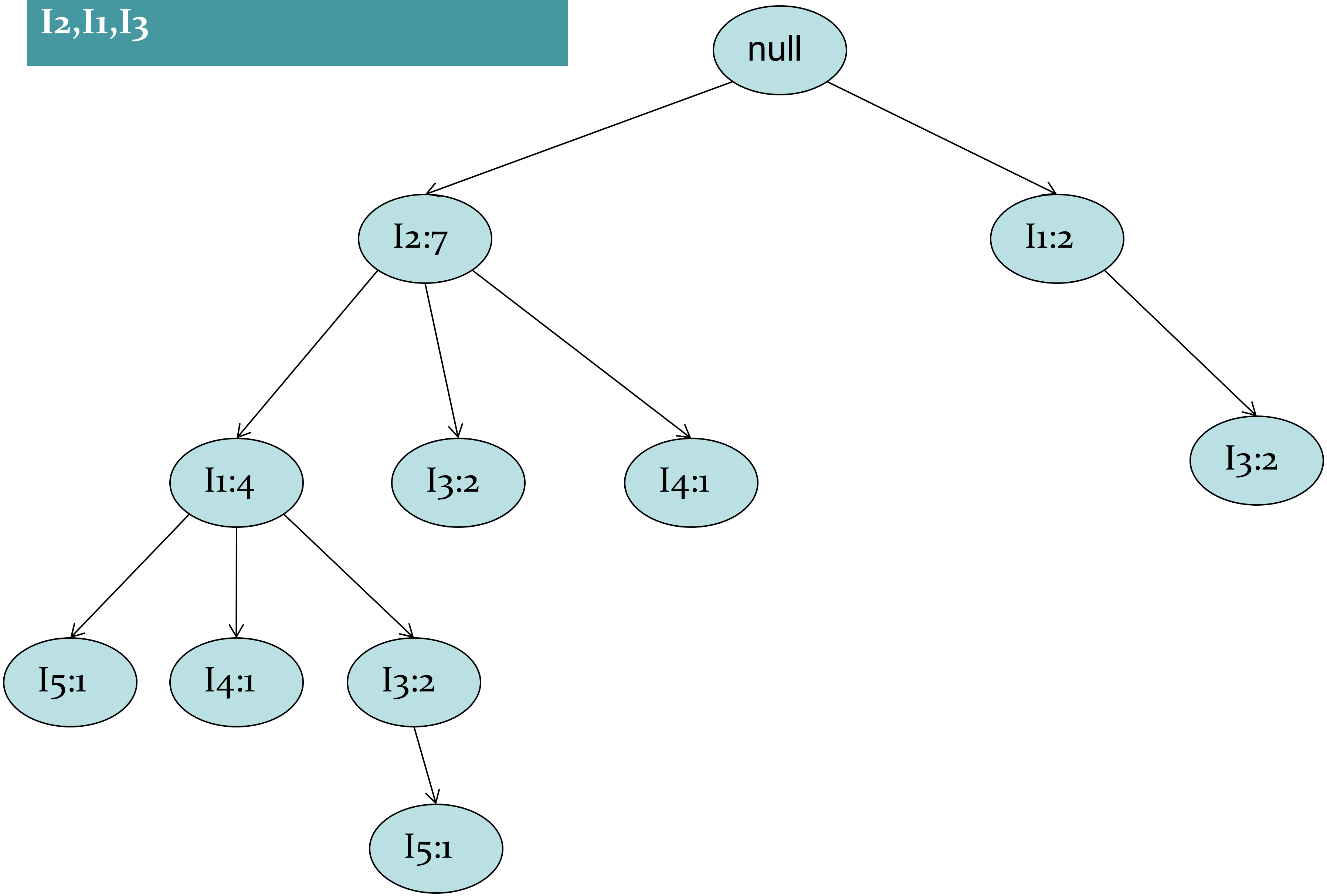
**For Transaction:**
I1,I3



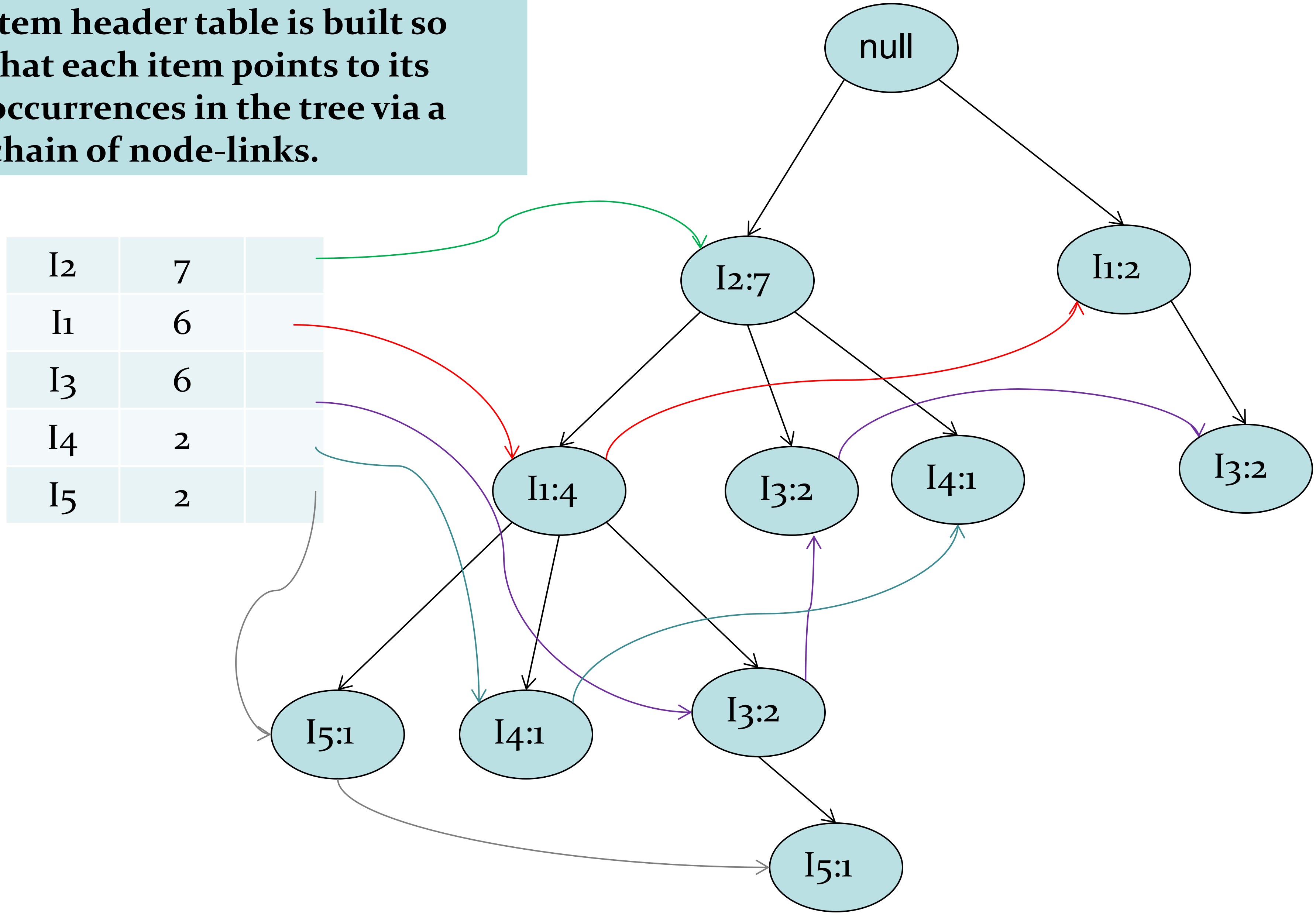
For Transaction:
I2,I1,I3,I5



For Transaction:
I2,I1,I3



To facilitate tree traversal, an item header table is built so that each item points to its occurrences in the tree via a chain of node-links.



| | | |
|---|---|---|
| I2 | 7 | |
| I1 | 6 | |
| I3 | 6 | |
| I4 | 2 | |
| I5 | 2 | |

# FP Growth

- FP Tree Construction Over!!
  Now we need to find conditional pattern base and Conditional FP Tree for each item

# Frequent Patters Generated

| Item | Conditional Pattern Base | Conditional FP-tree | Frequent Pattern Generated |
|---|---|---|---|
| I5 | {I2,I1 : 1},{I2,I1,I3 : 1} | {I2:2,I1:2} | {I2, I5: 2}, {I1, I5: 2},<br>{I2, I1, I5: 2} |
| I4 | {I2,I1:1},{I2:1} | {I2:2} | {I2, I4: 2} |
| I3 | {I2,I1:2},{I2:2},{I1:2} | {I2:4},{I1:2} | {I2, I3: 4}, {I1, I3: 2}, {I2, I1, I3: 2} |
| I1 | {I2:4} | {I2:4} | {I2, I1: 4} |
| I2 | | Ignore as no Branch | |

## Example 2: FP Growth

- Draw FP tree for the transaction items given below. Min. support=02

| TId | Items |
|---|---|
| T1 | b,e |
| T2 | a,b,c,e |
| T3 | b,c,e |
| T4 | a,c |
| T5 | a |

# Vertical Data formats to find frequent item sets

| TID | List of Item_IDs |
|---|---|
| T100 | I1, I2, I5 |
| T200 | I2, I4 |
| T300 | I2, I3 |
| T400 | I1, I2, I4 |
| T500 | I1, I3 |
| T600 | I2, I3 |
| T700 | I1, I3 |
| T800 | I1, I2, I3, I5 |
| T900 | I1, I2, I3 |

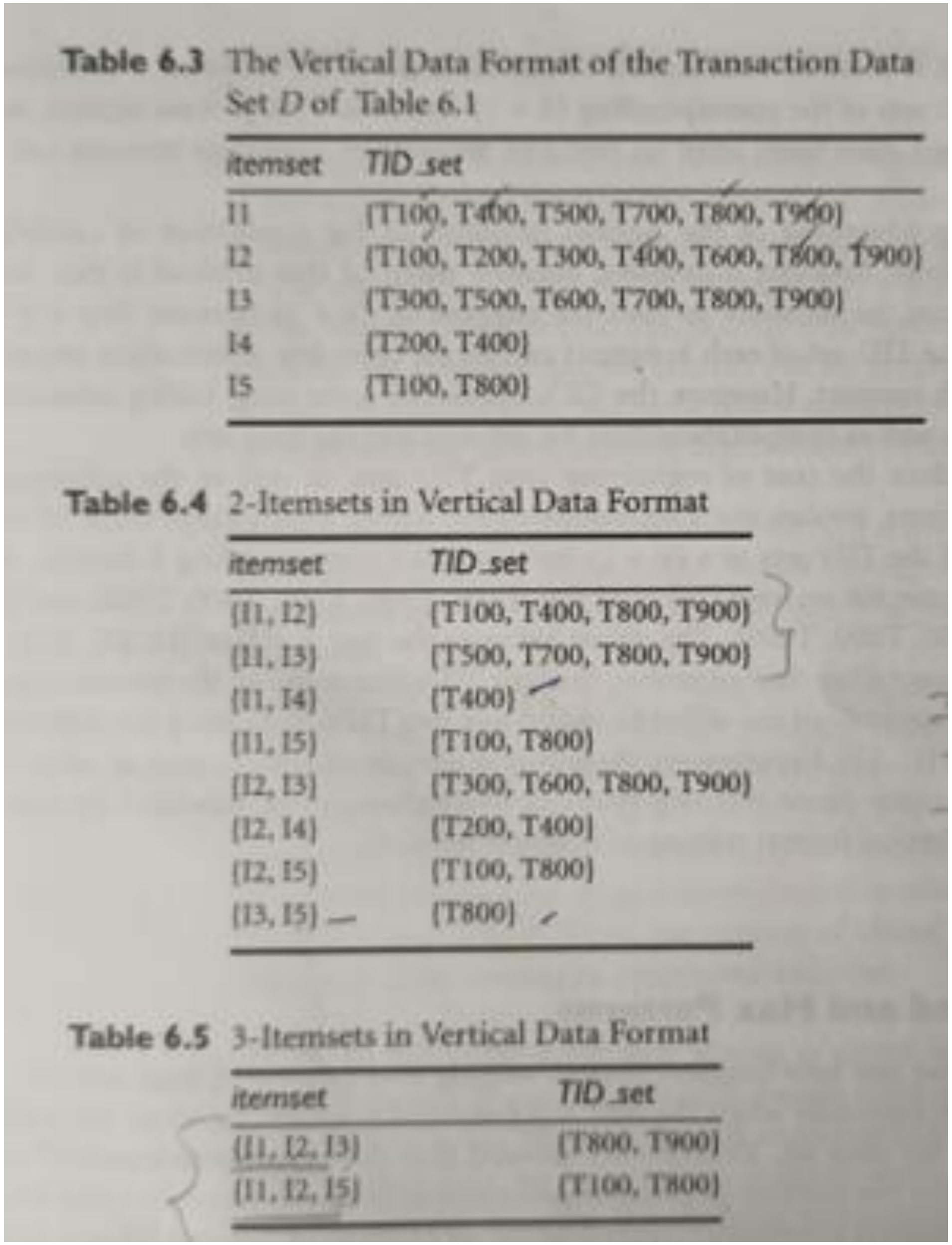


**Table 6.3** The Vertical Data Format of the Transaction Data Set *D* of Table 6.1

| *itemset* | *TID_set* |
|---|---|
| I1 | {T100, T400, T500, T700, T800, T900} |
| I2 | {T100, T200, T300, T400, T600, T800, T900} |
| I3 | {T300, T500, T600, T700, T800, T900} |
| I4 | {T200, T400} |
| I5 | {T100, T800} |

**Table 6.4** 2-Itemsets in Vertical Data Format

| *itemset* | *TID_set* |
|---|---|
| {I1, I2} | {T100, T400, T800, T900} |
| {I1, I3} | {T500, T700, T800, T900} |
| {I1, I4} | {T400} |
| {I1, I5} | {T100, T800} |
| {I2, I3} | {T300, T600, T800, T900} |
| {I2, I4} | {T200, T400} |
| {I2, I5} | {T100, T800} |
| {I3, I5} | {T800} |

**Table 6.5** 3-Itemsets in Vertical Data Format

| *itemset* | *TID_set* |
|---|---|
| {I1, I2, I3} | {T800, T900} |
| {I1, I2, I5} | {T100, T800} |