Lossless and Lossy Decomposition in 5th Normal Form

What is decomposition ?

To overcome data redundancy , partial and transitive functional dependency problems we break a relation into multiple relations in Normalization. Now , this process is also referred as decomposition. Sometimes we can join these decomposed tables in order to construct the original table. We have to be careful in this process to get back the original table properly. Because , sometimes after joining multiple tables together we may get the table with ambiguous and inconsistent data. Then that faulty decomposition method will be known as Lossy Decomposition. If we get the table error free then the method will be known as Lossless Decomposition.

Example 1

Relation : R

|  |  |  |
| --- | --- | --- |
| A | B | C |
| 1 | 2 | 1 |
| 2 | 2 | 2 |
| 3 | 3 | 2 |

|  |  |
| --- | --- |
| B | C |
| 2 | 1 |
| 2 | 2 |
| 3 | 2 |

Now decompose it into two tables (R1 and R2)

|  |  |
| --- | --- |
| A | B |
| 1 | 2 |
| 2 | 2 |
| 3 | 3 |

Now we will perform Natural join method to join these two tables.

Step 1 🡪 Cartesian Product

|  |  |  |  |
| --- | --- | --- | --- |
| A | B | B | C |
| 1 | 2 | 2 | 1 |
| 1 | 2 | 2 | 2 |
| 1 | 2 | 3 | 2 |
| 2 | 2 | 2 | 1 |
| 2 | 2 | 2 | 2 |
| 2 | 2 | 3 | 2 |
| 3 | 3 | 2 | 1 |
| 3 | 3 | 2 | 2 |
| 3 | 3 | 3 | 2 |

Step 2 🡪 Select where values of Attribute B are same

|  |  |  |  |
| --- | --- | --- | --- |
| A | B | B | C |
| 1 | 2 | 2 | 1 |
| 1 | 2 | 2 | 2 |
| 1 | 2 | 3 | 2 |
| 2 | 2 | 2 | 1 |
| 2 | 2 | 2 | 2 |
| 2 | 2 | 3 | 2 |
| 3 | 3 | 2 | 1 |
| 3 | 3 | 2 | 2 |
| 3 | 3 | 3 | 2 |

|  |  |  |
| --- | --- | --- |
| A | B | C |
| 1 | 2 | 1 |
| 1 | 2 | 2 |
| 2 | 2 | 1 |
| 2 | 2 | 2 |
| 3 | 3 | 2 |

So the New Table (R’) after projection will be

|  |  |  |  |
| --- | --- | --- | --- |
| A | B | B | C |
| 1 | 2 | 2 | 1 |
| 1 | 2 | 2 | 2 |
| 2 | 2 | 2 | 1 |
| 2 | 2 | 2 | 2 |
| 3 | 3 | 3 | 2 |

Now , in original table R we had 3 records and in R’ we have got 5 records.

There is no mistake in Natural Join Method. In this case we have done a Lossy Decomposition . So, what’s wrong we have done to decompose it ?

We have decomposed the original table R into R1 and R2 where attribute B was the reference field. And attribute B was not a candidate key in table R. We know only a primary key of one table can be present in another table as foreign key to act as a reference between these two tables.

In table R only attribute A has no redundant data and it can act as a primary key. Now if we decompose it again using attribute A as a reference then we can get a lossless decomposition.

Example 2

Relation : R

|  |  |  |
| --- | --- | --- |
| A | B | C |
| 1 | 2 | 1 |
| 2 | 2 | 2 |
| 3 | 3 | 2 |

|  |  |
| --- | --- |
| A | C |
| 1 | 1 |
| 2 | 2 |
| 3 | 2 |

Now decompose it into two tables (R1 and R2)

|  |  |
| --- | --- |
| A | B |
| 1 | 2 |
| 2 | 2 |
| 3 | 3 |

Now we will perform Natural join method to join these two tables.

Step 1 🡪 Cartesian Product

|  |  |  |  |
| --- | --- | --- | --- |
| A | B | A | C |
| 1 | 2 | 1 | 1 |
| 1 | 2 | 2 | 2 |
| 1 | 2 | 3 | 2 |
| 2 | 2 | 1 | 1 |
| 2 | 2 | 2 | 2 |
| 2 | 2 | 3 | 2 |
| 3 | 3 | 1 | 1 |
| 3 | 3 | 2 | 2 |
| 3 | 3 | 3 | 2 |

Step 2 🡪 Select where values of Attribute A are same

|  |  |  |  |
| --- | --- | --- | --- |
| A | B | A | C |
| 1 | 2 | 1 | 1 |
| 1 | 2 | 2 | 2 |
| 1 | 2 | 3 | 2 |
| 2 | 2 | 1 | 1 |
| 2 | 2 | 2 | 2 |
| 2 | 2 | 3 | 2 |
| 3 | 3 | 1 | 1 |
| 3 | 3 | 2 | 2 |
| 3 | 3 | 3 | 2 |

|  |  |  |
| --- | --- | --- |
| A | B | C |
| 1 | 2 | 1 |
| 2 | 2 | 2 |
| 3 | 3 | 2 |

So the New Table (R’) after projection will be

|  |  |  |  |
| --- | --- | --- | --- |
| A | B | A | C |
| 1 | 2 | 1 | 1 |
| 2 | 2 | 2 | 2 |
| 3 | 3 | 3 | 2 |

We have got the original table. So , the decomposition was lossless.