## UNIVERSITY ANALYTICS SYSTEM



### PREPARED BY:

|  |  |
| --- | --- |
| **ASSOCIATE NAME:** | **ASSOCIATE ID:** |
| Praveen Kumar | 2134624 ( team lead) |
| Utkarsh Mishra | 2126614 (team lead) |
| Buruhanudeen Nasurudeen | 2134997 |
| Haris Kumar | 2135032(team lead) |
| Nikhilanvitha Gorli | 2136619 |
| Jada Sravya | 2136620 |
| Chirag Mahajan | 2122261 |
| Mohammad Yasmin | 2134895 |
| Rakesh | 2136899 |
| Pandeeti Guru Satya Dattatreya | 2136976 |

**GUIDED BY:**

**MENTOR:** Kavin Kumar Govindaraj

**COACH:** Bhava Sruthi Arumugam

### LIST OF CONTENTS:

* **Introduction**
* Purpose and Scope of the Document 8
* Aim of the project 8
* Modules present in the project: 8
* [Architecture diagram 9](#_TOC_250002)
* [Flow chart 10](#_TOC_250001)
* [Functional Requirement 1 11-16](#_TOC_250000)
  + Student Dimension
  + Sources
  + Target
  + Mapping
  + Transformation
  + Workflow details
  + Task details

### Functional Requirement 2 17-21

* + College Dimension
  + Sources
  + Target
  + Mapping
  + Transformation
  + Workflow details
  + Task details

### Functional Requirement 3 22-25

* + Faculty Dimension
  + Sources
  + Target
  + Mapping
  + Transformation
  + Workflow details
  + Task details

### Functional Requirement 4 25-30

* + Department Dimension
  + Sources
  + Target
  + Mapping
  + Transformation
  + Workflow details
  + Task details

### Functional Requirement 5 30-34

* + Course Dimension
  + Sources
  + Target
  + Mapping
  + Transformation
  + Workflow details
  + Task details

### Functional Requirement 6 35-38

* + Time Dimension
  + Sources
  + Target
  + Mapping
  + Transformation
  + Workflow details
  + Task details

### Functional Requirement 7 39-44

* + Result Stagging Dimension
  + Sources
  + Target
  + Mapping
  + Transformation
  + Workflow details
  + Task details

### Functional Requirement 8 45-47

* + Result Aggregate Dimension
  + Sources
  + Target
  + Mapping
  + Transformation
  + Workflow details
  + Task details

### Functional Requirement 9 48-51

* + Integration
  + Workflow
  + Task Details

### Integrated Workflow and Task Details 51-52

* **Conclusion** 53

### LIST OF ABBREVATIONS:

|  |  |
| --- | --- |
| **TECHNICAL TERM** | **DESCRIPTION** |
| ETL | Extraction, Transformation and Load |
| SRC | Source |
| TGT | Target |
| SQ | Source Qualifier |
| WF | Workflow |
| JNR | Joiner |
| EXP | Expression |
| SRT | Sorter |
| FIL | Filter |
| UN | Union |
| SEQ | Sequence generator |
| RTR | Router |

**LIST OF TABLES:**

Table 1: Student Source Description Table 2: Student Target Description Table 3: Transformation Description Table 4: College Source Description Table 5: College Target Description Table 6: Transformation Description Table 7: Faculty Source Description Table 8: Faculty Target Description Table 9: Transformation Description Table 10: Department Source Description Table 11: Department Target Description Table 12: Transformation Description Table 13: Course Source Description Table 14: Course Target Description Table 15: Transformation Description Table 16: Time Source Description

Table 17: Time Target Description Table 18: Transformation Description

Table 19: Result Stagging Source Description

Table 20: Result Stagging Target Description

Table 21: Result Stagging Transformations Description Table 22: Result Fact Source Description

Table 23: Result Fact Target Description

Table 24: Results Fact Transformations Description Table 25: Result Aggregate Source Description Table 26: Result Aggregate Target Description

Table 27: Result Aggregate Transformations Description

## INTRODUCTION

As per the business requirements the project “University Analytics” is performed using Informatica PowerCenter (ETL Tool). Informatica is a data integration tool based on ETL architecture. Informatica PowerCenter is the widely used extraction, transformation, loading tool. The components within Informatica PowerCenter aid in extracting data from its source, transforming it as per business requirements and loading it into a target data warehouse. The system “University Analytics” has to be developed to manage the information of different students enrolled in colleges.

### Purpose & Scope of the project

The purpose of the software requirements document is to systematically capture requirements for the project and the system “University Analytics” to be developed. Both functional and non-functional requirements of this system are captured in this document. It also serves as the input for the project scoping. The scope of this document is limited to addressing the requirements from the business perspective. It is recommended that design aspects are not added in this document

### Modules in this Project

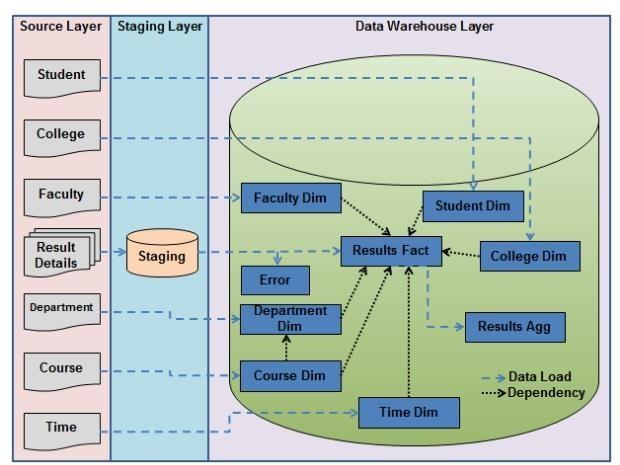
The following section will cover aspects related to University Analytics System (UAS) application. UAS is a system used for the analysis of various department/student/faculty performances.

The following are the modules in this proposed project

1. Student Dimension
2. College Dimension
3. Department Dimension
4. Course Dimension
5. Faculty Dimension
6. Time Dimension
7. Results Staging
8. Results Fact
9. Results Aggregate

### ARCHITECTURE DIAGRAM

**Logical Architecture:**



**Fig 1: Architecture Diagram**

### FLOW CHART

Mappings and Transformations

CDW\_ UAS\_ F\_AG G\_DA TA

CDW\_ UAS\_ D\_ST UDEN T

Import source files



START

Go to the designer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CDW\_ |  | CDW\_ |  | CDW\_ |
| UAS\_ |  | UAS\_ |  | UAS\_ |
| D\_CO |  | D\_FA |  | D\_DE |
| LLEG |  | CULT |  | PART |
| E |  | Y |  | MENT |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| CDW\_ |  | CDW\_ |  | CDW\_ |  |  |
| UAS\_ |  | UAS\_ |  | UAS\_ |  | CDW\_ |
| COUR |  | D\_TI |  | STG\_ |  | UAS\_ |
| SE |  | ME |  | RESU |  | F\_RES |
|  |  |  |  | LTS\_I |  | ULTS |
|  |  |  |  | N |  |  |

Workflow

Stop

Output

**Fig 2: Flow Chart**

# Functional Requirement 1

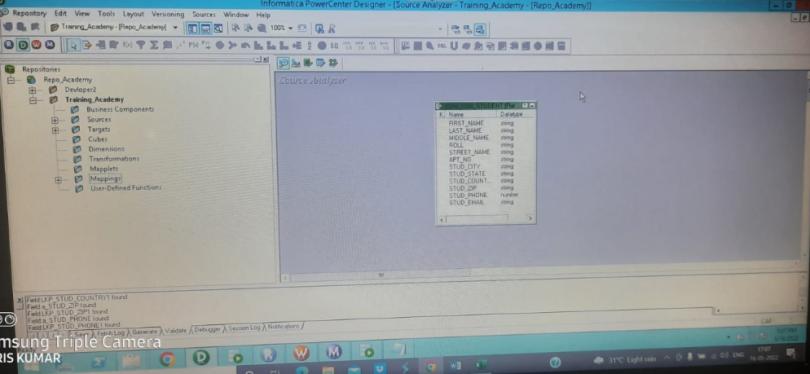
## STUDENT DIMENSION

**SOURCE**

Under the first Functional requirement, to generate the student Table of Type 2 SCD we created a Source flat files i.e.

|  |  |  |
| --- | --- | --- |
| **SOURCE FILE NAMES** | **DESCRIPTION** | **SOURCE FILES** |
| SRC\_CDW\_UAS\_Student.txt | This is a comma delimited(,) file with 12 input fields |  |

**Table 1: Student Source Description**

****

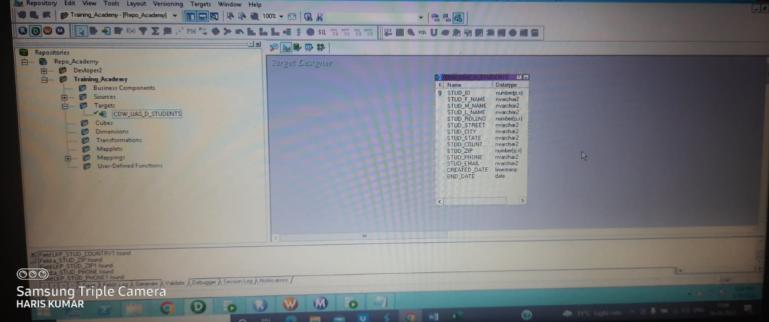
**Fig 3: Student Source Definition**

**TARGET**

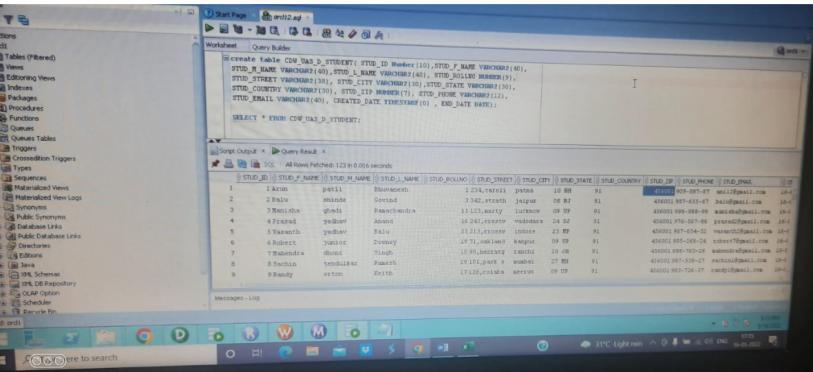
The target table has been generated in the Oracle database by using the source files as mentioned above.

|  |  |  |
| --- | --- | --- |
| **TARGET NAME** | **DESCRIPTION** | **TARGET FILE** |
| CDW\_UAS\_D\_STUDENT | Contains student data with history. |  |

**Table 2: Student Target Description**

****

**Fig 4: Student Target Definition**

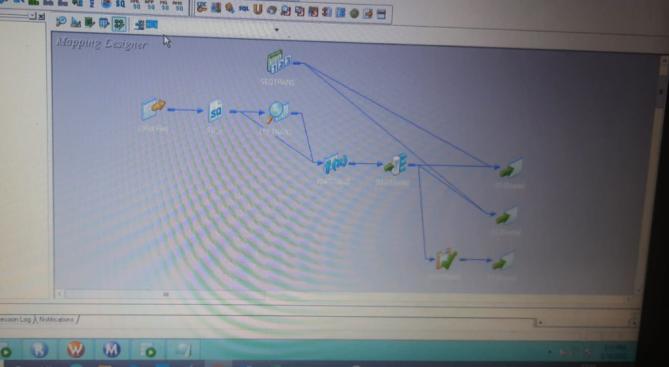
****

**Fig 5: Student Target Table**

**MAPPING**

The below figure shows the data flow from source to the target table by applying the necessary transformations to satisfy the business requirement

**Fig 6: Mapping 1**

****

**TRANSFORMATION**

The descriptions of the transformations used are mentioned in the table below.

|  |  |  |
| --- | --- | --- |
| **TRANSFORMATION NAME** | **DESCRIPTION** | **TRANSFORMATION LOGIC** |
| Lookup | CHECKING TARGET ROLL  AND SOURCE ROLL | STUDENT\_ROLL = ROLL |
|  | SQL OVERRIDE CONDITION | SELECT S.STUD\_ID as STUD\_ID, S.STUD\_F\_NAME as STUD\_F\_NAME, S.STUD\_M\_NAME as STUD\_M\_NAME, S.STUD\_L\_NAME as STUD\_L\_NAME, S.STUD\_STREET as STUD\_STREET, S.STUD\_CITY as STUD\_CITY, S.STUD\_STATE as STUD\_STATE, S.STUD\_COUNTRY as STUD\_COUNTRY, S.STUD\_ZIP as STUD\_ZIP, S.STUD\_PHONE as STUD\_PHONE, S.STUD\_EMAIL as STUD\_EMAIL, S.CREATED\_DATE as CREATED\_DATE, S.END\_DATE as END\_DATE, S.STUD\_ROLLNO as STUD\_ROLLNO FROM S  WHERE S.END\_DATE IS NULL |

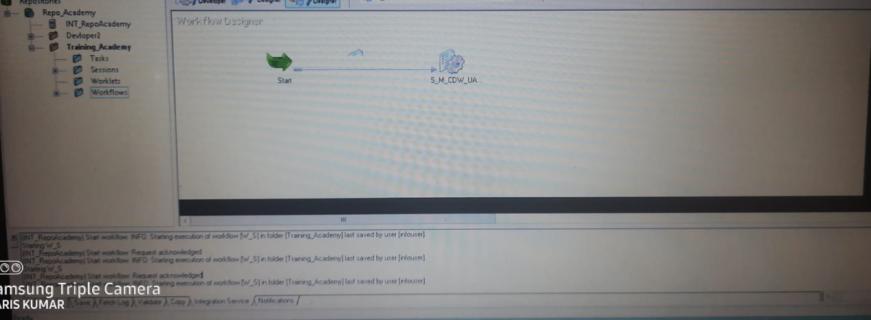
|  |  |  |
| --- | --- | --- |
|  |  |  |
| Expression | CONVERTING FIRST\_NAME TO TITLECASE | INITCAP(FIRST\_NAME) |
|  | CONVERTING MIDDLE\_NAME TO LOWERCASE | LOWER(MIDDLE\_NAME) |
|  | CONVERTING LOWER\_NAME TO TITLECASE | INITCAP(FIRST\_NAME) |
|  | ABORT THE SESSION IF THE ROLL\_NO IS  INVALID | IIF(ISNULL(ROLL),ABORT('Invalid Roll'), ROLL) |
|  | CONCATINATE APPARTMENT NO AND STREET NAME | APT\_NO || ',' || STREET\_NAME |
|  | CHANGE THE  FORMAT OF  PHONE\_NO TO XXX-XXX-XXXX | SUBSTR(STUD\_PHONE,1,3)||'- '||SUBSTR(STUD\_PHONE,4,3)||'- '||SUBSTR(STUD\_PHONE,7,4) |

|  |  |  |
| --- | --- | --- |
|  | CREATED\_DATE | SYSDATE |
|  | END\_DATE(FOR INSERTING) | NULL |
|  | END\_DATE(FOR UPDATING) | ADD\_TO\_DATE(SYSDATE,'SS',-1) |
|  | INSERT CONDITION (n\_INSERT) | IIF(ISNULL(LKP\_STUD\_ROLLNO),TRUE,FALS E) |
|  | UPDATE CONDITION (n\_UPDATE) | IIF(a\_ROLL=LKP\_STUD\_ROLLNO AND a\_FIRST\_NAME != LKP\_STUD\_F\_NAME OR a\_MIDDLE\_NAME != LKP\_STUD\_M\_NAME OR a\_LAST\_NAME != LKP\_STUD\_L\_NAME OR a\_STUD\_STREET != LKP\_STUD\_STREET OR STUD\_CITY != LKP\_STUD\_CITY1 OR STUD\_STATE != LKP\_STUD\_STATE1 OR STUD\_COUNTRY != LKP\_STUD\_COUNTRY1 OR STUD\_ZIP != LKP\_STUD\_ZIP1 OR v\_STUD\_PHONE != LKP\_STUD\_PHONE1 OR STUD\_EMAIL !=  LKP\_STUD\_EMAIL1,TRUE,FALSE) |
| Router | INSERTION | n\_INSERT |
|  | UPDATION | n\_UPDATE |
| Update Strategy | UPDATION INTO TARGET TABLE | DD\_UPDATE |
| Sequence Generator | GENERATING SURROGATE KEY FOR STUD\_ID | Start value=1 Increment by = 1 |

**Table 3: Transformation Description**

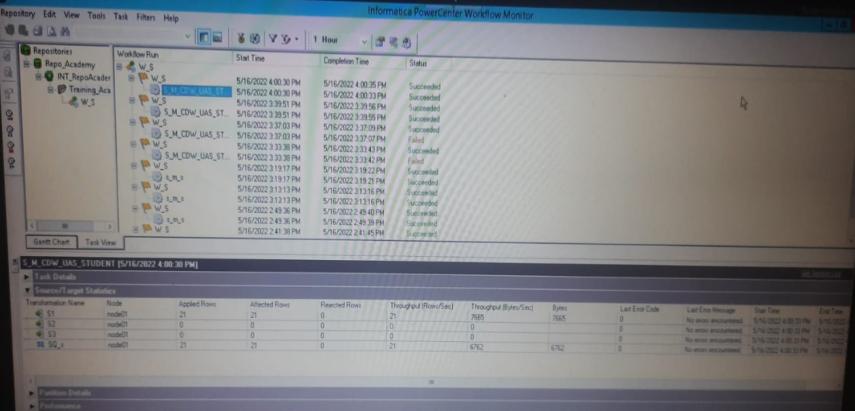
**WORKFLOW DETAILS**

**Fig 7: Workflow 1**

****

**TASK DETAILS**

The below figure shows that the session has executed successfully, and no rows were rejected. It also shows the execution time details

**Fig 8: Task Detail**

# Functional Requirement 2

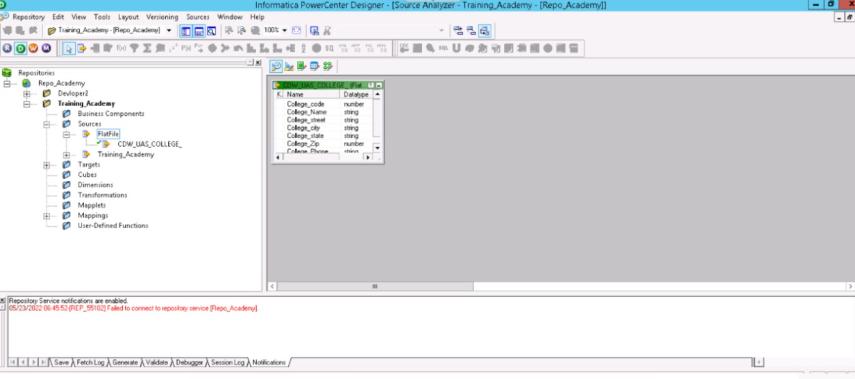
## COLLEGE DIMENSION

**SOURCE**

 Under the first Functional requirement, to generate the college dimension source table we created a Source flat file i.e.

|  |  |  |
| --- | --- | --- |
| **SOURCE FILE NAMES** | **DESCRIPTION** | **SOURCE FILES** |
| CDW\_UAS\_COLLEGE.txt | This is a fixed width file containing 7 fields. |  |

**Table 4: College Source Description**



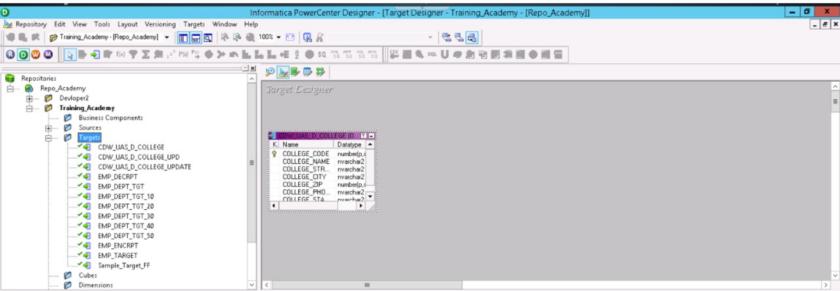
**Fig 9: College Dimension Source Definition**

**TARGET**

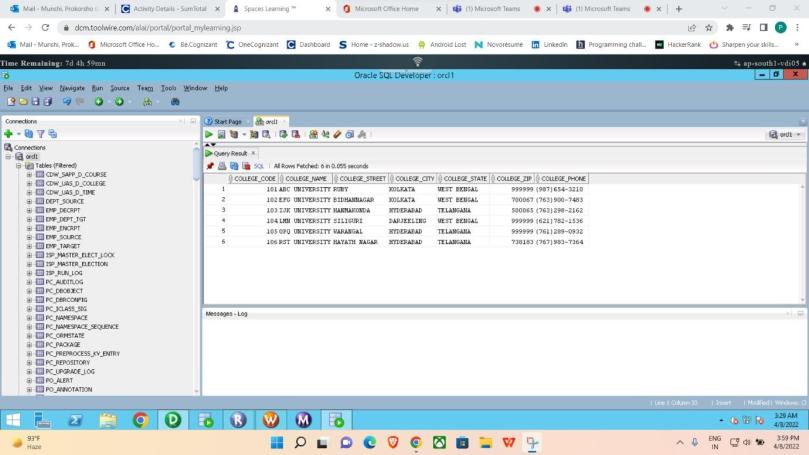
The target table has been generated in the Oracle database by using the source files as mentioned above.

|  |  |  |
| --- | --- | --- |
| TARGET FILE NAMES | DESCRIPTION | TARGET FILES |
| CDW\_UAS\_COLLEGE.txt | This is a fixed width file containing 7 fields. |  |

**Table 5: College Target Description**

****

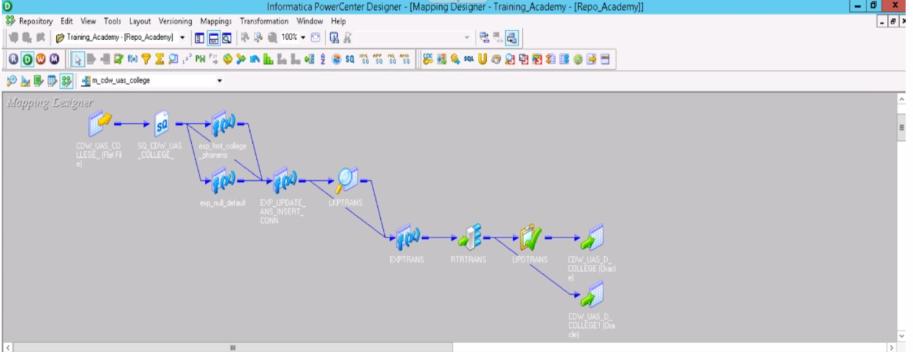
**Fig 10: College Target Definition**



**Fig 11: College Target Table**

**MAPPING**

The below figure shows the data flow from source to the target table by applying the necessary transformations to satisfy the business requirement



**Fig 12: Mapping 2**

**TRANSFORMATION**

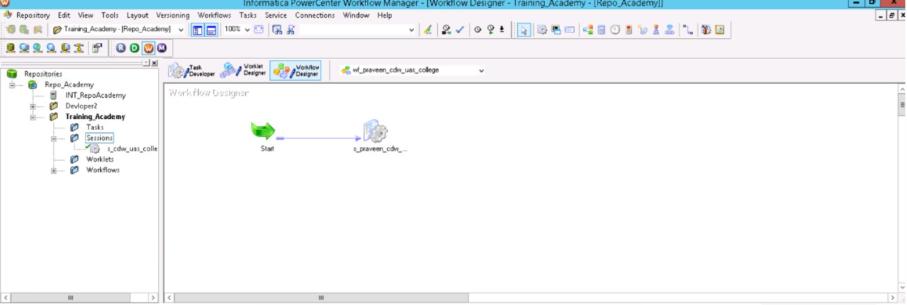
The descriptions of the transformations used are mentioned in the table below.

|  |  |  |
| --- | --- | --- |
| **TRANSFORM ATION**  **NAME** | **DESCRIPTION** | **TRANSFORMATION**  **LOGIC** |
| SORTER | TO SORT THE COLLEGE\_CODE | SORTING IN ASCENDING ORDER |
| LOOKUP | TO CHECK THE EXISTING COLLEGE  CODES IN THE DATABASE | COLLEGE\_CODE=COLLEGE\_CODE1 |
| EXPRESSION | IF THE SOURCE VALUE IS NULL, DEFAULT OTHERWISE DIRECT MOVE | IIF(ISNULL(COLLEGE\_ZIP),999999,COLLEGE\_ZI P) |
|  | To  format COLLEGE  \_PHONE in the (XXX)XXX-  XXXX format | '('||SUBSTR(COLLEGE\_PHONE,1,3)||')'||SUBSTR(C OLLEGE\_PHONE,4,3)||'- '||SUBSTR(COLLEGE\_PHONE,7,4) |
|  | FOR INSERTING NEW  RECORD(o\_new\_ record) | IIF(ISNULL(lkp\_COLLEGE\_CODE1),true,false) |

|  |  |  |
| --- | --- | --- |
|  | FOR UPDATED RECORD  (o\_updated\_record  ) | IIF(COLLEGE\_CODE = lkp\_COLLEGE\_CODE1 AND COLLEGE\_NAME != lkp\_COLLEGE\_NAME1 OR  COLLEGE\_STREET != lkp\_COLLEGE\_STREET1 OR COLLEGE\_CITY != lkp\_COLLEGE\_CITY1 OR COLLEGE\_STATE != lkp\_COLLEGE\_STATE1  OR v\_COLLEGE\_ZIP != lkp\_COLLEGE\_ZIP1 OR v\_COLLEGE\_PHONE !=  lkp\_COLLEGE\_PHONE1,true,false) |
| ROUTER | INSERTION | NEW\_RECORD= o\_new\_record |
|  | UPDATE | UPDATED\_RECORED- o\_updated\_record |
| UPDATE STRATEGY | UPDATE DATA | DD\_UPDATE |

**Table 6: College Transformation Description**

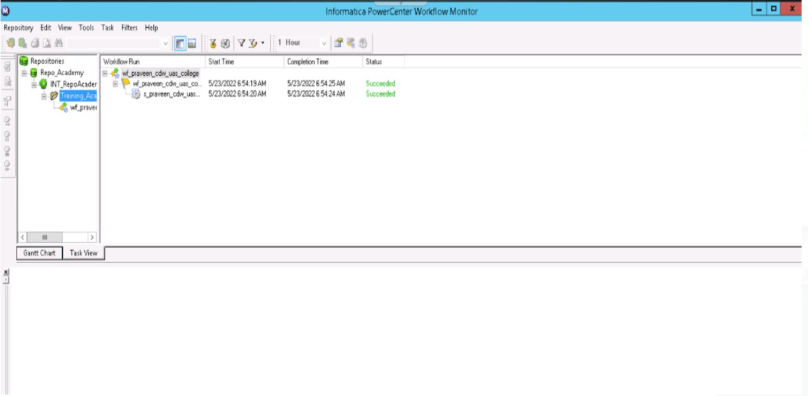
**WORKFLOW DETAILS**



**Fig 13: Workflow 2**

**TASK DETAILS**

The below figure shows that the session has executed successfully and no rows were rejected. It also shows the execution time details.



**Fig 14: Task Detail 2**

# Functional Requirement 3

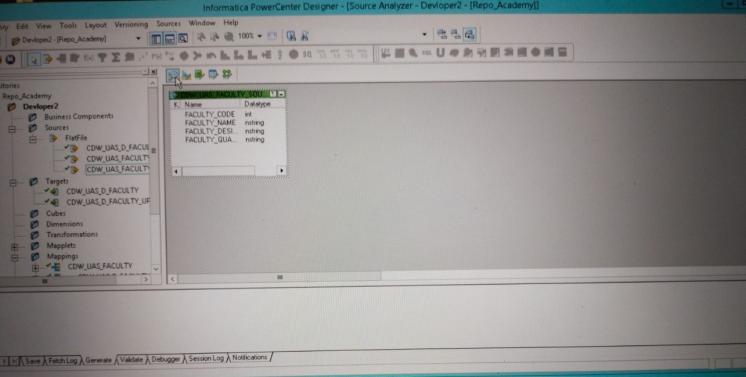
## FACULTY DIMENSION

**SOURCE**

Under the first Functional requirement, to generate the Faculty Table we created a Source flat file i.e.

|  |  |  |
| --- | --- | --- |
| **SOURCE FILE NAMES** | **DESCRIPTION** | **SOURCE FILES** |
| CDW\_UAS\_FACULTY.txt | This is a comma delimited source with 4 fields |  |

**Table 7: Faculty Source Description**

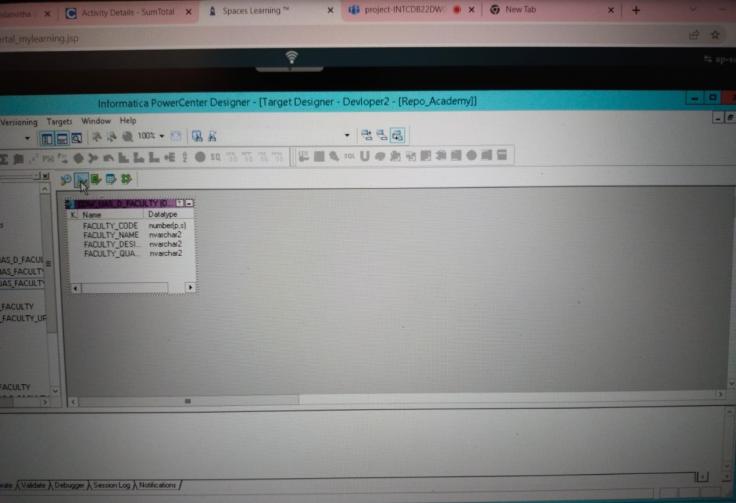
**Fig 15: Faculty Source Definition**

**TARGET**

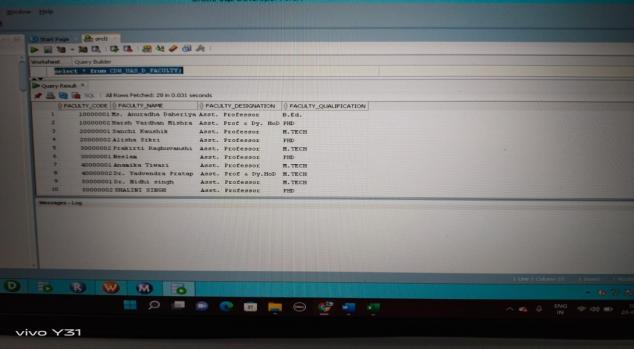
The group table has been generated in the Oracle database by using the source files as mentioned above.

|  |  |  |
| --- | --- | --- |
| **TARGET NAME** | **DESCRIPTION** | **TARGET FILE** |
| CDW\_UAS\_D\_FACULTY | Contains results faculty details of the colleges |  |

**Table 8: Faculty Target Description**

****

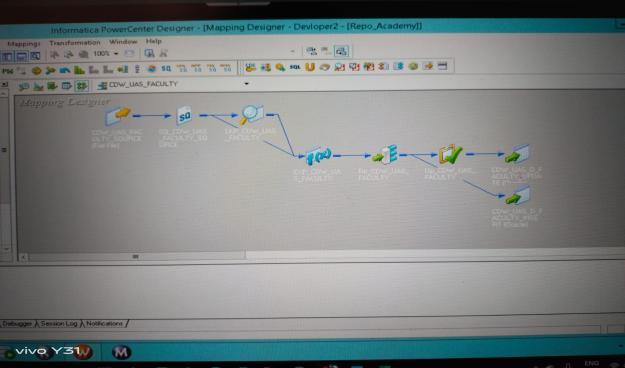
**Fig 16: Faculty Target Definition**



**Fig 17: Faculty Target Table**

**MAPPING**

The below figure shows the data flow from source to the target table by applying the necessary transformations to satisfy the business requirement.



**Fig 18: Mapping 3**

**TRANSFORMATION**

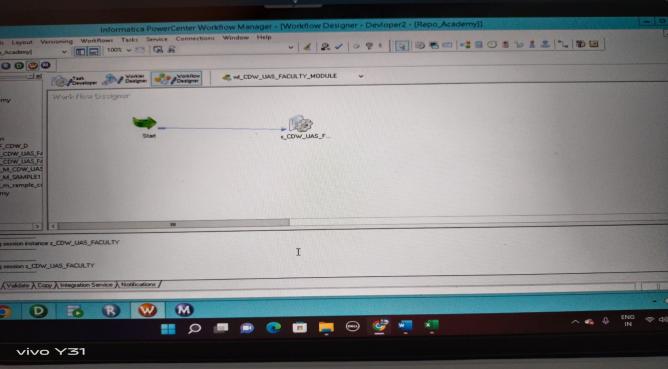
The descriptions of the transformations used are mentioned in the table below.

|  |  |  |
| --- | --- | --- |
| **TRANSFORMATION NAME** | **DESCRIPTION** | **TRANSFORMATION LOGIC** |
| **SORTER** | TO SORT FACULTY\_CODE | IN ASCENDING ORDER |

|  |  |  |
| --- | --- | --- |
| **LOOKUP** | TO CHECK FACULTY\_CODE PRESENT IN  TARGET TABLE OR NOT | FACULTY\_CODE=src\_FACULTY\_CODE |
| **EXPRESSION** | INSERTION | IIF(ISNULL(LKP\_FACULTY\_CODE),true,false) |
|  | UPDATION | IIF(FACULTY\_CODE = LKP\_FACULTY\_CODE AND FACULTY\_NAME != LKP\_FACULTY\_NAME1 OR FACULTY\_DESIGNATION != LKP\_FACULTY\_DESIGNATION1 OR FACULTY\_QUALIFICATION !=  LKP\_FACULTY\_QUALIFICATION1,true,false) |
| **ROUTER** | DIVIDE INTO TWO GROUPS | FOR INSERT, NEW\_RECORD\_FLAG FOR UPDATE, UPDATE\_RECORD\_FLAG |
| **UPDATE STRATEGY** | TO UPDATE THE PREVIOUS DATA | DD\_UPDATE |

**Table 9: Transformation Description**

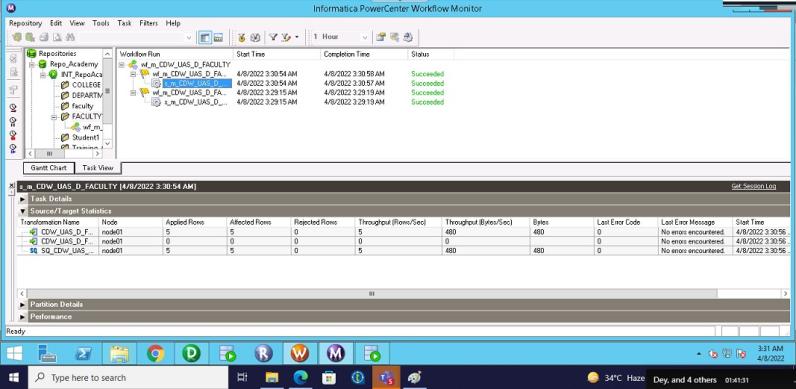
**WORKFLOW DETAILS**

****

**Fig 19: Workflow 3**

**TASK DETAILS**

The below figure shows that the session has executed successfully, and no rows were rejected. It also shows the execution time details.



**Fig 20: Task Details 3**

# Functional Requirement 4

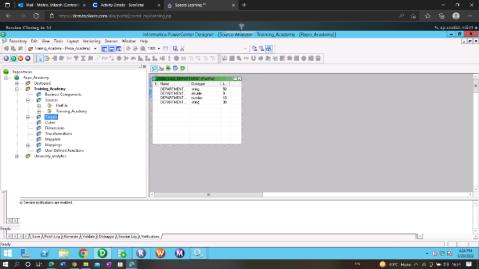
## DEPARTMENT DIMENSION

**SOURCE**

Under the first Functional requirement, to generate the Department Table we created a source file.

|  |  |  |
| --- | --- | --- |
| **SOURCE FILE NAMES** | **DESCRIPTION** | **SOURCE FILES** |
| CDW\_UAS\_DEPARTMENT.txt | This is a pipe delimited file (|) containing 4Fields. |  |

**Table 10: Department Source Description**

****

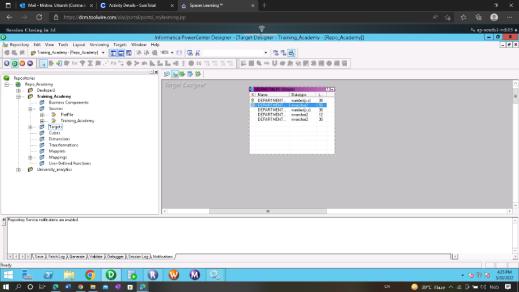
**Fig 21: Department Source Definition**

**TARGET**

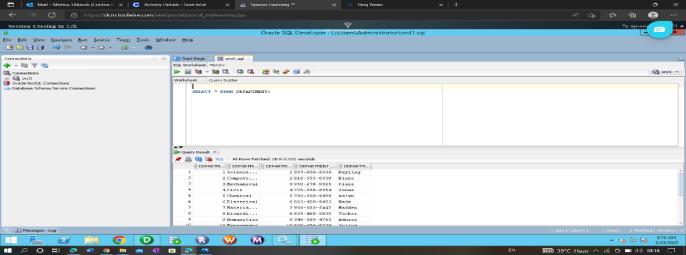
The group table has been generated in the Oracle database by using the source files as mentioned above.

|  |  |  |
| --- | --- | --- |
| **TARGET NAME** | **DESCRIPTION** | **TARGET FILE** |
| CDW\_UAS\_D\_DEPARTMENT | Contains results for all the exams happened across all colleges, with history |  |

**Table 11: Department Target Description**

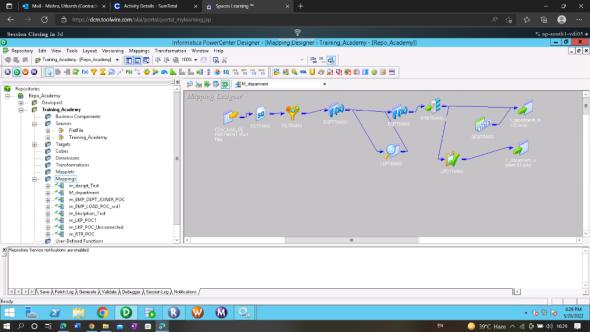


**Fig 22: Department Target Definition**

****

**Fig 23: Department Target Table**

**MAPPING**

****

**Fig 24: Mapping 4**

**TRANSFORMATION**

The descriptions of the transformations used are mentioned in the table below.

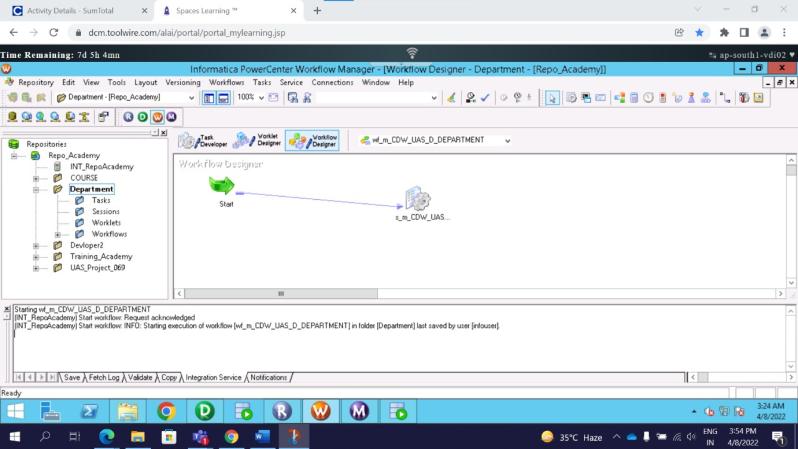
|  |  |  |
| --- | --- | --- |
| **TRANSFORMA TION**  **NAME** | **DESCRIPTIO N** | **TRANSFORMATION LOGIC** |
| FILTER | DIRECT MOVE (ABORT THE SESSION IF THE DEPARTMEN | IIF(IS\_NUMBER(DEPARTMENT\_NO),true,AB ORT('INVALID INPUT')) |

|  |  |  |
| --- | --- | --- |
|  | T\_NO IS INVALID). |  |
| EXPRESSION | REMOVE  %SIGN IF PRESENT AND LOAD ELSE DIRECT MOVE | REPLACESTR(0,DEPARTMENT\_NAME,'%','') |
|  | CONVERT TO NUMBER | TO\_DECIMAL(DEPARTMENT\_NO) |
|  | CONVERT TO STRING | TO\_CHAR(DEPARTMENT\_PHONE) |
|  | STANDARDIZ E TO XXX- XXX-XXXX | SUBSTR(v\_DEPARTMENT\_PHONE\_CHAR,1,3  )||'- '||SUBSTR(v\_DEPARTMENT\_PHONE\_CHAR,4,  3)||'- '||SUBSTR(v\_DEPARTMENT\_PHONE\_CHAR,7, 4) |
| LOOKUP | LOOKING UP FOR DEPARTMEN T\_NO IN THE TARGET TABLE | DEPARTMENT\_NO = o\_DEPARTMENT\_NO |
| EXPRESSION | INSERTION OF NEW DATA | IIF(ISNULL(lkp\_DEPARTMENT\_NO),true,false) |
|  | UPDATION OF NEW DATA | IIF(o\_DEPARTMENT\_NO = lkp\_DEPARTMENT\_NO AND o\_DEPARTMENT\_NAME != lkp\_DEPARTMENT\_NAME OR o\_DEPARTMENT\_PHONE != lkp\_DEPARTMENT\_PHONE OR DEPARTMENT\_HOD !=  lkp\_DEPARTMENT\_HEAD,true,false) |
| SEQUENCE GENERATOR | TO GIVE SURROGATE KEY TO DEPARTMEN T\_ID | Start value=1 Increment by 1 |

|  |  |  |
| --- | --- | --- |
| UPDATE STRATEGY | TO UPDATE THE DATA | dd\_update |

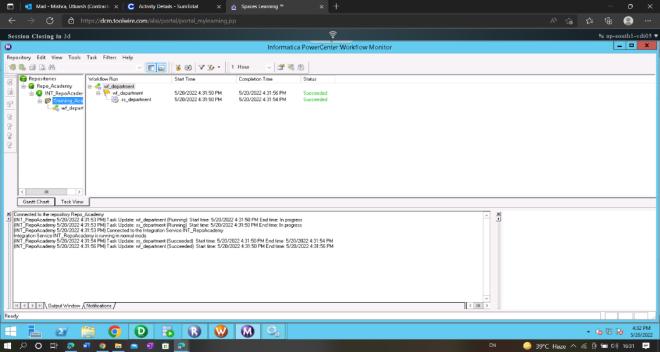
**Table 12: Transformation Details**

**WORKFLOW DETAILS**



**Fig 25: Workflow 4**

**TASK DETAILS**

The below figure shows that the session has executed successfully, and no rows were rejected. It also shows the execution time details.

**Fig 26: Task Details 4**

# Functional Requirement 5

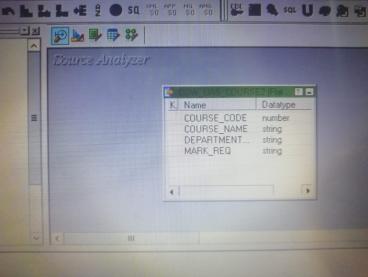
## COURSE DIMENSION

**SOURCE**

Under the first Functional requirement, to generate the Course Table we created a Source flat file i.e.

|  |  |  |
| --- | --- | --- |
| **SOURCE FILE NAMES** | **DESCRIPTION** | **SOURCE FILES** |
| CDW\_UAS\_COURSE.txt | This is a comma delimited file file containing 4 fields. | CDW\_UAS\_COURSE.t  xt |

**Table 13: Course Source Description**

****

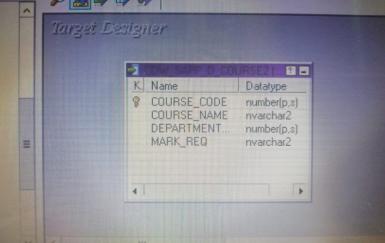
**Fig 27: Course Source Definition**

**TARGET**

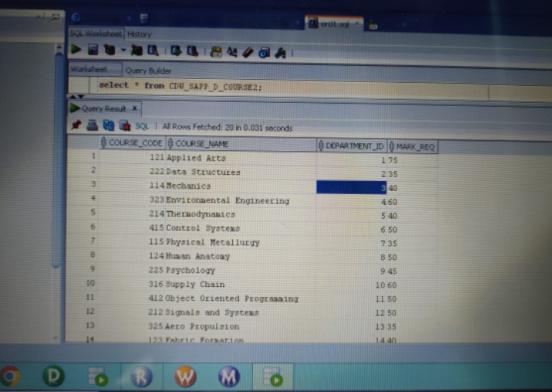
The group table has been generated in the Oracle database by using the source files as mentioned above.

|  |  |  |
| --- | --- | --- |
| **TARGET NAME** | **DESCRIPTION** | **TARGET FILE** |
| CDW\_SAPP\_D\_COURSE | Contains results for all the exams happened across all colleges, with history |  |

**Table 14: Course Target Description**

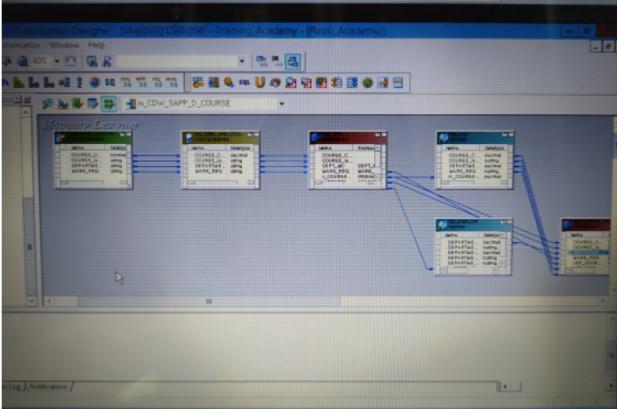
****

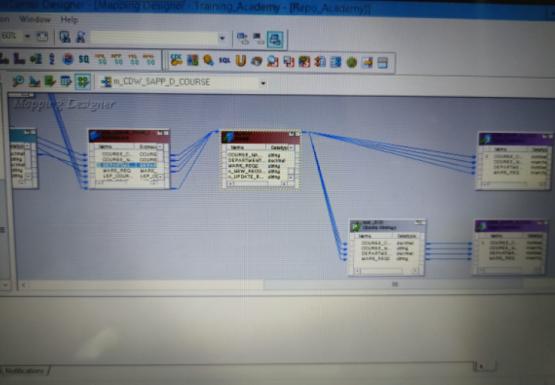
**Fig 28: Course Target Definition**



**Fig 29: Course Target Table**

**MAPPING**

****The below figure shows the data flow from source to the target table by applying the necessary transformations to satisfy the business requirement

****

**Fig 30: Mapping 5**

**TRANSFORMATION**

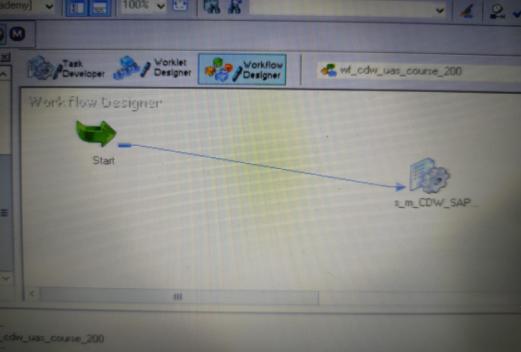
The descriptions of the transformations used are mentioned in the table below.

|  |  |  |
| --- | --- | --- |
| **TRANSFORMATIO N**  **NAME** | **DESCRIPTION** | **TRANSFORMATION LOGIC** |
| EXPRESSION | ABORT THE SESSION IF CODE IS INVALID | IIF(ISNULL(COURSE\_CODE),ABORT('Invalid  Course Code'),COURSE\_CODE) |
|  | TRIM THE TRAILING SPACES AND  LOAD | RTRIM(COURSE\_NAME,' ') |
|  | TO CHANGE DEPARTMENT\_N O FROM STRING TO DECIMAL | TO\_DECIMAL(DEPT\_NO) |
| LOOKUP1 | LOOKING FOR DEPARTMENT\_N O FROM DEPARTMENT  TABLE AND | DEPARTMENT\_NO = in\_DEPT \_NO |

|  |  |  |
| --- | --- | --- |
|  | LOADING DEPARTMENT\_I  D FOR THE TARGET |  |
| LOOKUP2 | LOOKING UP FOR COURSE\_CODE IN TARGET TABLE | COURSE\_CODE = in\_COURSE\_CODE |
| EXPRESSION | INSERT | IIF(ISNULL(LKP\_COURSE\_CODE),TRUE,FA LSE) |
|  | UPDATE | IIF(LKP\_COURSE\_CODE = COURSE\_CODE AND COURSE\_NAME != LKP\_COURSE\_NAME OR DEPARTMENT\_ID  != LKP\_DEPARTMENT\_ID OR MARK\_REQ  != LKP\_MARK\_REQ,TRUE,FALSE) |
| ROUTER | INSERT | o\_NEW\_RECORD |
|  | UPDATE | o\_UPDATE\_RECORD |
| UPDATE STRATEGY | TO UPDATE DATA | 1 |

**Table 15: Transformation Details**

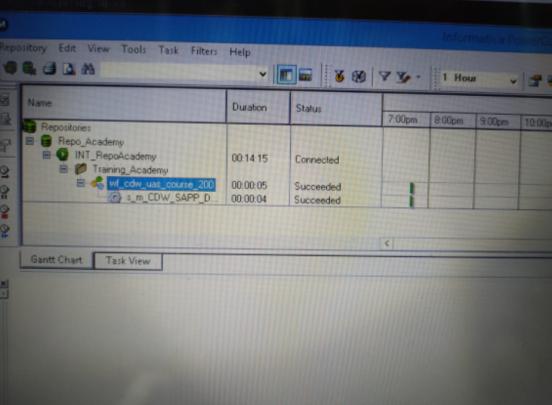
**WORKFLOW DETAILS**

****

**Fig 31: Workflow 5**

**TASK DETAILS**

The below figure shows that the session has executed successfully and no rows were rejected. It also shows the execution time details

****

**Fig 32: Task Details 5**

# Functional Requirement 6

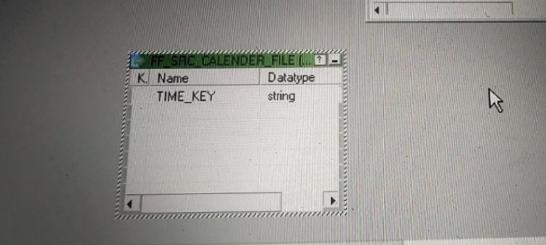
## TIME DIMENSION

**SOURCE**

Under the first Functional requirement, to generate the Time dimension Table we created a Source flat file i.e.

|  |  |  |
| --- | --- | --- |
| **SOURCE FILE NAMES** | **DESCRIPTION** | **SOURCE FILES** |
| CDW\_UAS\_TIME.txt.txt | This is a fixed width file containing 5 fields. |  |

**Table 16: Time Source Description**

****

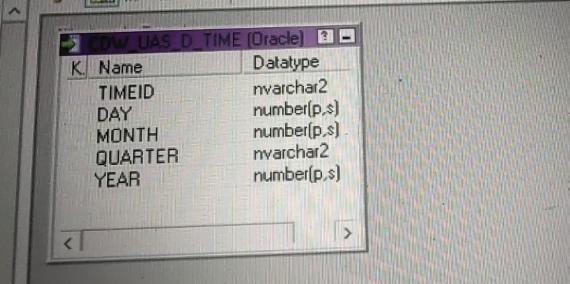
**Table 33: Time Source Definition**

**TARGET**

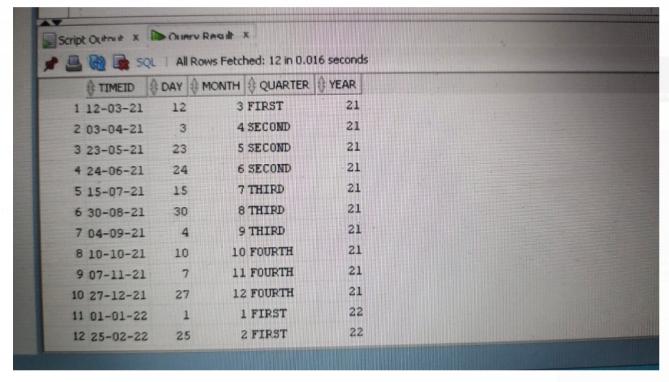
The target table has been generated in the Oracle database by using the source files as mentioned above.

|  |  |  |
| --- | --- | --- |
| **TARGET NAME** | **DESCRIPTION** | **TARGET FILE** |
| CDW\_UAS\_D\_TIME | Contains the time data |  |

**Table 17: Time Target Description**

****

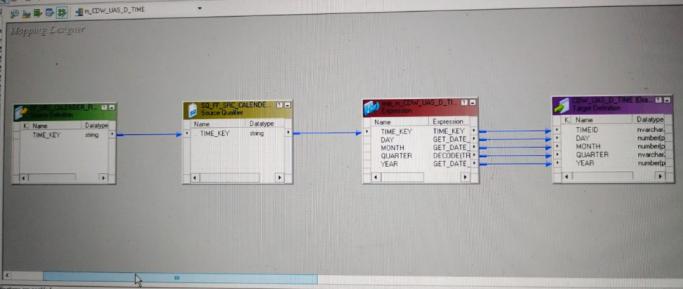
**Fig 34: Time Target Definition**

****

**Fig 35: Time Target Table**

**MAPPING**

The below figure shows the data flow from source to the target table by applying the necessary transformations to satisfy the business requirement.

****

**Fig 36: Mapping 6**

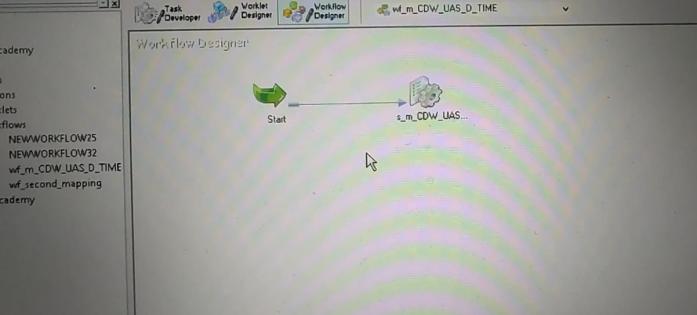
**TRANSFORMATION**

The descriptions of the transformations used are mentioned in the table below.

|  |  |  |
| --- | --- | --- |
| **TRANSFORMATION**  **NAME** | **DESCRIPTION** | **TRANSFORMATION** **LOGIC** |
| EXPRESSION | TO GET THE DATE PART FROM TIME\_ID(V\_DATE) | TO\_DATE(time\_key,'DD-MM-YYYY') |
|  | GET THE MONTH PART | GET\_DATE\_PART( V\_DATE, 'MM' ) |
|  | GET THE DAY PART | GET\_DATE\_PART( V\_DATE, 'DD' ) |
|  | GET THE YEAR PART | GET\_DATE\_PART( V\_DATE, 'YYYY' ) |
|  | GET THE QUARTER PART | IIF(V\_MONTH >= 1 AND V\_MONTH <= 3,'FIRST',IIF(V\_MONTH>=4 AND V\_MONTH <=  6,'SECOND',IIF(V\_MONTH>=7 AND V\_MONTH <= 9,'THIRD','FOURTH'))) |

**Table 18: Transformation Details**

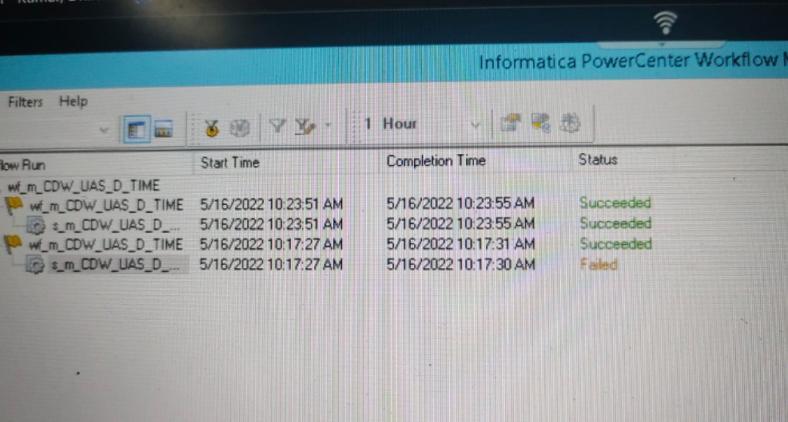
**WORKFLOW DETAILS**



**Fig 37: Workflow 6**

**TASK DETAILS**

The below figure shows that the session has executed successfully, and no rows were rejected. It also shows the execution time details.



**Fig 38: Task Details**

# Functional Requirement 7

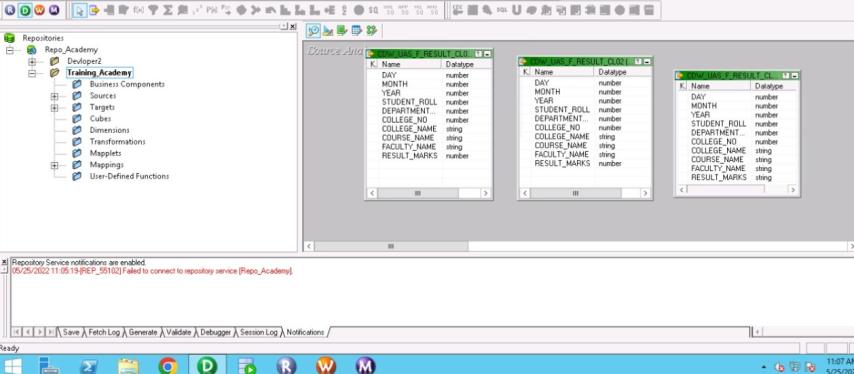
## RESULT STAGING

**SOURCE**

Under the first Functional requirement, to generate the result staging Table we created three Source flat files i.e.

|  |  |  |
| --- | --- | --- |
| **SOURCE FILE NAMES** | **DESCRIPTION** | **SOURCE FILES** |
| CDW\_UAS\_F\_RESULT\_CL1.txt | This is a Comma separated file containing 12 fields. |  |
| CDW\_UAS\_F\_RESULT\_CL2.txt | This is a Comma separated file containing 12 fields |  |
| CDW\_UAS\_F\_RESULT\_CL3.txt | This is a fixed width file containing 12 fields. |  |

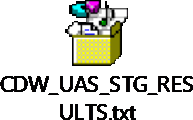
**Table 19: Result Staging Source Description**



**Fig 39: Result Staging Source Definition**

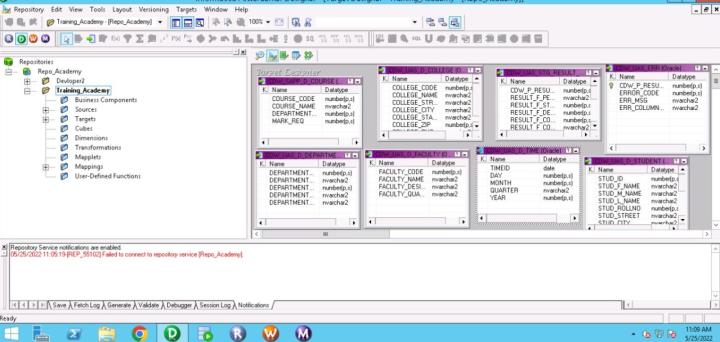
**TARGET**

The target table has been generated in the Oracle database by using the source files as mentioned above.

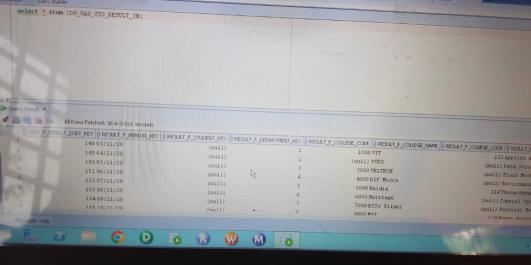


|  |  |  |
| --- | --- | --- |
| **TARGET NAME** | **DESCRIPTION** | **TARGET FILE** |
| CDW\_UAS\_F\_RESULT | Containing fact table data from result staging |  |
| CDW\_UAS\_ERR | Contains Error data from Result staging |  |

**Table 20: Result Staging Target Description**



**Fig 40: Result Staging Target Definition**



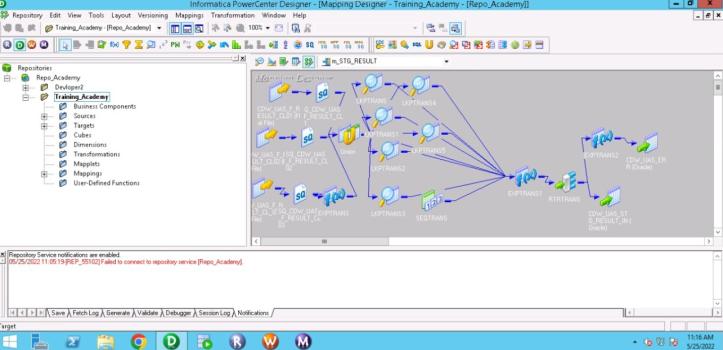
**Fig 41: Result Staging Database**



**Fig 42: Result Staging Error Database**

**MAPPING**

 The below figure shows the data flow from source to the target table by applying the necessary transformations to satisfy the business requirement.



**Fig 43: Result Staging Mapping**

**TRANSFORMATION**

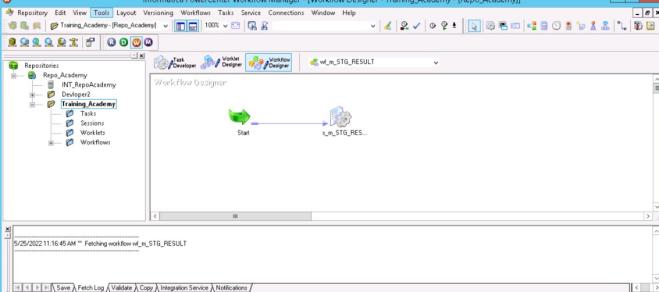
The descriptions of the transformations used are mentioned in the table below.

|  |  |  |
| --- | --- | --- |
| **TRANSFOR MATION**  **NAME** | **DESCRIPTION** | **TRANSFORMATION**   **LOGIC** |
| Union transformation | for combining the three source files data | created two groups  1.comma   1. comma 2. fixed delimited |
| Lookup transf ormation | lookup with student table | STUD\_ROLLNO=IN\_Student\_roll |
| Lookup transformation | lookup with course table | COURSE\_NAME=IN\_COURSE\_NAME |
| Lookup transformation | Lookup with faculty table | FACULTY\_NAME=IN\_FACULTY\_NAME |
| Lookup transformation | Lookup with time table | DAY=IN\_DAY,MONTH=IN\_MONTH,YEAR=IN\_YE  AR |

|  |  |  |
| --- | --- | --- |
| Lookup transformation | Lookup with college table | COLLEGE\_NAME=IN\_COLLEGE\_NAME |
| Lookup transformation | Lookup with department table | DEPARTMENT\_NO=IN\_DEPARTMENT\_NO |
| Expression transformation | Changing the data type | TO\_DECIMAL(LKP\_MARK\_REQ) |
|  | for segregating pass or fail | IIF(NOT ISNULL(RESULT\_MARKS),IIF(RESULT\_MARKS>= V\_MARK\_REQ,'PASS','FAIL'),NULL) |
|  | For generating error code | IIF(NOT IS\_NUMBER(TO\_CHAR(RESULT\_MARKS)) AND ISNULL(LKP\_TIME\_ID)AND ISNULL(V\_RESULT\_GRADE),1, IIF(NOT IS\_NUMBER(TO\_CHAR(RESULT\_MARKS))AND ISNULL(LKP\_TIME\_ID),2,IIF(ISNULL(LKP\_TIME\_I D) AND ISNULL(V\_RESULT\_GRADE),3,IIF(NOT IS\_NUMBER(TO\_CHAR(RESULT\_MARKS)) AND ISNULL(V\_RESULT\_GRADE),4, IIF(NOT IS\_NUMBER( TO\_CHAR(RESULT\_MARKS)),5, IIF(ISNULL(LKP\_TIME\_ID),6, IIF(ISNULL( V\_RESULT\_GRADE),7,0))))))) |
| Router Transformatio n | For creating error and fact tables | O\_ERROR\_FLAG != 0 |
| Expression transformation | for creating error message | IIF(O\_ERROR\_FLAG1=1,'INVALID MARKS,DATEAND GRADE',IIF(O\_ERROR\_FLAG1=2,'INVALID DATE AND MARKS',IIF(O\_ERROR\_FLAG1=3,'INVALID DATE AND GRADE',IIF(O\_ERROR\_FLAG1=4,'INVALID GRADE AND MARKS',IIF(O\_ERROR\_FLAG1=5,'RESULT MARKS CONTAINS INCORRECT DATA',IIF(O\_ERROR\_FLAG1=6,'INVALID DATE',IIF(O\_ERROR\_FLAG1=7,'RESULT GRADE IS  EMPTY'))))))) |
|  | For creating error column | IIF(O\_ERROR\_FLAG1=1,'RESULT\_MARKS,RESULT  \_F\_PERIOD\_KEY,RESULT\_GRADE',IIF(O\_ERROR\_ FLAG1=2,'RESULT\_F\_PERIOD\_KEY,RESULT\_MAR KS',IIF(O\_ERROR\_FLAG1=3,'RESULT\_GRADE,RES ULT\_F\_PERIOD\_KEY',IIF(O\_ERROR\_FLAG1=4,'RES ULT\_GRADE,RESULT\_MARKS',IIF(O\_ERROR\_FLA G1=5,'RESULT\_MARKS',IIF(O\_ERROR\_FLAG1=6,'R ESULT\_F\_PERIOD\_KEY',IIF(O\_ERROR\_FLAG1=7,'R ESULT\_GRADE'))))))) |

**Table 21: Transformation**

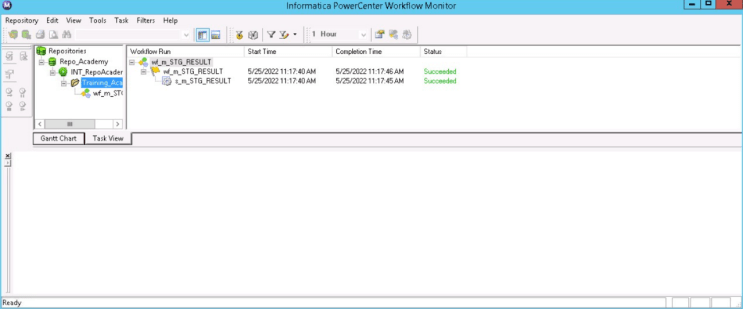
**WORKFLOW DETAILS**

****

**Fig 44: Result staging Workflow**

**TASK DETAILS**

The  below figure shows that the session has executed successfully and no rows were rejected. It also shows the execution time details.



**Fig 45: Result Staging Task detail**

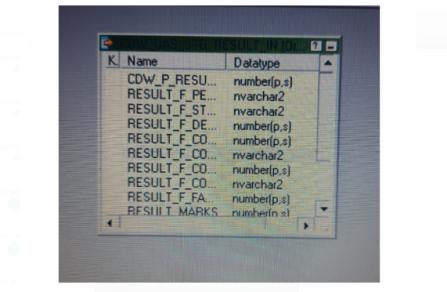
# Functional Requirement 8

## RESULT FACT

**SOURCE**

|  |  |  |
| --- | --- | --- |
| **SOURCE FILE NAMES** | **DESCRIPTION** | **SOURCE FILES** |
| CDW\_UAS\_STG\_RESULT\_IN | Containing fact table data from result staging | Target Table of Result Staging |

**Table 22: Result Fact Source Description**



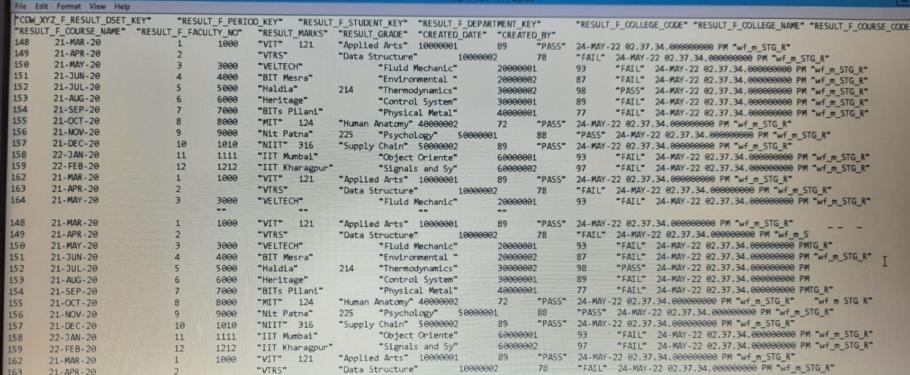
**Fig 46: Result Fact Source Definition**

**TARGET**

|  |  |
| --- | --- |
| TARGET NAME | DESCRIPTION |
| CDW\_UAS\_F\_RESULT | Contains Result Data. |

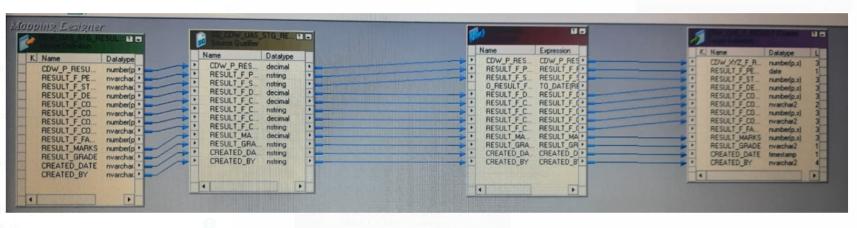
****

**Table 23: Result Fact Target Description**



**Fig 47: Result Fact Target**

**MAPPING**

****

**Fig 48: Result Fact Mapping**

**TRANSFORMATION**

The descriptions of the transformations used are mentioned in the table below.

|  |  |  |
| --- | --- | --- |
| **TRANSFORMATION NAME** | **DESCRIPTION** | **TRANSFORMATION** |
| Expression | This transformation is used for direct link | Direct Move |
|  | Converts a character string to a date datatype | TO\_DATE(RESULT\_F\_PERIOD\_KEY,'DD- MM-YYYY') |

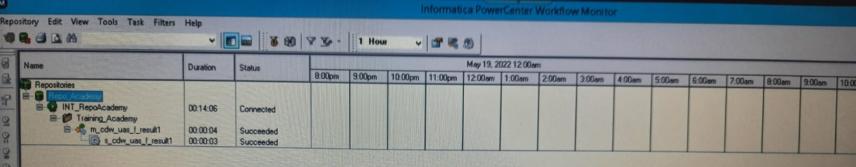
**Table 24: Transformation Details**

**WORKFLOW DETAILS**

****

**Fig 49: Result Fact Workflow**

**TASK DETAILS**

****

**Fig 50: Result Fact Task detail**

# Functional Requirement 9

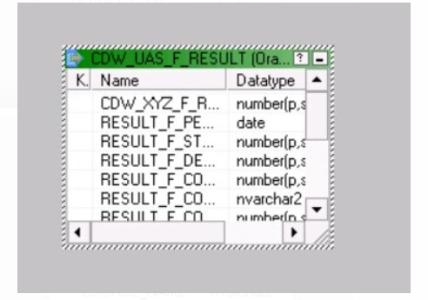
## RESULT AGGREGATE

**SOURCE**

Under the first Functional requirement, to generate the Result Aggregate Table we have created a Source flat file i.e.

|  |  |  |
| --- | --- | --- |
| **SOURCE FILE NAME** | **DESCRIPTION** | **SOURCE FILE** |
| CDW\_UAS\_F\_RESULT | Containing Fact table data | Target from Result Fact |

**Table 25: Results Aggregate Source Description**



**Fig 51: Results Aggregate Source Definition**

**TARGET**

The group table has been generated in the Oracle database by using the source files as mentioned above.

|  |  |  |
| --- | --- | --- |
| **TARGET NAME** | **DESCRIPTION** | **TARGET FILE** |
| **CDW\_UAS\_F\_AGG\_DATA** | Contains results for all the exams happened across all colleges, with history |  |

**Table 26: Result Aggregate Target Description**



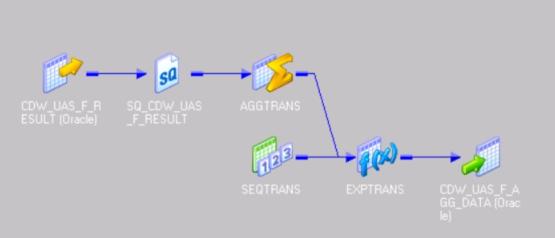
**Fig 52: Result Aggregate Target Definition**



**Fig 53: Result Aggregate Target**

**MAPPING**

The below figure shows the data flow from source to the target table by applying the necessary transformations to satisfy the business requirement.



**Fig 54: Result Aggregate Mapping**

**TRANSFORMATION**

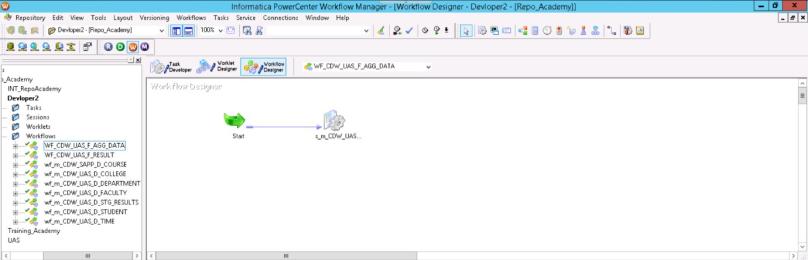
The descriptions of the transformations used are mentioned in the table below.

|  |  |  |
| --- | --- | --- |
| **TRANSFORMATI ON**  **NAME** | **DESCRIPTION** | **TRANSFORMATION LOGIC** |
| Source Qualifier | SQL Override Condition | SELECT CDW\_UAS\_F\_RESULT.RESULT\_F\_COLLEGE\_ CODE, CDW\_UAS\_F\_RESULT.RESULT\_F\_COLLEGE\_ NAME, CDW\_UAS\_F\_RESULT.RESULT\_F\_COURSE\_C ODE, CDW\_UAS\_F\_RESULT.RESULT\_F\_COURSE\_N AME, CDW\_UAS\_F\_RESULT.RESULT\_MARKS, CDW\_UAS\_F\_RESULT.RESULT\_GRADE  FROM |

|  |  |  |
| --- | --- | --- |
|  |  | CDW\_UAS\_F\_RESULT WHERE  EXTRACT(month from  CDW\_UAS\_F\_RESULT.RESULT\_F\_PERIOD\_KE Y )=7 |
| Aggregate | This transformation is used to count the total no. of pass and maximum marks and grouping by COLLEGE\_CO DE, COURSE\_COD  E. | COUNT(RESULT\_GRADE, RESULT\_GRADE='PASS')  MAX(RESULT\_MARKS)  Group by:  COLLEGE\_CODE, COURSE\_CODE |
| Expression | This transformation is used for inserting CREATED\_DA  TE as  SYSDATE. | CREATED\_DATE=SYSDATE |
| Sequence generator | This transformation is used to give surrogate key to  target table. | START VALUE=1  RESET option is selected. |

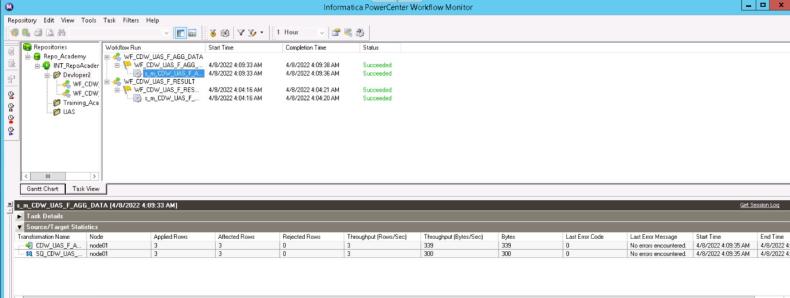
**Table 27: Transformation Details**

**WORKFLOW DETAILS**



**Fig 55: Workflow 9**

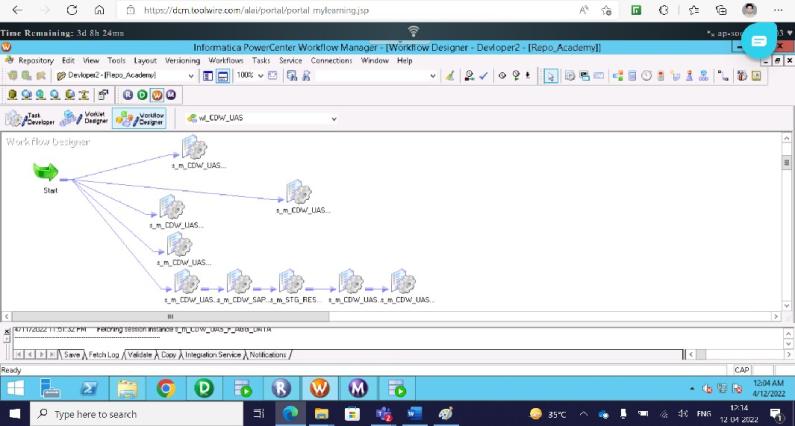
**TASK DETAILS**



The below figure shows that the session has executed successfully, and no rows were rejected. It also shows the execution

**Fig 56: Task Details**

## INTEGRATED WORKFLOW

**Fig 57: Integrated Workflow Details**

## INTEGRATED TASK DETAILS

## Graphical user interface, text Description automatically generated

**Fig 58: Integrated Task Details**

## CONCLUSION

The system **“UNIVERSITY ANALYTICS SYSTEM”** has been developed considering all the business requirements. The primary objective of the project has been satisfied by generating nine target databases by validating the constraints mentioned and the system has been operated only on the modules described in the source document and has included additional functionalities. The UAS is a data warehouse developed to manage the information of different Students enrolled in universities for the analysis of their results depending on various department and the faculty performances.