29/04/19

* Software Installation *

* Go to control pannel and uninstall java from your machine.
* Install jdk 7.
* unzip the software TOS-BD-20141207-1530-V5.6.1zip
* Install oracle.
  a) unzip the oracle XE112-win64.zip
  b) click on setup to install oracle.
* while installing oracle
  you have to set : pwd : manager
  : re enter the pwd : manager
* Go to comenand prompt and start typing. sql plus
  user id : system
  pwd : manager
* unzip the sql developer - 17.2.0.188.1159_x64.zip
  Go to unzipped floder and click on the sql developer icon.
  → Create a new connection by clicking on + button.
  user name : system
  pwd : manager
  host : local host
  port : 1521
  sid : xe.

— Note :-
first time when you open Talend make sure your system is connect to the Internet.

* goto unzipped floader of TOS-BD-20141204-1530-V5
click on 3 icons (.exe files)

first time when you open talend
→ give project name : AMEX
→ Select project click on open button
→ when you open talend make sure popups will come and you clicked on accept all and install option

* Install notepad ++.

30/07/19

Creating orcal db connection :- (db → Data base)

① → Go to repository
② → Select : Meta data
→ click on small arrow (→)
→ And choose Db connection.
→ Right click on select create connection (orcal_local host)
→ Give some name (~~oracle~~) * click on next button

Now choose Db type as oracle with service name
And fil the foorm.

login : System
Psud : manager
Service : local host
port : 1521
Service name : xe.

→ click on check button you should see connection

as successfull. click on finish button.

→ you can see the connection which you created in Repository

↳ Media data

↳ Db connection

↳ connection name (oracle-local-hourt)

Scaning the ~~me~~

**Scaning the Metadata of Delimited of CSV file (comma separated values file):

→ Go to reporting * metadata

* ~~Rig~~ file delimited

↳ right click.

→ any say create file delimited.

→ Give some Name mostly file Name.

→ Click on Next Button.

→ Browse for file using Browse Button

By default it shows only csv files.

→ you can change the filter from.

.csv to *.* and choose the file.

→ Click on Next Button and fill the form.

→ By choosing proper field separator.

→ Proper row separator.

→ If the file has heading choose set ☑ heading row has column names option.

* click on refresh Review.

you should see the data in tabular format

→ Click on Next Button the Talend guess the Metadata as per the source data.

→ you can change the metadata as per the Business * (Business aspect).

Click on finish Button).

→ file with pipe delimited and No header:-

→ follow the same as while choosing Separated choose "custom ANSI".

→ and change the delimited as per the source.

→ If the file as no header, make sure you are not selected set heading row as Column Names option.

31-07-19

** loading Data from file to Table:-

→ Go to repository ~~repository~~ Job design.

→ right click and say create job.

→ Give some name like job-file-to-table

→ Click on finish Button.

→ Now you can see talend.

↳ Go to metadata

↳ delimited section.

→ Select the file which we can scan earlier (metadata) drag and drop to the job.

And select T file input Delimited.

→ Drag and drop the oracle connection from metadata to the job.

→ Select T-oracal output.

→ Select the input icon
↳ right click
↳ select row, select main.
↳ And click on the o/p component.

→ Now Double click on the T-oracle o/p
give the table name Double coats.

→ And select action on table as create table if doesn't exist.

→ Run the job using RunTab (or) F6.

**Retriving table Data :-

→ Go to Reporitory
↳ Select oracal connection
↳ right click
↳ And select Retrive schema.
↳ click on Next Button.
↳ Select the uses from the list it.

→ Click on small arrow beside the system.

and select the required tables by selecting check box.

→ Click on Next Button, click on finish.

→ you can see the table metadata in the oracle connection.

→ By clicking on small arrow.

* ~~Table to file~~

** Table to file:-

→ Drag and drop the table & metadata from the scanned tables to the job.

→ on the select t-orcal input

→ Now go to pallet and search for o/p delimited

→ Drag and drop the component to the job.

→ click on the designer and start typing delimited.

→ And choose T-file o/p delimited.

Connect your i/p throw the o/p using row dobule click on T-file o/p delimited.

→ Choose the file path where the file need to be created by using Browse Button.

→ Run the job.

→ By default talend create the file, without header, you can choose include header option and you can choose the delimiters for row & column.

→ As per the requirement.

→ for each and every run, talend will over rides the target file.

→ if you want to append the data to the Bottom of the file, choose append option in the o/p delimited.

01/08/19

* Sql query:-

→ Select,
→ empno,
→ Sal,
→ dept name,
→ loc,
→ From system Emp - 08072019 where deptno "10 order by sal desc"

if you want remove duplicate.

query : * select distinct
empno
Sal
empname ....

from system. Emp - 08072019 where deptno = 10 order by sal desc"

** T-oracle input :-

By default T-oracle i/p comes with select query in " dobule coats.

You can change the query as per the requirement like adding where class.

order by classes

distinct clause.....

→ When you modify the query, make sure you hitted the guers schema.

** Applying filter on t-file input delimited :-

→ Take file as input

→ connect into t-filter row component

→ Dauble click on it.

Click on + Button to add a column.

Input colum	function	operator	Value
dept no	Empty	Equals	10
Sal	Empty	Greater (or) equal to	15000

** T-Sort Row :-

→ Sort the records based on some key column either ascending (or) descending.

→ we go with t Sort Row.

→ Click on + Button to add a column. ← → options in the t-sort Row

→ Select type

Schema column	sort num or alpha	order asc (or) des.
Sal	Num	asc

+

** Removing duplicates :-

→ we go with t uniq row.

→ options wit in the t- uniq row.

→ select all the columns as ☑ key attribute.

Column	Key attribute	Case sensitive
empno	☑	☐
empname	☑	☐
Sal	☑	☐
HRA	☑	☐
dept No	☑	☐
dept Name	☑	☐
loc	☑	☐

05/08/19

** t- filter row → t sort row

↳ t file output delimited - 1

t unique → (rejected, unique) → oracle

↳ duplicates → t- file output delimited - 1

05/08/19

** t- filter columns

→ To remove unnecessary column and select only requirmen required columns from the flow we can go with t-filter column.

→ Go to edit schema & copy only required colums from left to right.

** t- replicated :-

→ To copy the same data into multiple o/p go with t-replicate.

→ No options avaliable t-replicate.

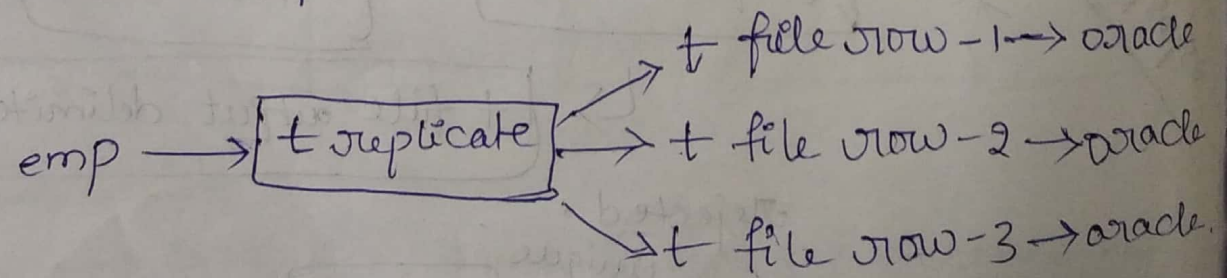
→ you can take n-number of mains from t-replicate

** t- Sample row :-

→ you can specify the rows.

example : 1..100 (1 to 100)
101..200 (100 to 200)
1,5,7 (1 and 5 and 7)

** Multiple → t Replicate



06/08/19

Header, footer and limit :-

**Header :- → In i/p component if you specify the header as "n".

→ it skips first 'n' no. of records from the file while reading.

**footer :- it skips "n" no. of records from the Bottom of the file.

*limit : once after start the reading file it stops at nth line.

**t-unite :- → works like union all operation in database.

→ if you have same metadata flow, if you want to combined them as a single o/p, you can go with t-unite.

options in the t-unite :

Click on sync columns.

t-unite will not remove duplicates.

to get union functionality use t-unique row along with t-unite.

*** Multitread execution:-

↳ To run the sub jobs in the parallel we need enable multitread execution option × cvaliable at job tab.

** Prejob and Postjob:-

↳ Some subjobs you want run at first ~~you~~ no Can go with "pre job" on compoed ok.

↳ if you want to Run ~~if you Some subjob~~ at last: "post job" on component" ok.

07/08/19

** t-file list:

↳ floder Containing multiple files with same at meta data need to load into a single target.

Metadata

~~The conct count of the files and Keep an~~

Changes → The count of files in the i/p floader will keep on changes.

Step:-1

↳ Scan the metadata of single ~~floorer the file~~ a floder.

↳ Drag and drop job (Make it as i/p)

Step:-2

→ connect it to target.

↳ Now take the component called t-file list

↳ Double click on ~~the table~~ it.

↳ and Choose the directory ~~for~~ the ~~files~~. from where you to pick the files.

Click on "(+)" Button to a file.

example: "*.csv"
"*.txt"
"*.*"
emp*.csv.

↳ Connect t file list to t file i/p delimited using row iterate link.

↳ Double click on i/p delimited component.

Change the property type from: Repository to [Built-in]

↳ Remove the file location use cntrl and space Button x to the choose global ~~variable~~. variable. called

→ t-file list current file path.

↳ Run the job.

↳ if you are using file as an o/p, make sure you select "append" option.

↳ if you target is table don't choose ~~ta~~ options drop and truncate.

** Reading files from subdirectories:-

↳ We an option called include sub directories in the t-file list component.

** improving the performance:-

↳ Select the iterate link

↳ Right click

↳ Go to setting

☑ ↳ Select enable parallel option. and give some Number (4/5).

08/08/19

**Normalize:-

i/p:-

empno, ename, Skills

100, Mahesh, informatica; informatica; data stage; oracle

104, mani, informa; tale; datart; ora

101, kalyan, informatica; C; C++.

o/p:-

100	Mahesh	informatica
100	Mahesh	informatica
100	Mahesh	data stage
100	Mahesh	oracle.
104	Mani	Mani.

Step 1:-

↳ Scan the metadata of the file.

↳ Drag and drop the job. → input

↳ connect it to t-normalize.

↳ Double click on

↳ Select the normalize column.

Ex:- Skills.

↳ And Select the delimited column.

Ex:

↳ connect into target.

↳ By default remove duplicates if you want Remove duplicate use t-unique row after Normalize.

↳ Else use an option called get Rid of duplicate rows from o/p, in t-noormalized component.

**Denormalize:-

Input:- empno, ename, Skills

100, Mahesh, informatica

100, Mahesh, talend

100, Mahesh, oracle.

101, Kalyan, A

101, Kalyan, B

101, Kalyan, C

101, Kalyan, D

output:-

empno, ename, skills

100 Mahesh, informatic; talend; oracle

101 Kalyan A; B; C; B

↳ Scan the metadata file make it as i/p.
↳ Connect it to denormalize.
↳ Double click on denormalize.
↳ Go to edit Schema.
↳ copy the columns from left to right.
↳ using Double arrow.
↳ And increase the denormalize length.
↳ Click on 'OK'.
↳ Click on (+) Button.
↳ select the column to do denormalize.

Ex:- Skills.

↳ select the delimited which you want in target.

Ex:- (;)

↳ Run the job.
↳ By default if you want remove use t-uniq Row.
↳ Before the denormalize (or) use an option called ☑ merge same value in t-denormalize.

09/08/19

** Buffer :- Buffer is a component will help you to keep the data in Memory for some memory purpose and reuse it Multiple times in the same job using Buffer. - input and Buffer output.

** Hash :- Hash is a Memory component which keeps the data in the memory for some memory purpose and reuse it Multiple times. using hash i can store different Metadata & into the Memory. data

** Scenario :- Folder containing Multiple files with same Metadata need to load in a Single target using sorting operation.

13/8/19

** tMap :-

→ Add output table option (Right hand side - top most Corner (+) Button).

→ Drag and drop columns from left

→ Remove unwanted Columns from right hand - side using edit schema.

→ Remove unwanted output tables using (-) Button on the right hand side topmost Corner.

→ Activate, unactivate expression filter.

Ex :- dept NO == 10.

and & &.

* Filter condition in the t-Map on string Column :-

row1.empname.equals ("mahesh")

Addition operation +

row.1.empname + row1.dept Full name

row.1.sal + row.1.HRA Full Salary.

for space :- Row.1.empname + " " + row.1.deptname

⟶ Full Name.

to change date Column ⟶ ctrl space.

⇒ Adding extra Columns in the target.

→ Column filter.

→ (Header) Row filter.

→ (4 ff) Multiple filter condition.

[By using expression Builder]

→ Common condition

(left side added)

[ex: salary - - -]

row.1.sal > 20k.

→ filter condition in string Column.

→ Added extra Columns [Ex: full Name, full sal]

if we want space row.1.emp.name + " " + row.1.dept.

→ data

talend date

get current date.

"dd-MM-YYYY". [ctal space] to change date.

Ex :- EEE dd-MMM-YYYY HH:mm - - -

14/8/19

** if condition :-

Ex:- if we want more than 15,000 sal - Grade A
- Grade B

Add a column. → Grade

row.1.sal >= 15,000 ? "Grade A" : 'Grade B'.

condition ? true : false.

* Multiple if condition:

row.1 sal < 15,000 && row1.sal >= 5000 ? "Grade B" : "Grade C".

row 1 sal < 15,000 && row1.sal >= 5000 ? "Grade B" : "Gra

Emp. name:

All need in upper case function

String handling only using on strings.

String handling → upper case.

** Alpha: Checks whether data is sorted in Alphabet (or) not.

[True/false → Boolean]

* BTRIM :- Remove extra space. from the Back side

* CHANGE: to replace value. string.

("row.1.emp.name, ".", "_".)

↓

(. replace by _)

to remove replace with empty value. " "

COUNT:- Count the no. of occurances. → repeated values (integer.)

EREPLACE:- Replace value

FTRIM:- Remove extra space from front side.

INDEX:- Position. (Count start from 0). integer.

(* the hellow world 1. "," hellow") → u:

it counts with "0".

if doesn't found you will get "-1".

1st occurance

IS-AIPHA:- True/False (Boolean)

To check value is in alphabetics or not.

(order doesn't Matter).

* LEFT:- How many characters we want from left side.

→ (" hello world 1.7")

* LENGTH:- Mobno. integer.

* RIGHT:- How many char. from right side

["hello world 1.8"]

* SPACE:-

row 1. dept name.

row 1. emp name + string

* STR:- (" ", 6)

* TRIM:- front and last

* Upper Case:

SPACE:

row1.deptnam

row1.empname + String

STR: (" ", 6)

front & last.

TRIM: front and last

upper case:

upper case:

22/08/19

Scenario 1:-

Mobile Numbers Reddy's

string Handling.LEN(row1.mobilenum) = 12 ? "Yes":

Scenario:-2 mail id - count.

Scenario :-3

22/08/19

Scenario

* Add extra column in target call ename_vaild = flag.

Column Column

* ename contains only Alphabits say "true" Else "~~false~~" "No".

* .StringHandling.LEFT(row1.mobilenum, 2).equals("

*

* Scenario:- 1 22/08/19

① Mobile Numbers.

String handling. LEN (row1. mobilenum) == 12 ? "Yes" : "No"

* Scenario 2 :- { Tlog Row (Table View)

Mail. id - count

* Scenario 3:- Add an extra column in the target called employee Name - valid - flag.

→ if the employee Name contains only Alphabets. Say "true" Else "NO".

↳ Mobile Nums.

* String handling. LEFT (row1. Mobile num. 2). equals ("91").

→ Need '4' output files based on the Mobile Number country code.

→ if NO. starts with '91' it should go to India. txt.

→ if it starts with '01' it should go us.txt

04 → UK

02 → Canada.

* Add an extra column in the target called "country" if Mobile NO. starts with '91' it should be india.

01 → USA

02 → Canada

04 → UK

* valid email id's in one output.
* invalid email id's into another o/p.
* valid Mobile no.s in one o/p.
* Invalid Mobile no.s in another o/p.
* valid emp.Names → one output.
* Invalid emp. Names → another o/p.

(Expression)

→ Need 2 extra columns in the target.
→ called Mail Name and domain Name.
→ So Split the Mail id into 2 parts:
Before @ → one output → Mail-name
after @ → another output → domain-name

aabaa@gmail.com|;aabaa;|@gmail.com

*Padding: [Expression]

→ In my target the length of the emp.no.is fixed to "5".
→ If the i/p column length is <5, then pad "0" to the employee Number to the left.
→ In the same way. employee name length fixed to 15.
→ if the length is Not 15, add space as Padding Character.

5-stringHandling.LEN(rowl.eno).

String Handling.STR('0',5-stringHandling.LEN(rowl.eno)+rowl.eno)

* name space [in the place of 0]

* StringHandling.STR(" ", 15 - StringHandling.LEN

23/08/19

Date file:-

Compare date function:

Proper → <=
improper >
used in 2nd func

* TalendDate.compareDate(row1.Shipment_Start_date, row1.Shipment - end - date) <= 0. Vaild
>0 / == 1 in vaild.

Difference date:-

String → long

extra Column } No. of days

Add date function:- some thing to date/month/year.

String → Integer
Days → Integer
only 10 digits
Long.

compare Date:- Compare 2 dates and gives you which one is bigger and which on is smaller by giving 0, -1, +1.

Diff date:- Difference b/w 2 dates 0.

formate Date:- Converts from dates to string.

Get current date [in string]:- get date.

Get first day of Month: first date.

Get last day of Month: last date.

Get part of date:

TalendDate.getPartOfDate("WEEK_OF_YEAR",) TalendDate,

get

26/08/19

Get part of Date:-

Get Random Date:

imp ** Is Date:-

Scenario:-

Date pattern:

date-id	date-pattern	month	days of year	days of month
1/1/2010				
2/2/2010				
3/3/2010				
1/4/2010				
1/5/2010				

week of month	week of year	dayname	holiday	Bank holiday.

→ java.lang.NullPointerException.

When we do operations on a Null Column Value. Null pointer exception

How to handle Null:

→ Relational.ISNULL (row1.Sal) ? 100 : row1.Sal+100

→ Relational.ISNULL (row1.HRA) ? Null : row1.HRA-50.

Numaric:

Numaric → Sequence } S1

Mathematical:

↳ ABS [Always returns positive value].

MOD:- [Always returns reminder value].

*] Scenario 2:-

Take 10 records file.

1, 3, 5, 7, 9 → 1 file.

2, 4, 6, 8, 10 → 1 file.

Key: Seq+Mod

Scenario 3:-

Take 10 records file.

I/P →	O/P:-
100	110
108	109
107	102
104	103
105	106
106	105
103	104
102	107
109	108
110	100

Scenario 4:-

first 4 records } any
last 4 records } file

Joins :-

↳ 5 types.

↳ * Left / left outer join
* Right outer join
* Inner / Equai Join
* Full outer join
* Cross (x) join.

* Left outer join :- 

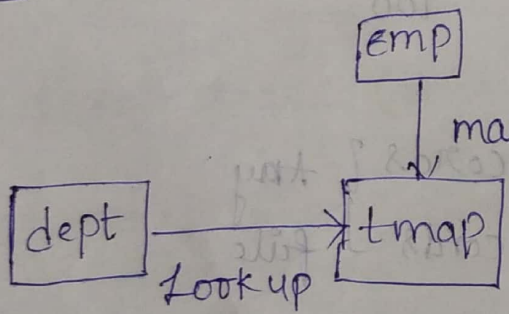
→ All the Records from left table will go to target.
→ if there is a match, you will be getting a matched records.
→ if there are multiple matches you will get multiple Records.
→ If there is no match, still the record go to target with joint columns target as Null.

Emp (L)

eno	ename	Sal	HRA	DNO
100	A	1000	100	10
101	B	1800	180	20
102	C	1600	160	10
103	D	1500	150	40
104	E	1400	140	80
105	F	1200	120	90

Dept (R)

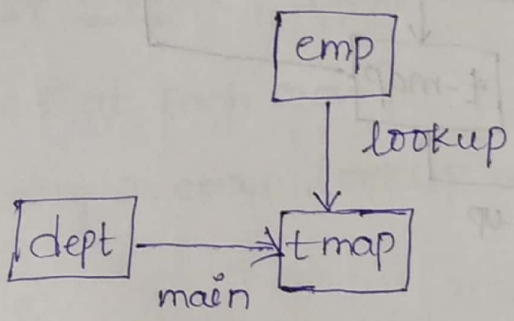
D.Dno	Dname	LOC
10	HRA	Hyd
20	MKt	che
40	Fin	Blr
20	Lab	Pune
20	Prod	Del
60	R&D	Noida



Unique match – All matchs.

* Right / Right outer join:

→ All Records from right table will go to target.
→ if there is a match, you will get matched records.
→ if there are multiple matches, you will get multiple records.
→ if there is no matches still the reverse will go to target with joint column value as well.

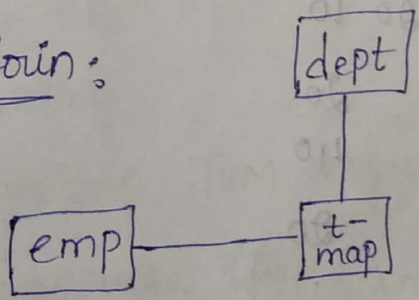


Inner / Equi join:

→ Only matched Records from Both the table.
→ if there is a match, matched - Record
→ if there is multiple matches multiple Records.
→ if there is no match, we won't consider in the target.

left [inner]
unique [All matches.]

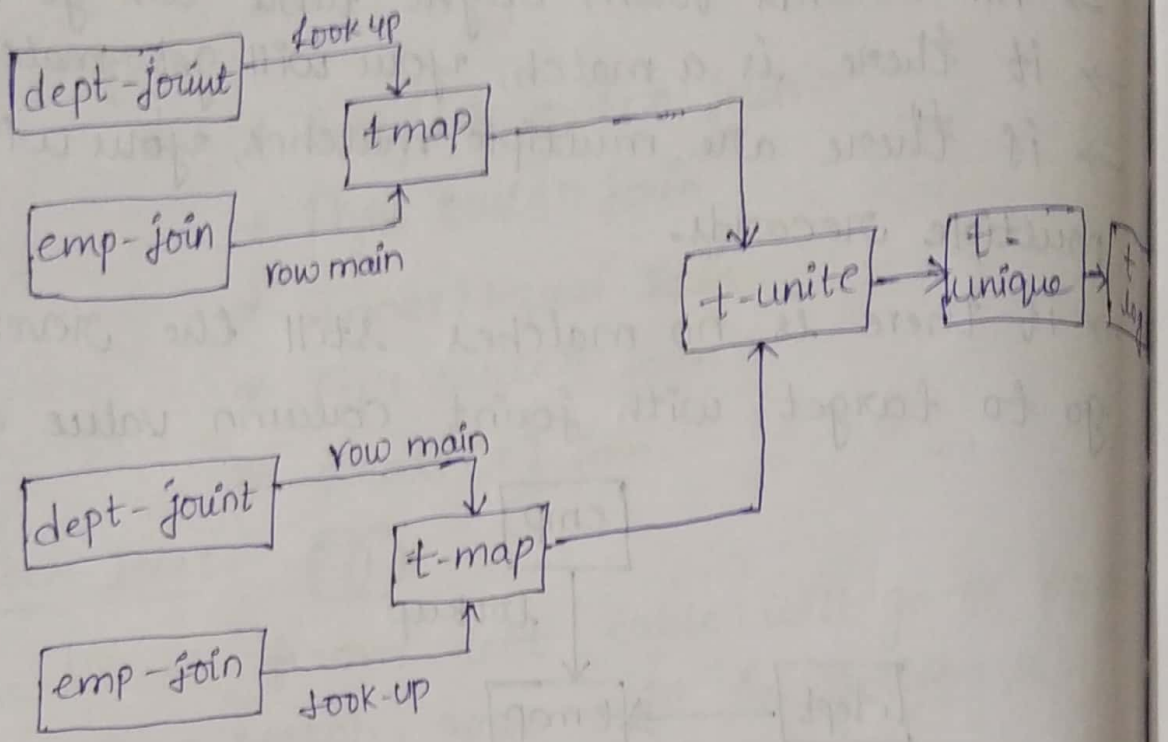
Full outer join:



1st left outer
2nd Right outer

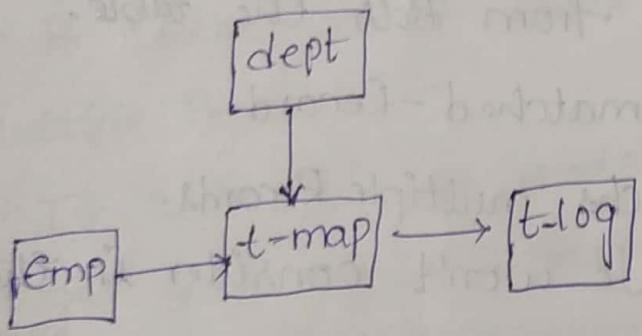
Next
tunite
tunique.

** Full outer join:



** cross-join :-

* No key column
* Copy all i/p records to o/p.



100	A	1000	100	10	10
100					20
					40
					20
101	B	1800	180	20	10
101					20
					40
					20

Match model :- first match

↓

Only first

unique match → last record

Default : left outer joint } latest record will taken.
: unique match }

Look up model: [load once]

[reload at each row]

for every record lookup will load again and again.

** Reload at each row (cache):-

** Keep memory issue / Memory out of Bounds exception:

Advance settings.

☑ use specific JVM arguments.

if we want 5GB -XMS 5120M max
-XMs 3072M min.

** Steps :

→ Go to Run tab.

→ Advance setting.

→ ☑ use specific JVM arguments.

And increase the { Xms (min)
Xmx (max)

⇒ Store temp data [true]

* Click on tmap component
temp data [temp folder]

→ then it stores data in hard disk.
[BIN file].

** Inner join records:

→ if we want to join inner join reject records

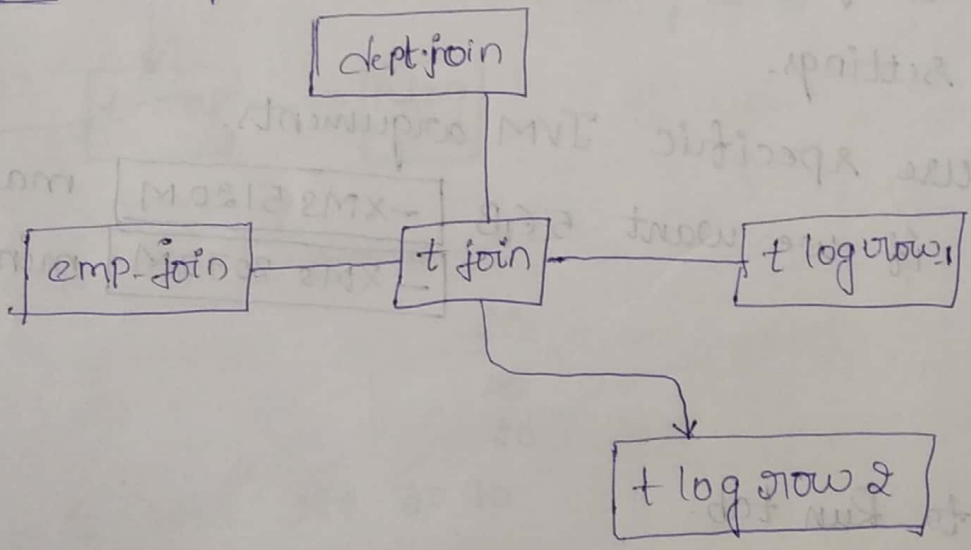
→ catch look up inner join rejects [true].

→

→ Catch o/p records [true] → reject records on out put.

29/08/19

* t-joint component:-



→ Allows only two sources get join.

→ By defalut left outer join unique match.

→ By subtiling right outer join unique match.

→ by selecting option called with reject o/p.

→ I can get inner join with unique

and rejected Records the Main flow.

**Steps:-

→ connect your sources to the t-Join

→ Double click on t-join.

→ Go to Schema

→ copy columns btt to Right from both main and look up.

→ Click on "Ok".

→ Select on option called include look up columns in o/p.

→ Click on "+" Button to add look up columns and o/p columns.

→ come to key detination section.

→ Click on "+" Button to add a key detination

Scenario:-

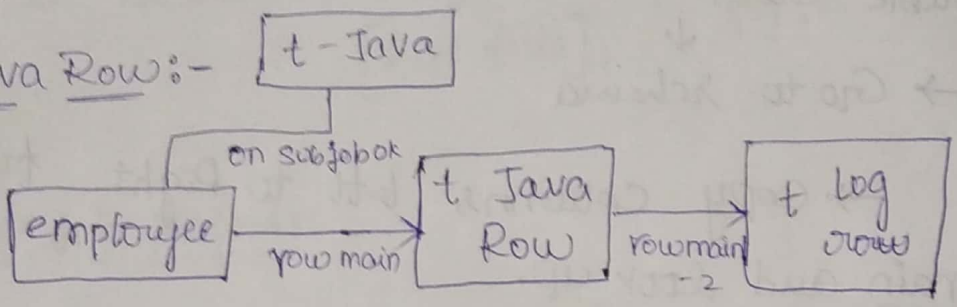
① Take Some file need first 100, next 200 next 300 record target using [t-map]

② using t-Join need achive first match

t-Java:-

→ System.out.println("The job "+jobName+" started ... and project name "+projectName);

t-Java Row:-



→ Double click
↓
→ Edit Schema

→ Here we have chance to add extra columns

→ click on generator code.

→ Automatic Script was written.

t- Java:-

* t- Java is the component where you can write the jave code

* which will get excute directly on the JVM. (java vir

* This component will not have edit schema.

* So we can't connect to the follow.

t- Java Row:-

* t- Java Row is the component. will have edit schema.

* By you can add extra columns on the edit schema and we can write expressions in the o/p.

* Same like t-map.

t- Java flex:

* we have start code, main code and end code.

* The code which we are writting in the start code first.

* The code which we are writting in the main Code will ~~executed~~ execute for each and every row.

* The code which we are writting in the end code execute at the last.

* To reset the Numaric sequence we use the t-java component.

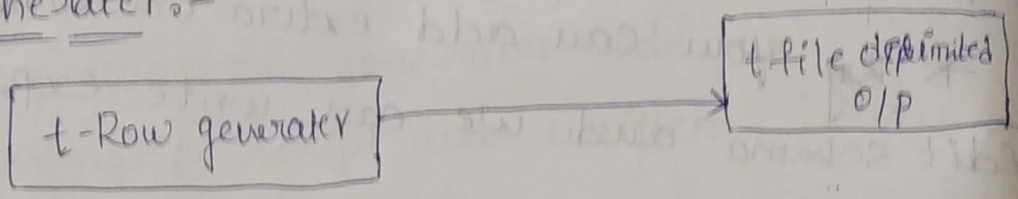
Numaric.sequence (S1, 1, 1);

30/08/2019

* Advanced mode t-filter row:-

input_row.deptno == 10 && input_row.sal > 15000 ||
input_row.deptno == 20 && input_row.sal > 15000.

* t-Row generater:-



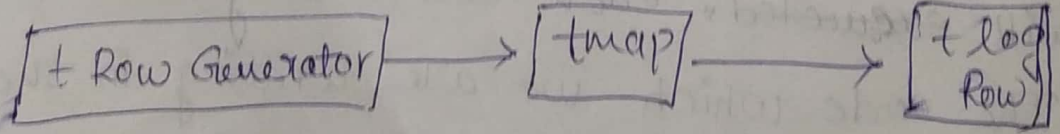
→ Double Click on it.

→ Give the number of rows you want generate

→ Click on + Button add required columns an corresponing data and corresponding function.

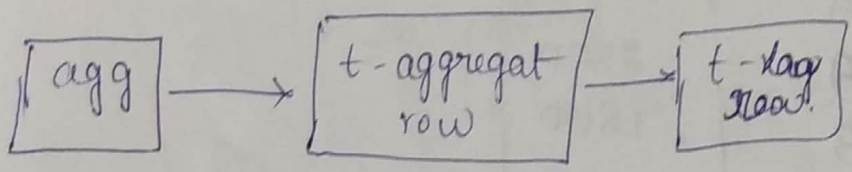
→ connect to target.

→ Run the job.



→ To push the Record from t-map ~~you can~~ we can use t-Row.

t-aggregate row:-



* Go to Advanced setting Change the Column. [;]

Scenario 1:

① Need remi column in the list column.

Scenario 2:-

i/p:-

Col_1,	Col_2,	Col_3,	Col_4,	Col_5
1,	x,	x,	x,	x
x,	1,	x,	x,	x
x,	x,	1,	x,	x
x,	x,	x,	1,	x
x	x	x	x,	1
o/p:-	1	1	1	1	x

Scenario 3:-

e.no	ename	Skill
100	mahesh	informatica
100	mahesh	Talend
100	mahesh	oracle
101	mani	Talend
101	mani	Java
101	mani	MS SQL
101	mani	unix

o/p:-

e.no	ena	skills
100	mahesh	i,t,o
100	mani	Ta,Ja,m unix

I/P:

e.no	ename	Sal
100	A	1000
100	A	2500
101	B	1800
101	B	1300
100	A	6000
101	B	1400

O/P:

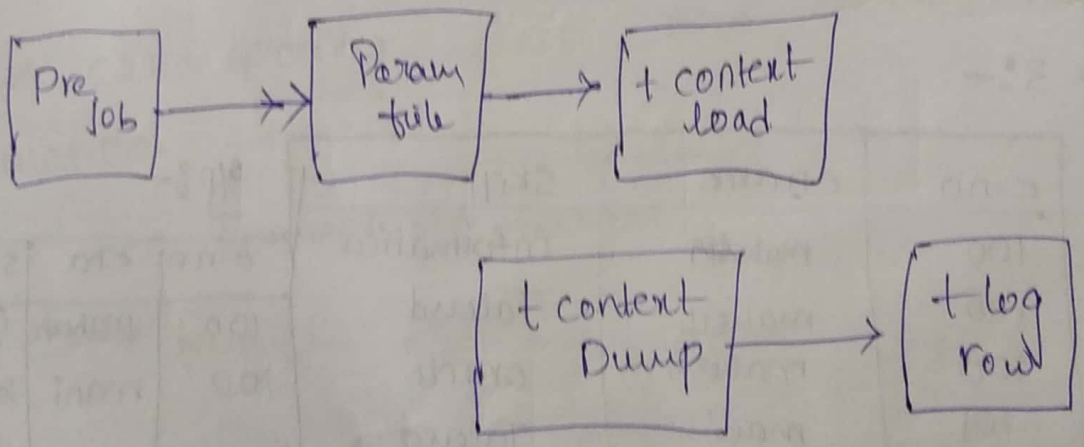
eno	ename	Sal
100	A	6000
101	B	1800

03/09/19

t- Context Load:-

* UAT → User acceptance testing
* QA → Testing or Quality Analysis

04/09/19



Project Setting

↳ Job setting

↳ implicit context load.

* Implicit context load:-

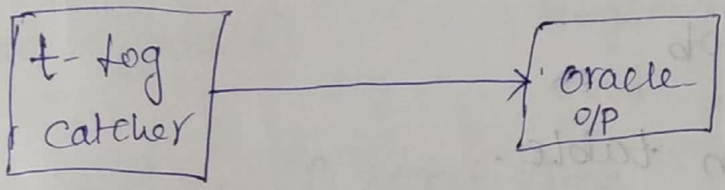
↳ from file

↓

Browse

05/09/19

t-log catcher:- * log catcher shows where errors occured.



t-start catcher: [start & end job status]

Column:-

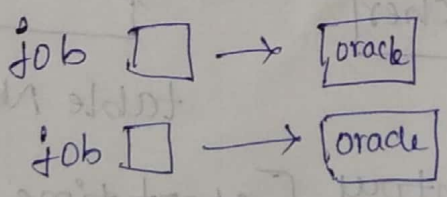
~~moment~~

~~pid~~

~~father pid~~

job:-log-start

2 jobs:- Run

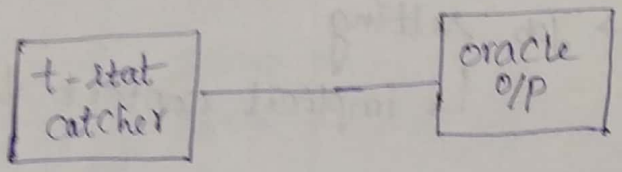


if user name / file name wrong

☑ die on error.

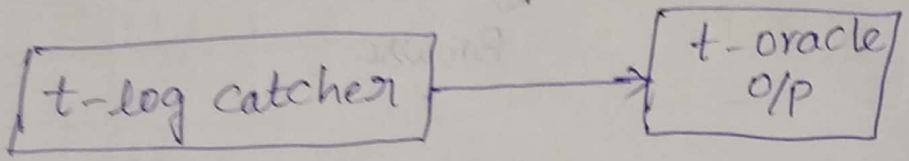
Job failed

i/p: t-stat catcher:- captures the execution_log.



job Run status:
job Duration

* Where error exactly occured: "Exception"



o/p

* captured on table.
* Give table name
* Run job

t-

Select * from table.

* it shows Errors.

log-catcher:- captured the job failures, with moment when, reasons.

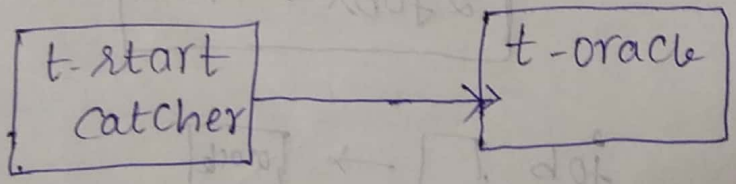


table Name.

Begin entry
End entry
[start time Job _ End time Job]

if local host given as wrong!

Audit info: To Capture audit information.

→ t-log catcher: Error log.

* t-stat catcher: Begin, End → failure

process id. [job execution log]

job Run status: success/fail.

job Duration.

Bottle Neck: which component took time to run the job. Each component → Adv. setting.

☑ t start catcher statistics.

origin [Null] Begin / End

Component - { start time / end time

Entire project: [project level setting]:

file :-

→ edit project prop

stats and logs

☑ use stat

☑ use logs

☑ on data base

stats table.

logs table.

I/P:

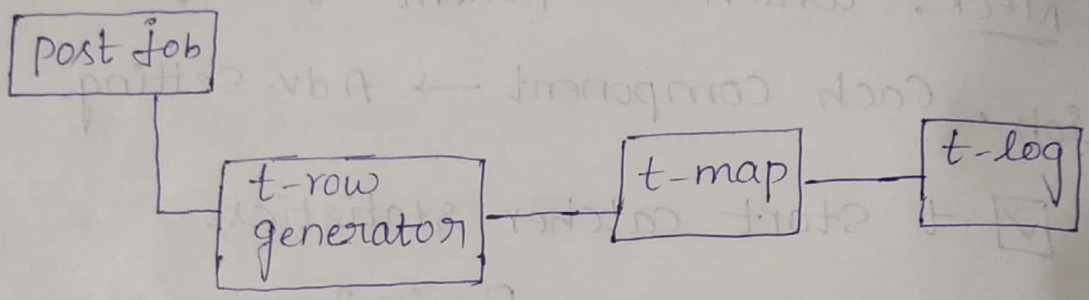
Each job:- [job level setting]:-

stats & logs: ☑ stats table
logs table.

☑ t start catcher statistics.

Audit into count:

O/P



* Add Columns:-

job name
pro name
Pid
emp-file_count
tgt 1_count [table name]
date file_count

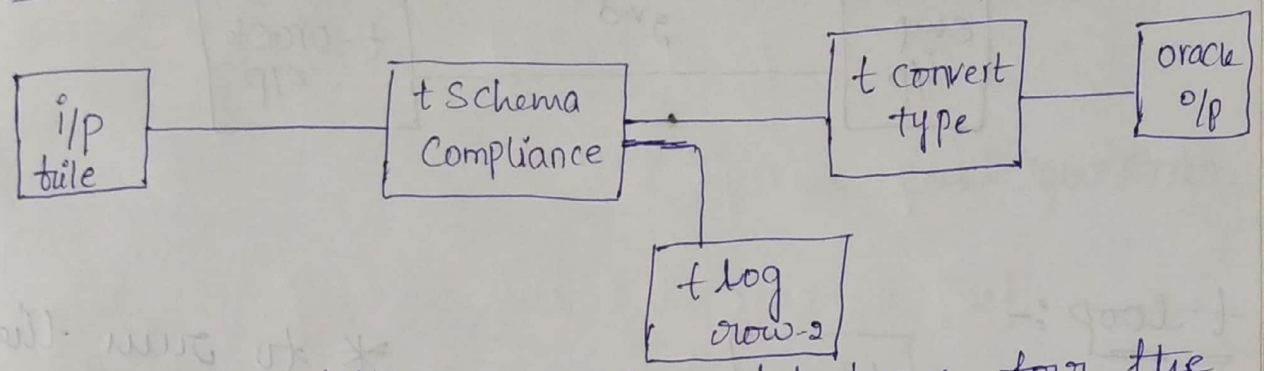
NB - line global Variable :-

↓

Gives you the number of records processed by particular component.

06/09/19

t- Schema compliance :-



* Read the i/p file as string datatypes for the column.
* Connect to schema Compliance.
* Options in the schema check. ~~custo~~ custome defined compliance.
and defined your rules.

Example : data type
date pattern
Nullable
max length.

* Schema complience
* convert type
* Generate DOC - HTML
* Built
* memory issue
* Scheduling

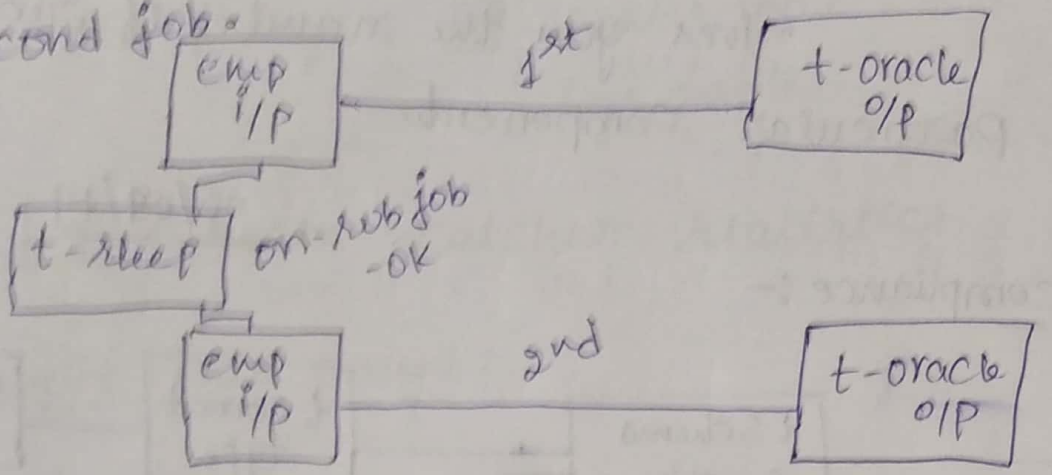
Repository vs Built-in :-

↳ centralised (Best)

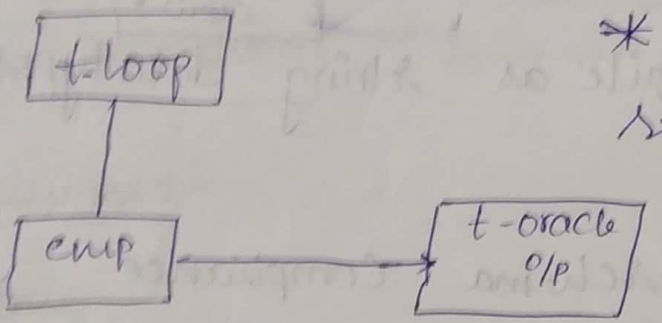
↓ local

09/09/19

t-sleep :- It gives some gap to run second job.



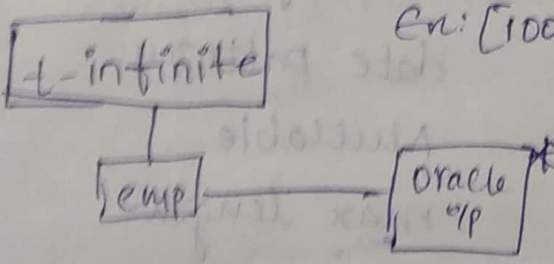
t-loop :-



* to run the job so many time.
from (start) to (max)

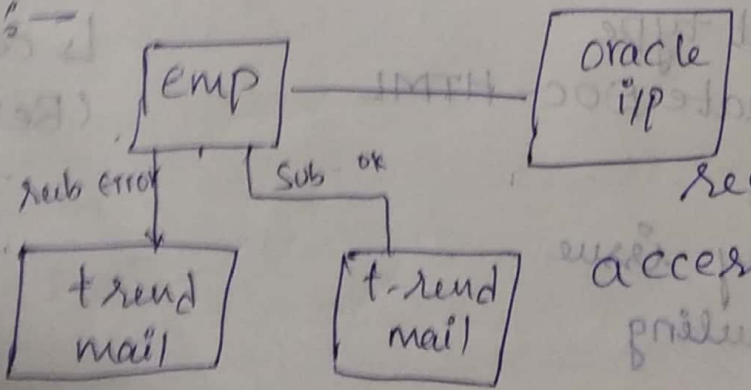
t-infil

t- infinite loop :-



* wait at each iteration
eg: [1000] (in millisecond)
* so may times run.

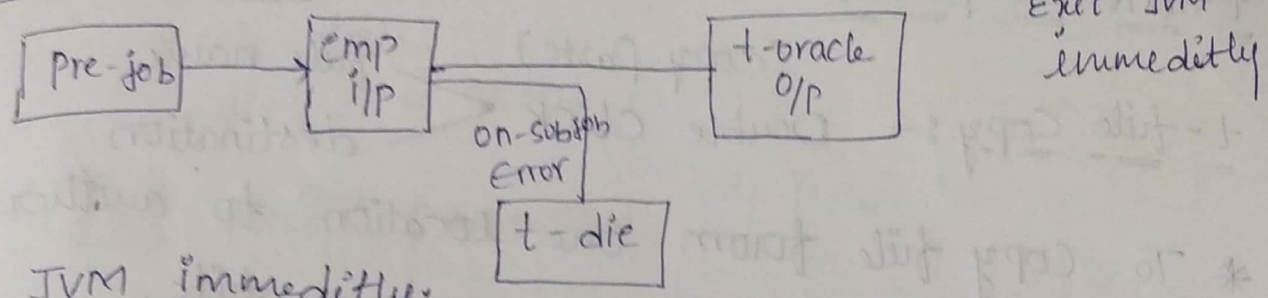
t- Send mail :-



* Allow le
secure apps to
access your gma
[or]
*

sub:

t-die:-



Exit JVM immeditly:-

* to kill the job any time on some trigger.
* like on-sub-Job-ok (or) on-sub-Job-Error.
* we go with t-die.

t-System:-

* to exquete unix, windows (or) das commands to talend.

* How to exquete batch file from command prompt?

t-SSH: [secure shall] → not in a local machine.
→ exquetes unix/das commaud remote box.

t-Run job:-

* Create job - workflow.
* drag all job. what you want to run.

10/09/19

t-file Components :-

t-file Copy :- (copy Paste) Double Click < file name, destination

* To copy file from one location to another.
* If we Enable ☑ remove source file.
[cut paste].

* Copy only single file

☑ Rename ⟹ Name change

☑ copy directory ⟶ full directory / filode mo

t-file delete :-

To delete file.

single [folder/file].

fail on error. [empty file].txt error

delete file / folder.

t-file Archive :-

Dir - file → copy

Archive here. (Rename)

format : Zip.

Compress level : best

if enable ☑ encrypt files [password]

we have a chance to password.

t-file unarchive:

Arch: file
dir: - ↙ copy

☑ Need password.

t-file properties: [contains edit schema]

file: —
it give properties.

[t]-file → [t]-log row

t-file touch:

create empty file.
[0 bytes]. [unix-touch
create empty file]

file name touch.csv
file name touch.txt
⟶ .xl

Scenario:-
folder contain multiple files.

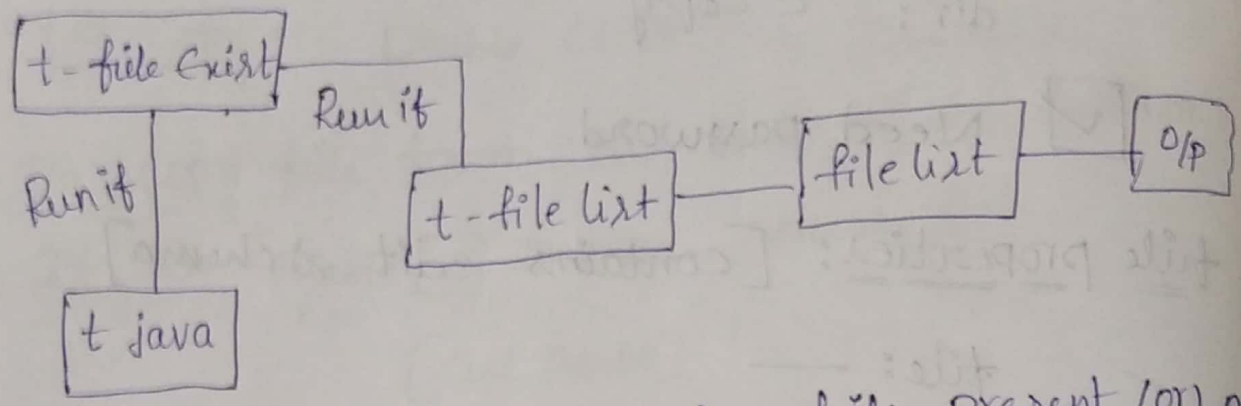
2 log files:

log file 1:
total files total size

log file 2.
file names time stamp Asc order.

11/10/19

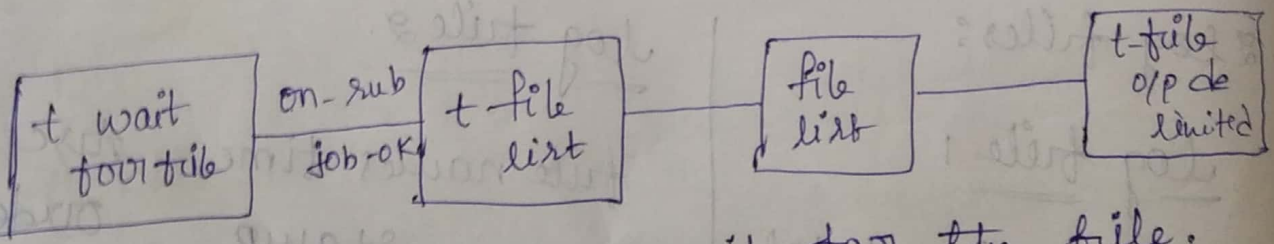
t-file Exist :- — Run if.



* it will check wether the file present (or) not.
if present there other wise false.
* if we file exit component we will use "Run if" condition.
* Run if → right click give condition.
(it is a boolean type).

t-wait for file :- file mask

Choose Include present file ☑
Exit loop → Then



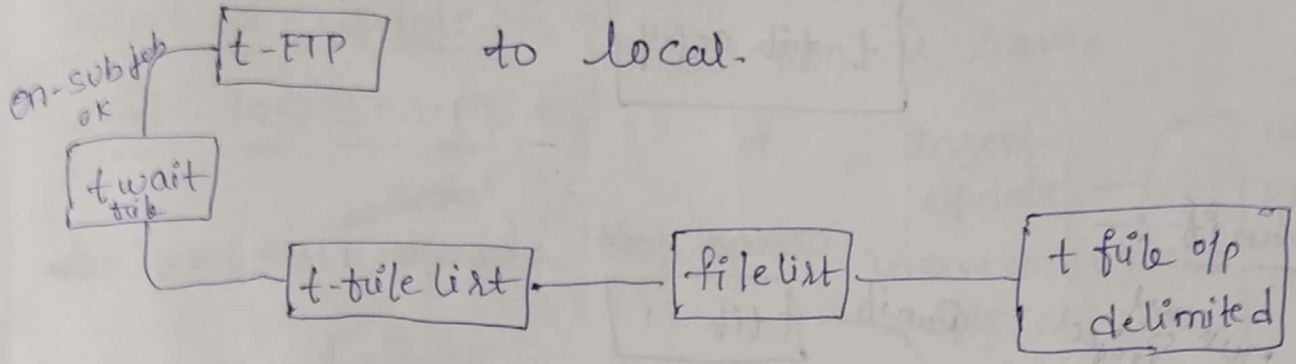
* it will keep on wait for the file.
when the done file appends it will run the job.
* In the t-wait for file :- we need to enable "include present".

winscp : (windows secure copy):

* To copy the file from unix box to windows.

we go with "winscp.tool" (or) file "Filo".

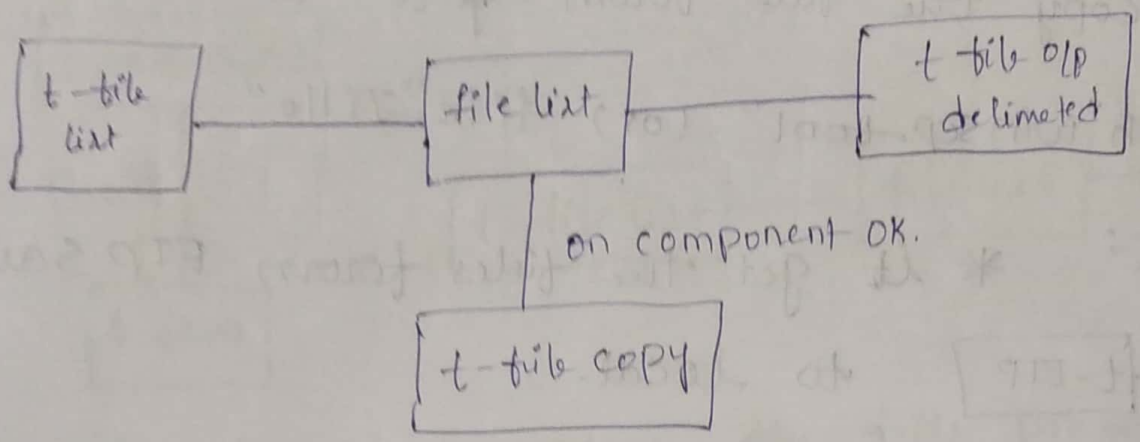
t-FTP get: * it get the files from FTP saver to local.



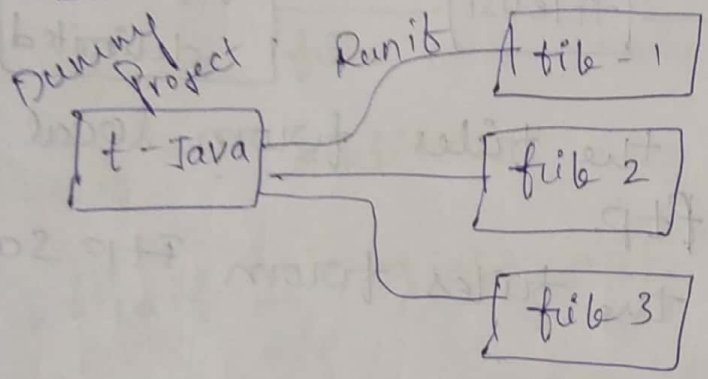
t-FTP PUT :- It puts the files from local sava to ftp.

t-FTP Delete : To delete the files from Ftp sava

On Subjob OK :-



Run if :-



Use Run if >3 (or) <0.

Condition: context.jobid = 1
context.jobid = 2
context.jobid = 3

13/09/19

** t oracle input :

Improving performance t-oracle input :-

* Go to Advance setting.
☑ * and select use cursor option.

* parallel hint in the oracle input.

~~Select /*+parallel(5)~~ / Query: Select /*+ parallel(5)

* Check wheather index created on Key Column (or) Group by Column. Else if not create them.

t- oracle output :-

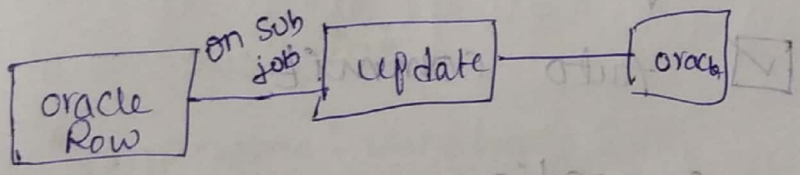
Insert (or) update :- } function is same
update (or) Insert :- }

~~for update x schema key option~~

Insert
update
Insert (or) update
update (or) Insert
Delete
use Key Column

→ Go to Advanced setting
☑ increase the commit size and Batch size.

17/09/19 t- oracle Row :- * it is used to truncate. (or) delete (or) update by excute query : "truncate"



to excute some statement directly on a database.

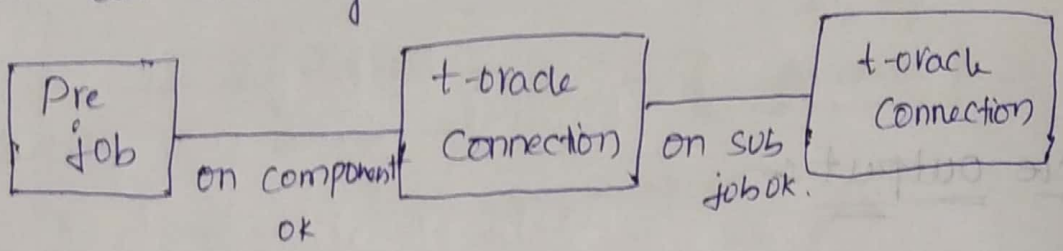
Multiple jobs : To Run by only one connection.

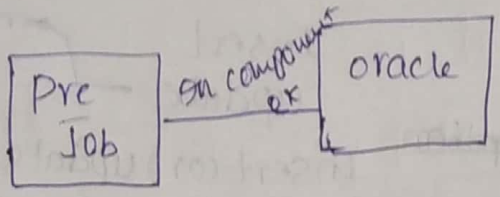
t oracle connection :- }
to run multiple jobs }

t- oracle connection:-

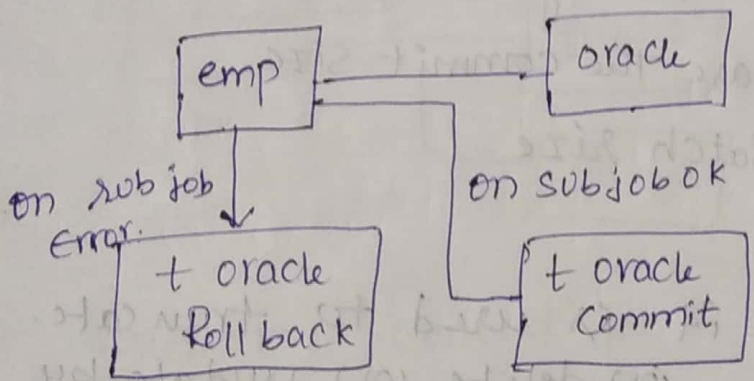
use t- orack commit
t- oracle Rollback

☑ use an existing connection.





* t. orack commit is used in t-oracle connection only.



* Go to Advance settings

☑ Auto commit

When use connection.

must use t Commit/ t Roll Back.

* ☑ use an existing connection.

Component to every component.

Bulk:- 18/09/19

t- oracle output Bulk execution:

Data loaded speedy.

* This component contain some rules.
* we didn't use this component as t oracle o/p.
* it should be plain table

we use this when it's only truncated.

* To use this component target table should not contain any constrains like primary key, foreign key, unique key and triggers and indexe
* Target table should be always truncate and load.

for this we need to create table.

```
create table bulk...
(
  "empno" int,
  "empname" varchar2(50),
  "Sal" int,
  "HRA" int,
  "deptno" int,
  "dept name" varchar2(30),
  "loc" varchar2(10),
)
```

* while creating a target table, make sure - target table columns should be in double quotes.

* Action on table default.

file name : file location.

3 files : total file . csv . (data file)

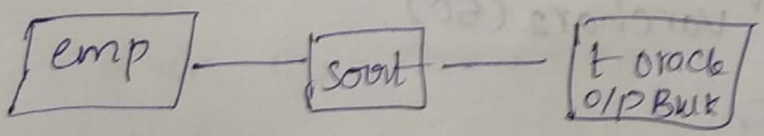
ctr . file (control)

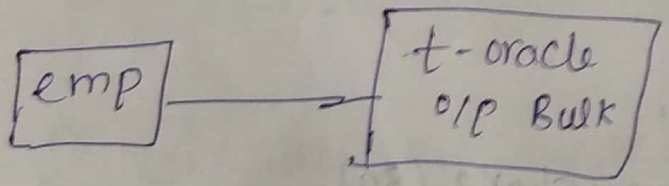
log file → time span
Begin
end ...

→ Bulk is loads data from the flow to the table with high speed.

→ Before loading it creates csv file and ctr file and loads data from csv to table directly.

→ To use this component target table should be empty (or) we can go with truncate option





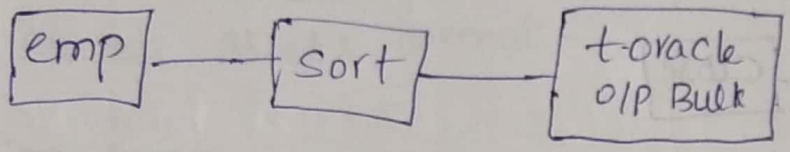
* At this time we had an another compon

t- oracle Bulk execution.

oracle execution component:

table.
file.
copy schema.

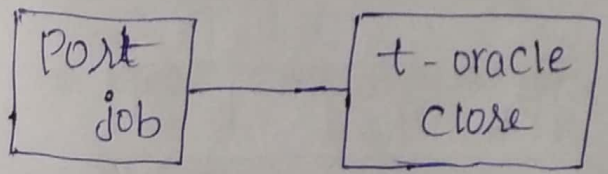
deactivate.



t-oracle execution comp
copy. file
file
copy schema

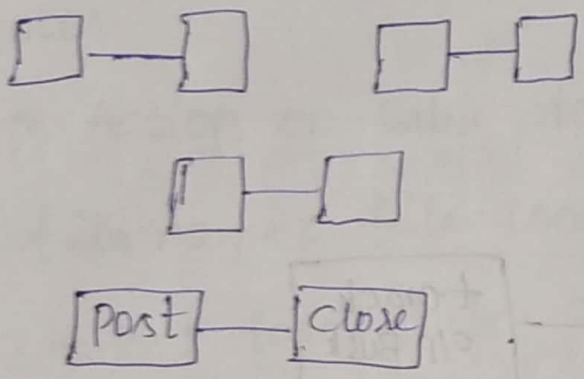
* to know exact Error (where it should occured)
Go to code.
* Search line no [Ctrl L —].
(or)
* Debug Run.

* When we use t-oracle connection.
* if it doesnot closed.
then use t-oracle close.



Prejob

↳ Connection.



Best practise:

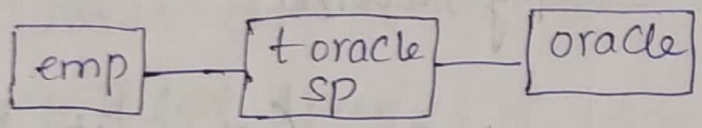
→ context create
→ tstat catcher
→ t log catcher
→ schema compliance check

Stored procedure:

How to call stored procedure in Talend?

t-oracle sp:-

t-oracle Row



double click

Sp name : "tax cal".

edit Schema: extra column tax.

Parameters: HRA, empno, Sal } HRA, Tax.

we go with t-oracle sp / t-oracle Row

Reading: 19/09/19

Excell file throw Talend:

1st Check properties xlsx, 10MB

☑ Read excel (2007 file format (xlsx))

Read excell assets format.

Generation Mode:

[less memory consumed four large excel (event-model)]

Select file → Choose file

Next ---- finish

Take as input.

if other Excell.

* All sheets have same metadata, Select all;

Next ---

* four small file generation mode:

[

* if we want sheet name also in O/P.

* Click on t map

* Add column as River,

Select → current sheet [global variable]

t-file output excel:

* if we want excel file as output: (create excel file)

☑ [2007 file format] (Read excel)

file name.

----/folder/excelout.xlsx" & sheet name;

To put that sheet any where like middle of sheet / corner ----

☑ is absolute Y pos. ←

first cell x[5/-] --- Y[8/-]

to Scan metadata for

create file positional:],

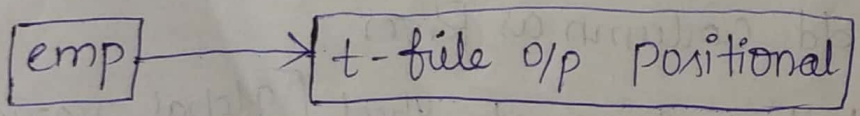
Select file

Select lengths, next --

take as input → ----

t-file output positional:

Output positional: T file output positional.



20/09/19

file name ✓

☑ include header.

Reading xml file input:

Scan meta data Loop limit [-1]

Schema viewer → Next. unlimited Records

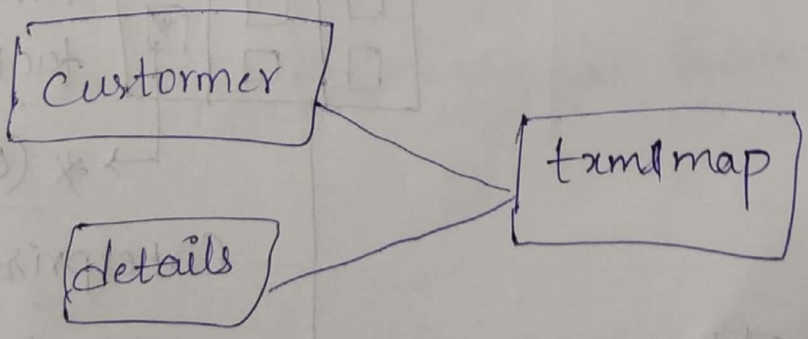
→ Absolute x path. Expression [-1].

Ex: / power mart / Repository / folder / source / source field.

→ field to extract Expression.

Ex:- copy → drag here.

Xml creation (output) :- txml map component is used.



JSON file :-

Create

⊙ i/p

Next

Browse file

Abs. path [-I].

↳ Relative / absolute path expression.

↳
↳
↳

Sales force :

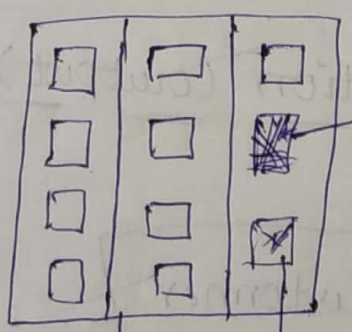
GE power

Greenplum

ODS

oracle
Excel
xml

→ Talend



* here tables ar
called as
rimension table
fact table

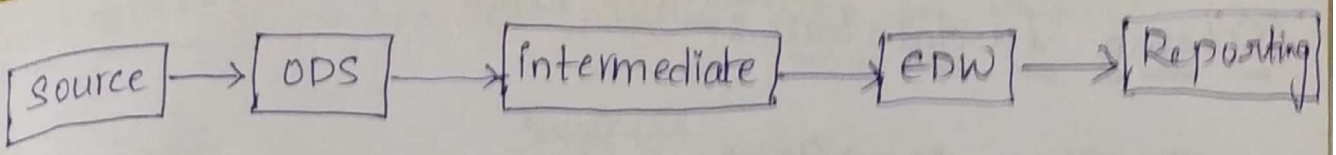
* (EDW) :

Enterprise data
ware house

Reposting layer

Reports

validations
(intermediate tables)



Dimensions :-

Facts :-

→ with respect to dimensions only you can derive facts.

21/09/19

* Dimensional table well have a detailed information.
* Dimensional table well have primary key.

Fact table :-

* fact table is a foreign keys of all and fact values.
* We can make an entry only when the recover existing dimensional table

Schemia in database :- ↱ Data ware house

* Star, Snowflake.

Star schema :- way of representing fact tables :-

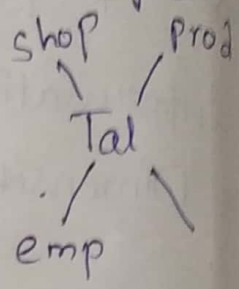
dept dim

Shop id
prod id
emp id

Prod. dim

pro id
pro name

Star type:



Shop dim

Shop id
Shop name
Shop dimension

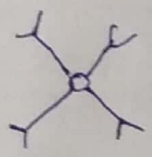
Talend dim

cal id
date
month
week

empdim

emp id
emp name
sal
HRA

Snow flake :- → we are mostly used in projects.



Business

4 dept → 10, 20, 30, 40

4K employees

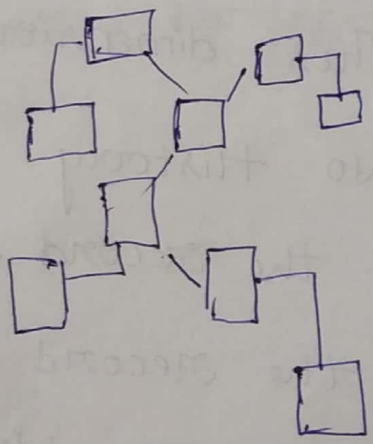
empid	ename	sal	HRA	dept name
⋮			⋮	10
				10
1000				⋮
1001				20

* Duplicate are there.

→ then remove all columns.

again create table

D.no	D name	loc
10	-	-
20	-	-
30	-	-



Speed : star

memory (less) : snow flake.

Galaxy schema :-

Dimensions :- 3 types of dimensions.

* Slowly Changing dimension :-

- SCD-I
- SCD-II
- SCD-III

* SCD-I :-

i/p

eno	ename	Sal	HRA	Dept no
100	A	1000	100	10
101	B	1800	180	20
102	C	1500	150	10
103	D	1200	120	20

* This dimensions always maintain

* No History

If the record is coming as a New

if the record is coming with update

if the record is coming

ignore it (or) Rejected it.

* SCD-II : * 4 extra columns is there.

emp key	empno	ename	sal	HRA	D.NO	SCDstdate
1	100	A	1000	100	10	17/08/19
2	101	B	1800	180	20	17/08/19
3	102	C	1500	150	10	17/08/19
4	104	D	1200	120	20	17/08/19

SCD Enddate	Activeflag
31/12/1999	Y
31/12/1999	Y
31/12/1999	Y
31/12/1999	Y

* update
* we are used in Primary key.
* Here primary key is a emp key.

* SCD to maintains complete history.
* By adding 4 extra columns in the target. Call surogate key (empkey).
* SC ~~Started~~, ~~Endedate~~
*

Surogate Key:

* which primary key in SCD-II table.

which is generated by ETL

Ex: Talend.

* using sequence numbers (numaric. sequence functions)

* if the record
* Insert it in the target with as new sequence number start date as curr date.
* End date as max date (31/12/1999).
* Active flag as 'Y' or '1'.

*If the record is coming with
* Insert it as a new record with new - sequence number and update the old records with end date as current date.
* Active flag as 'N'.
* To update the old record we use Surrogate Key as the key column.

SCD-3 :-

eno	ename	pre sal	curr-sal	pre-HRA	curr-HRA	Dept no / ~~Etl date~~
100	A	~~10~~	1000		100	10
101	B	1800	~~-1~~1900		180	20
102	C		1500	150	250	10
103	D		1200		120	20

Etl date
18/8/19
18/8/19
18/8/19
18/8/19

Same - ignore
Change - pre
New Record - current

25/09/19

SCD 7: Passing context parameter

~~How to~~

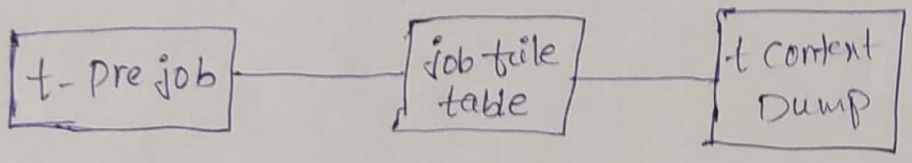
Passing context variables from parent to child:-

* ☑ transmit whole context.

Only some variables changes:

Content parameter →
(+) Button.

Child to parent:-



* Go to child job take context Dump component and content to Butter o/p.
* Come to parent job.
  * Select t-run job.
    * Go to component job.
    * click on copy child job schema.
* Now t-run job to t context load using row main.

10/9/19

4 Tables (Frame work)

* Job execution log
* Audit table
* Error file 1
* Error file 2.

t - set Global Var.

↓

Key → max key.

row6.max

Job execution log: Execution id

proid

job name

~~execution id~~