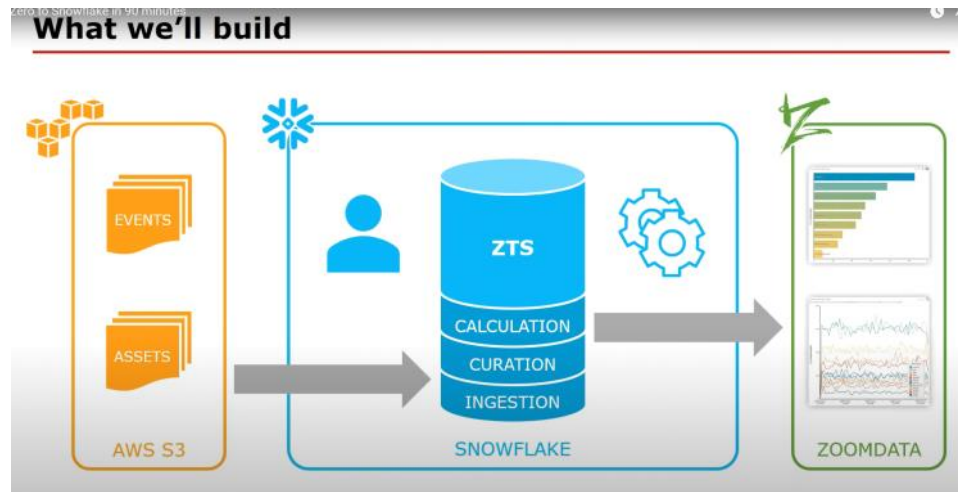
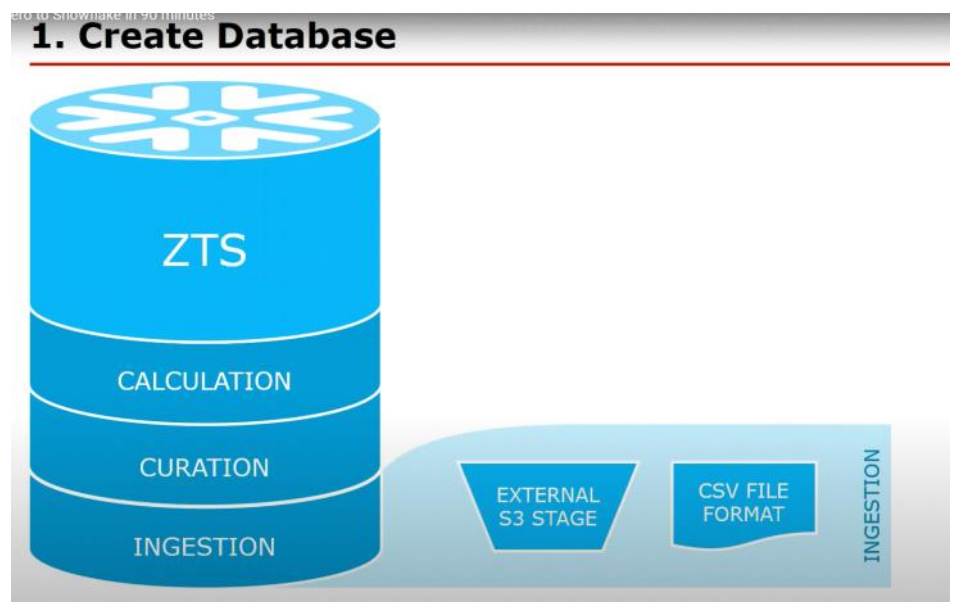


Snowflake

Sunday, April 18, 2021 10:00 AM



#1.create database



The screenshot shows the Snowflake web interface. The top navigation bar includes 'snowflake', 'Databases', 'Warehouses', 'Worksheet', and 'History'. The 'Databases' section is active, showing a table of databases.

Database	Origin	Creation Time	Owner	Comment
SNOWFLAKE_SAMPLE_DATA	SFC_SAMPLES.SA...	3/19/18 4:49:38 PM	ACCOUNTADMIN	TPC-H, OpenWeatherMap, etc
UTIL_DB		3/19/18 4:49:35 PM	SYSADMIN	utility database

#database: database gives us lists of the high level database objects that we got.

Databases > SNOWFLAKE_SAMPLE_DATA

Tables		Views	Schemas	Stages	File Formats	Sequences
Create...	Create Like...	Clone...	Load Data...	Drop...	Transfer Ownership	
Table Name	Schema	Creation Time ▾	Owner	Rows	Size	Comment
STORE_SALES	TPCDS_SF1...	2/14/18 6:32:49 AM		<div><div></div></div> 288.0G	<div><div></div></div> 10.9TB	
CATALOG_SALES	TPCDS_SF1...	2/14/18 6:32:49 AM		<div><div></div></div> 144.0G	<div><div></div></div> 9.3TB	

Warehouses: the pools of compute resources that we use to do operations, e.g. put data in, process data, query data,
#create database and switch to it

Worksheet : predefined SQL statement

```
CREATE OR REPLACE DATABASE ZTS;
USE ZTS
```

#create schemas

```
Create or replace schema ingestion;
Create or replace schema curation;
Create or replace schema calculation;
```

Show schemas;

Row	created_on	name	is_default	is_current	database_name	owner	comment	options	retention_time
1	2018-05-03 10:19:2...	CALCULATION	N	Y	ZTS	ZTS_ADMIN			1
2	2018-05-03 10:19:2...	CURATION	N	N	ZTS	ZTS_ADMIN			1
3	2018-05-03 10:19:3...	INFORMATION_SC...	N	N	ZTS		Views describing th...		1
4	2018-05-03 10:19:2...	INGESTION	N	N	ZTS	ZTS_ADMIN			1
5	2018-05-03 10:18:3...	PUBLIC	N	N	ZTS	ZTS_ADMIN			1

Public is the default schema, information_schema is where all the system views are in terms of the metadata and database itself

#create an external, s3 stage(public bucket)

Create or replace stage ingestion.s3_stage

Url = 's3://dlx-zero-to-snowflake'

Credentials = {

Aws_key_id = 'AKIAIRIY07UUKKJXX64A'

'YX67150BujDXff1JPe0wtJU8qESyAf3NQkbKZYe9'

AWS_SECRET_KEY =

};

List @ingestion.s3_stage; #draw us out a list of all the objects and files in that stage

Row	name	size	md5	last_modified
1	s3://dlx-zero-to-snowflake/assets/20180302-102317...	34785	e5665c6c754e92c4210ef86735ad8888	Fri, 2 Mar 2018 10:23:21 GMT
2	s3://dlx-zero-to-snowflake/events/20180302-092512...	5602753	59b53e7c91334c23d46eb8920ff12fb-2	Fri, 2 Mar 2018 09:25:23 GMT
3	s3://dlx-zero-to-snowflake/events/20180302-092512...	5649112	c9c04cd06ff428fb78fb3559c9d9b784-2	Fri, 2 Mar 2018 09:25:23 GMT
4	s3://dlx-zero-to-snowflake/events/20180302-092512...	5613139	e3240727a35d270901dcd56cb24e5d78-2	Fri, 2 Mar 2018 09:25:23 GMT
5	s3://dlx-zero-to-snowflake/events/20180302-092512...	5629833	669ce6f3b74b0c2d7112c55141f87181-2	Fri, 2 Mar 2018 09:25:23 GMT
6	s3://dlx-zero-to-snowflake/events/20180302-092512...	5615627	0bda05ee9f4c12e25fe23b6d130092e3-2	Fri, 2 Mar 2018 09:25:32 GMT

#create CSV files format for ingestion

```
-- Create CSV file format for ingestion
CREATE OR REPLACE FILE FORMAT INGESTION.CSV_FORMAT
  TYPE = 'CSV'
  COMPRESSION = 'AUTO'
  FIELD_DELIMITER = ','
  RECORD_DELIMITER = '\n'
  SKIP_HEADER = 1
  FIELD_OPTIONALLY_ENCLOSED_BY = '\042'
  ERROR_ON_COLUMN_COUNT_MISMATCH = TRUE
  ESCAPE = 'NONE'
  ESCAPE_UNENCLOSED_FIELD = '\134'
  NULL_IF = ('NULL');
```

Databases > ZTS

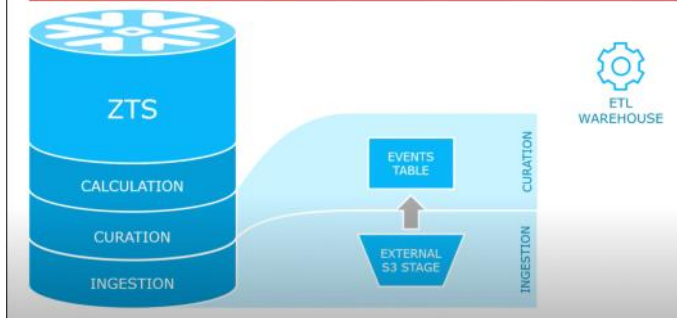
Tables Views Schemas Stages **File Formats** Sequences

+ Create... Clone... Edit... Drop... Transfer Ownership

File Format	Schema	Type	Creation Time	Owner	Comment
CSV_FORMAT	INGESTION	CSV	11:22:18 AM	ZTS_ADMIN	

2. curate events data

2. Curate Events Data



#create curation table for events

Create or replace table curation.events{

```

  Id string,
  Asset_id string,
  Metric_id string,
  Value number(18,9),
  Created int

```

};

Desc table curation.events; -- Also look at this via the database tab

snowflake

Databases Warehouses Worksheet History

Databases > ZTS > EVENTS (CURATION)

Tables Views Schemas Stages File Formats Sequences

Load Table

Column Name	Ordinal	Type	Nullable	Default	Comment
ID	1	VARCHAR(16777216)	true	NULL	
ASSET_ID	2	VARCHAR(16777216)	true	NULL	
METRIC_ID	3	VARCHAR(16777216)	true	NULL	
VALUE	4	NUMBER(18,9)	true	NULL	
CREATED	5	NUMBER(38,0)	true	NULL	

#now need a warehouse

CREATE or replace warehouse zts_etl_mh with

```

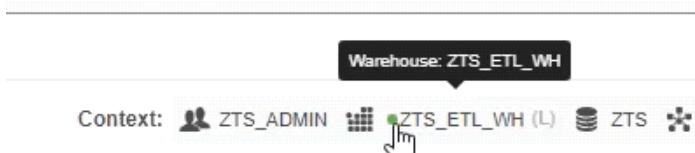
  Warehouse_size = 'large'
  Auto_suspend = 300
  Auto_resume = true
  Comment = 'Zero to Snowflake ETL Warehouse';

```

Grant usage on warehouse zts_etl_wh to role zts_admin;

Use warehouse zts_etl_wh;

#green dot means live



copy in events from s3 stage(LARGE 1M20S)

Copy into curation.events

From @ingestion.s3_stage/events/

FILE_FORMAT = (FORMAT_NAME = 'INGESTION.CSV_FORMAT');

SELECT * FROM CURATION.EVENTS LIMIT 10;

Row	ID	ASSET_ID	METRIC_ID	VALUE	CREATED
1	49cf7594-bcca-4cd7-b4ad-3318d1d7f2cb	aab33e6c-7fcc-33e5-b422-e0cf5e21de12	77a379dc-6603-3d6a-8d2c-55b7ec685f65	0.401249200	1523338880
2	a5f71dac-34c2-4e78-b25f-9106cd04bc1f	98cf19c2-e12e-37cc-9e0c-e73f285da5f6	c40cf251-a6d0-3712-aaee-209ccae1fed3	0.133727875	1523338776
3	8ead7a30-2747-46b2-a186-1c99d9aad4e8	44d6c837-1bca-3c69-9eff-30b235aa624f	376b4c88-9410-325a-8ed5-67deae77bf0d	0.149774081	1523339809
4	11ae4f96-fc2c-4dab-8f3d-aa80a92ef737	dd3aa249-aa9b-308a-8501-412d38630f67	d93864ab-bf1a-3b1c-b551-c89f95ccbc88	0.281515376	1523339479
5	2a3a5efc-e381-4750-82ec-318a09f93c10	b9aafce4-175b-3c92-839a-bfd50ad5e5bc	c40cf251-a6d0-3712-aaee-209ccae1fed3	0.964294689	1523336596
6	eb35f8ca-f515-498d-ae88-52583328259a	0c172076-fc19-3572-af7b-8af1bc29b173	f8e59b70-32de-37b4-9523-fabb19895712	0.650793384	1523338231

#Try again with 2xl (30S)

TRUNCATE TABLE CURATION.EVENTS;

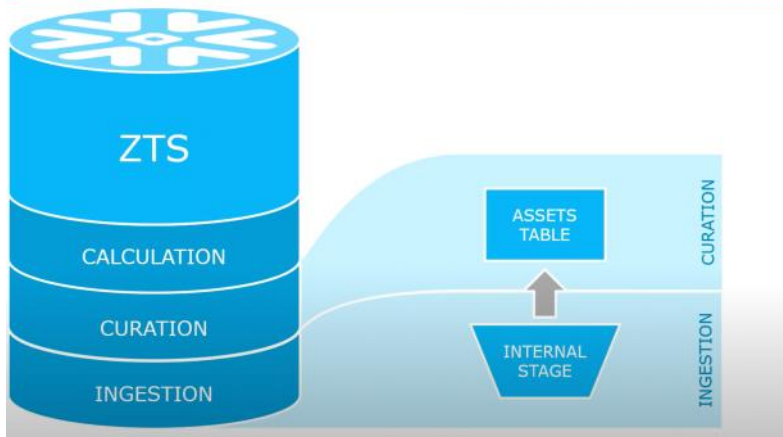
ALTER WAREHOUSE ZTS_ETL_WH SET WAREHOUSE_SIZE = 'XXLARGE';

#switch the warehouse back to large for now (THIS SORT OF THING CAN BE AUTOMATED)

ALTER WAREHOUSE ZTS_ETL_WH SET WAREHOUSE_SIZE = 'LARGE';

#3.CURATE ASSETS DATA

3. Curate Assets Data



#CREATE AN INTERNAL, SNOWFLAKE-NAMAGED STAGE

CREATE OR REPLACE STAGE INGESTION.INTERNAL_STAGE;

Databases > ZTS					
Tables Views Schemas Stages File Formats Sequences					
+ Create... Clone... Edit... Drop... Transfer Ownership					
Stage	Schema	Location	Creation Time	Owner	Comment
INTERNAL_STAGE	INGESTION	Snowflake	11:35:42 AM	ZTS_ADMIN	
S3_STAGE	INGESTION	s3://dx-zero-to-snowflake	11:20:17 AM	ZTS_ADMIN	

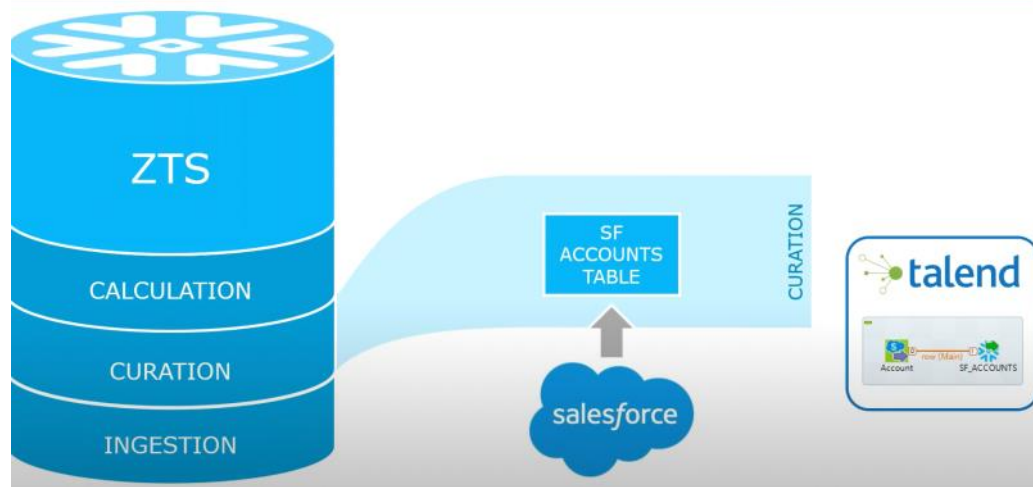
-- push this assets file using snowsql

-- Push the assets file using SnowSQL

```
/*
snowsql --accountname datalytx.eu-west-1 --username ZTS_ALEX
USE ZTS;
USE WAREHOUSE ZTS_ETL_WH;
PUT file:///D:/Temp/ZeroToSnowflake/assets/* @INGESTION.INTERNAL_STAGE/assets/;
*/
```

USE INGESTION.INTERNAL_STAGE;

4. Curate Salesforce Data



#LOOK AT THE ACCOUNTS IN SALESFORCE
#CREATE SF ACCOUNTS TABLE

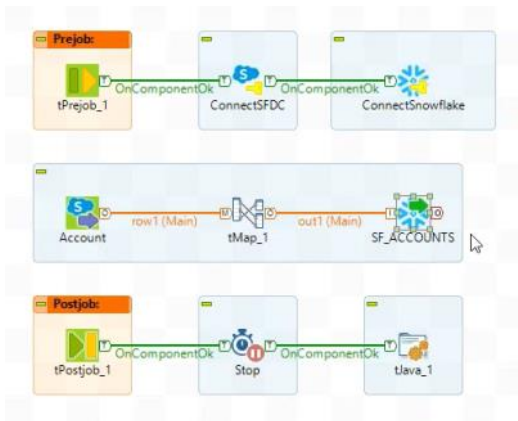
CREATE OR REPLACE TABLE CURATION.SF_ACCOUNTS (

```
ID STRING,  
ISDELETED BOOLEAN,  
MASTERRECORDID STRING,  
NAME STRING,  
TYPE STRING,  
PARENTID STRING,  
BILLINGSTREET STRING,  
BILLINGCITY STRING,  
BILLINGSTATE STRING,  
BILLINGPOSTALCODE STRING,  
BILLINGCOUNTRY FLOAT,  
BILLINGLATITUDE FLOAT,  
BILLINGGEOCODEACCURACY STRING,  
BILLINGADDRESS STRING,  
SHIPPINGSTREET STRING,  
SHIPPINGCITY STRING,  
SHIPPINGSTATE STRING,  
SHIPPINGPOSTALCODE STRING,  
SHIPPINGCOUNTRY FLOAT,  
SHIPPINGLATITUDE FLOAT,  
SHIPPINGGEOCODEACCURACY STRING,  
SHIPPINGADDRESS STRING,  
PHONE STRING,
```

```
ANNUALREVENUE NUMBER(18,2),  
NUMBEROFEMPLOYEES INTEGER,  
OWNERSHIP STRING,  
TICKERSYMBOL STRING,  
DESCRIPTION STRING,  
RATING STRING,  
SITE STRING,  
OWNERID STRING,  
CREATEDDATE TIMESTAMP_NTZ,  
CREATEDBYID STRING,  
LASTMODIFIEDDATE TIMESTAMP_NTZ,  
LASTMODIFIEDBYID STRING,  
SYSTEMMODSTAMP TIMESTAMP_NTZ,  
LASTACTIVITYDATE TIMESTAMP_NTZ,  
LASTVIEWEDDATE TIMESTAMP_NTZ,  
LASTREFERENCEDATE TIMESTAMP_NTZ,  
JIGSAW STRING,  
JIGSAWCOMPANYID STRING,  
CLEANSTATUS STRING,  
ACCOUNTSOURCE STRING,  
DUNSNUMBER STRING,  
TRADESTYLE STRING,  
NAICS CODE STRING,  
NAICSDESC STRING,  
YEARSTARTED STRING,  
SICDESC STRING,  
DANDBCOMPANYID STRING,  
CUSTOMERPRIORITY__C STRING,
```

```
CUSTOMERPRIORITY__C STRING,  
SLA__C STRING,  
ACTIVE__C STRING,  
NUMBEROFLOCATIONS__C FLOAT,  
UPSELLOPPORTUNITY__C STRING,  
SLASERIALNUMBER__C STRING,  
SLAEXPIRATIONDATE__C TIMESTAMP_NTZ  
);
```

#LOAD THE sf ACCOUNTS USING Talend

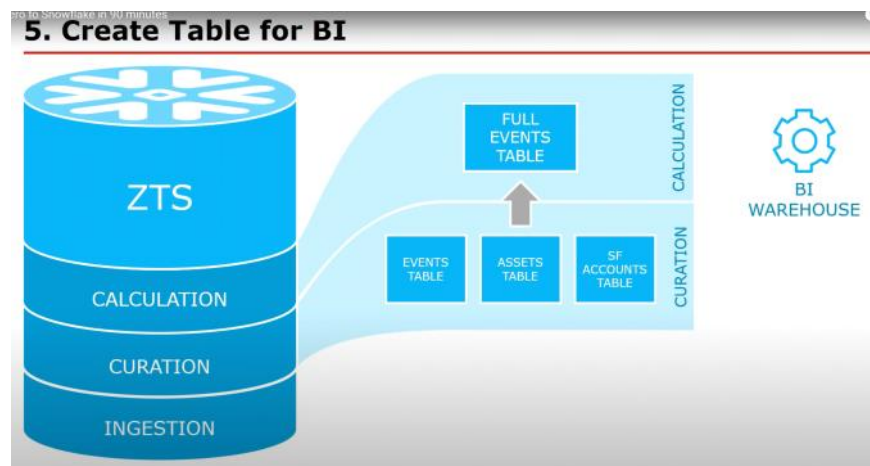


#run job Demo/ZeroToSnowflake/ZeroToSnowflake30_IngestSFDCAccountsToSnoeflake

SELECT * FROM CURATION.SF_ACCOUNTS LIMIT 10;

Row	ID	ISDELETED	MASTERRE...	NAME	TYPE	PARENTID	BILLINGSTR...	BILLINGCITY	BILLINGSTATE	BILLINGPO...	BILLINGCO...	BILLINGLATI...	BILL
1	0010Y00001...	FALSE	NULL	Dalton & Ste...	Customer - ...	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
2	0010Y00001...	FALSE	NULL	A. Jones Co...	Customer - ...	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
3	0010Y00001...	FALSE	NULL	Taylor Comp...	Customer - ...	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
4	0010Y00001...	FALSE	NULL	Thompson & ...	Customer - ...	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
5	0010Y00001...	FALSE	NULL	Stokes & Bar...	Customer - ...	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
6	0010Y00001...	FALSE	NULL	Edwards Fur...	Customer - ...	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

5. Create Table for BI



-- Create a flattened events table, joining with the other tables

```
CREATE OR REPLACE TABLE CALCULATION.EVENTS AS
SELECT
  E.ID AS EVENT_ID,
  E.METRIC_ID,
  E.VALUE,
  TO_TIMESTAMP_NTZ(E.CREATED) AS EVENT_CREATED,
  DATE_TRUNC('MINUTE', TO_TIMESTAMP_NTZ(E.CREATED)) AS EVENT_CREATED_MINUTE,
  A.ID AS ASSET_ID,
  A.TYPE AS ASSET_TYPE,
  A.ATTRIBUTES AS ASSET_ATTRIBUTES,
  TO_TIMESTAMP_NTZ(A.CREATED) AS ASSET_CREATED,
  TO_TIMESTAMP_NTZ(A.LAST_UPDATED) AS ASSET_LAST_UPDATED,
  A.CLIENT_ID,
  A.CLIENT_NAME,
  S.ID AS SF_ACCOUNT_ID,
  S.NAME AS SF_ACCOUNT_NAME,
  S.TYPE AS SF_ACCOUNT_TYPE,
  S.INDUSTRY AS SF_ACCOUNT_INDUSTRY
FROM CURATION.EVENTS E
INNER JOIN CURATION.ASSETS A ON E.ASSET_ID = A.ID
INNER JOIN CURATION.SF_ACCOUNTS S ON A.CLIENT_SFDC_ID = S.ID;
```

#take a look at the table, noting that data comes from the WH cache
SELECT * FROM CALCULATION.EVENTS LIMIT 10;

Results

Data Preview

Query ID

SQL

3.44s

924 0MB bytes scanned

10 rows

LATEST

Filter result...

Copy

Columns

Raw	EVENT_ID	METRIC_ID	VALUE	EVENT_CR...	EVENT_CR...	ASSET_ID	ASSET_TYPE	ASSET_ATT	ASSET_CR...	ASSET_LAS...	CLIENT_ID	CLIENT_NA...	SF
1	6d084020-99...	e4a518a2-fa...	0.975695495	2018-01-08 ...	2018-01-08 ...	c8393895-9e...	LAPTOP	()	2018-03-02 ...	2018-03-02 ...	ee94ee42-f3f...	DALTON & S...	001
2	a5131d88-4ff...	77a379dc-66...	0.570241416	2018-01-08 ...	2018-01-08 ...	e3047ea9-44...	SERVER	()	2018-03-02 ...	2018-03-02 ...	ab5b1c50-2c...	A. JONES C...	001
3	2bbc78ed-40...	536dd227-38...	0.847163238	2018-01-08 ...	2018-01-08 ...	c36266b8-8b...	LAPTOP	()	2018-03-02 ...	2018-03-02 ...	2c96045d-4e...	THOMPSON...	001
4	595b8c33-5e...	95132e71-b9...	0.099793426	2018-01-08 ...	2018-01-08 ...	87995305-30...	LAPTOP	()	2018-03-02 ...	2018-03-02 ...	59c93b0f-22...	LEWIS COM...	001
5	a7c79fa4-62...	376b4c96-64...	0.815043771	2018-01-08 ...	2018-01-08 ...	06c7ed24-ba...	VAN	()	2018-03-02 ...	2018-03-02 ...	c79a3d4e-c1...	PAGE LUK	001

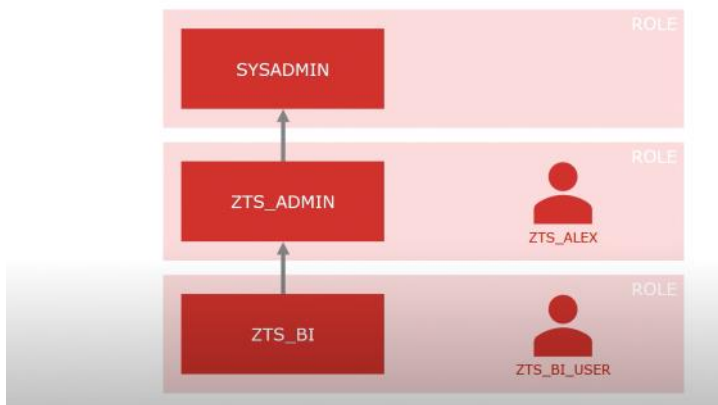
#RUN THE SELECT AGAIN, NOTING THAT DO DATA IS SCANNED(RESULTS CACHE)

#CREATE WAREHOUSE FOR bi

```
CREATE OR REPLACE WAREHOUSE ZTS_BI_WH WITH
  WAREHOUSE_SIZE = 'MEDIUM'
  AUTO_SUSPEND = 300
  AUTO_RESUME = TRUE
  COMMENT = 'Zero to Snowflake BI Warehouse';
```

6. Set permission for BI

6. Set Permissions for BI



#assume accountadmin role

Use role accountadmin; #or use menu

#create role for BI

```
Create or replace role ZTS_BI COMMENT = 'Zero to Snowflake - BI Role';
GRANT ROLE ZTS_BI TO ROLE ZTS_ADMIN;
```

```
GRANT USAGE ON DATABASE ZTS TO ROLE ZTS_BI;
GRANT USAGE ON SCHEMA ZTS.CALCULATION TO ROLE ZTS_BI;
GRANT USAGE ON WAREHOUSE ZTS_BI_WH TO ROLE ZTS_BI;
```

#GRANT BI ROLE SELECT ACCESS TO FLATTENED EVENTS TABLE

```
GRANT SELECT ON CALCULATION.EVENTS TO ROLE ZTS_BI;
```

#CREATE USER FOR BI (USER UI - SWITCH ROLE TO ACCOUNTADMIN

```
CREATE OR REPLACE USER ZTS_BI_USER
  PASSWORD = 'Brear5'
  DISPLAY_NAME = 'BI User'
  COMMENT = 'Zero to Snowflake - BI user'
  DEFAULT_ROLE = ZTS_BI
  DEFAULT_WAREHOUSE = ZTS_BI_WH
;
```

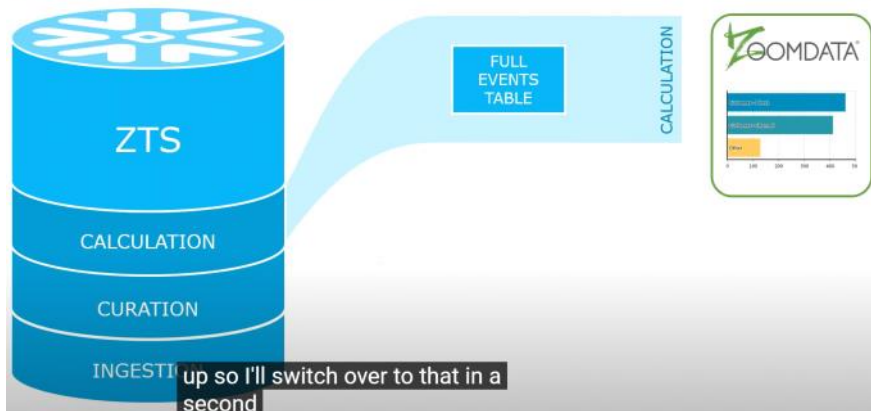
```
GRANT ROLE ZTS_BI TO USER ZTS_BI_USER;
```

#switch back to ZTS_ADMIN ROLE

```
USE ROLE ZTS_ADMIN;
```

7. SETUP BI

7. Set Up BI



8. SHOW CONCURRENCY

8. Show Concurrency



#USE etl WAREHOUSE

USE WAREHOUSE ZTS_ETL_WH;

#TURN OFF THE RESULTS CACHE FOR NOW

ALTER SESSION SET USE_CACHED_RESULT = FALSE;

#CALCULATE DIFFERENCE OF DIFFERENCES (3 POINTS) AVERAGED BY ASSET TYPE

```
SELECT ASSET_TYPE, AVG(DIFF) FROM (
  SELECT
    ASSET_TYPE,
    VALUE AS CURRENT_VALUE,
    LAG(VALUE) OVER(PARTITION BY ASSET_ID, METRIC_ID ORDER BY EVENT_CREATED) AS PREV_VALUE,
    LEAD(VALUE) OVER ( PARTITION BY ASSET_ID, METRIC_ID ORDER BY EVENT_CREATED) AS NEXT_VALUE,
    CURRENT_VALUE - PREV_VALUE AS PREV_DIFF,
    NEXT_VALUE - CURRENT_VALUE AS NEXT_DIFF,
    NEXT_DIFF - PREV_DIFF AS DIFF
  FROM CALCULATION.EVENTS
)
GROUP BY ASSET_TYPE
ORDER BY ASSET_TYPE
;
```

#Show zoomdata still working

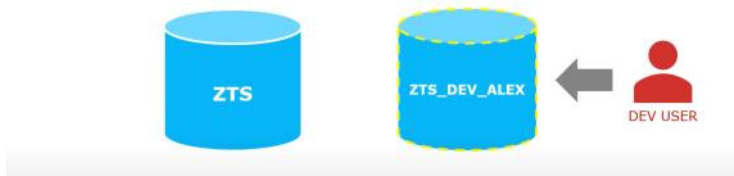
#look at the query profile

#turn the results cache back on

9. CREATE DEVELOPMENT SANDBOX

ZTS_DEV_ALEX :It's a zero copy clone which means it takes a copy of the database and all the metadata but it refers back to the original underlying blocks on storage now.

9. Create Development Sandbox



#assume ZTS_ADMIN ROLE (ALSO SWITCH ui BACK TO ZTS_ADMIN)

USE ROLE ZTS_ADMIN;

#CREATE A TEMPORARY DEVELOPMENT DATABASE

CREATE OR REPLACE DATABASE ZTS_DEV_ALEX CLONE ZTS;
USE ZTS_DEV_ALEX;

The screenshot shows the Snowflake web interface. The 'Databases' tab is selected, displaying a table of existing databases. The table has columns: Database, Origin, Creation Time, Owner, and Comment.

Database	Origin	Creation Time	Owner	Comment
ZTS_DEV_ALEX		12:06:40 PM	ZTS_ADMIN	Zero to Snowflake
ZTS		11:18:32 AM	ZTS_ADMIN	Zero to Snowflake
SNOWFLAKE_SAMPLE_DATA	SFC_SAMPLES SA...	3/19/18 4:49:38 PM	ACCOUNTADMIN	TPC-H, OpenWeatherMap, etc
UTIL_DB		3/19/18 4:49:35 PM	SYSADMIN	utility database

Databases > ZTS_DEV_ALEX

The screenshot shows the 'Tables' tab for the 'ZTS_DEV_ALEX' database. It displays a table of tables with columns: Table Name, Schema, Creation Time, Owner, Rows, Size, and Comment.

Table Name	Schema	Creation Time	Owner	Rows	Size	Comment
EVENTS	CALCULATI...	12:06:40 PM	ZTS_ADMIN	249.7M	11.3GB	
ASSETS	CURATION	12:06:40 PM	ZTS_ADMIN	1K	44.5KB	
EVENTS	CURATION	12:06:40 PM	ZTS_ADMIN	250.0M	9.7GB	
SF_ACCOUNTS	CURATION	12:06:40 PM	ZTS_ADMIN	20	4.5KB	

#show the asset types we currently have

```
SELECT TYPE, COUNT(ID) AS NUM_ASSETS
FROM CUEATION.ASSETS
GROUP BY TYPE
ORDER BY NUM_ASSETS;
```

The screenshot shows the results of a query. The table has two columns: TYPE and NUM_ASSETS. The results are as follows:

Row	TYPE	NUM_ASSETS
1	PLANE	17
2	YACHT	33
3	SERVER	42
4	DRONE	48
5	HELICOPTER	52
6	SWITCH	60

#INSERT A LOAD OF TEST ASSETS

```
INSERT INTO CURATION.ASSETS (ID, TYPE) VALUES
('00000000-0000-0000-0000-000000000001','TEST_ASSET'),
('00000000-0000-0000-0000-000000000002','TEST_ASSET'),
('00000000-0000-0000-0000-000000000003','TEST_ASSET'),
('00000000-0000-0000-0000-000000000004','TEST_ASSET'),
('00000000-0000-0000-0000-000000000005','TEST_ASSET');
```

#show the test assets are now in our sandbox

```
SELECT COUNT(ID) AS NUM_ASSETS
FROM ZTS_DEV_ALEX.CURATION.ASSETS
WHERE TYPE = 'TEST_ASSET';
```

Row	NUM_ASSETS
1	5

#but not in the production database

```
SELECT COUNT(ID) AS NUM_ASSETS
FROM ZTS.CURATION.ASSETS
WHERE TYPE = 'TEST_ASSET';
```

Row	NUM_ASSETS
1	0

10. WORK WITH JASON DATA

10. Work with JSON Data

JSON_FIELD	
{	
"make": "Ford",	JSON_FIELD:make
"model": {	
"base": "Focus",	JSON_FIELD:model.base
"trimLevel": "Titanium"	JSON_FIELD:model.trimLevel
}	
}	

#still in the sandbox, look at some JSON data

```
SELECT ATTRIBUTES FROM CURATION.ASSETS
WHERE TYPE = 'CAR'
LIMIT 1;
```

#SEE WHAT THE DATA LOOKS LIKE:

Details	
1	{
2	"engine": {
3	"fuel": "PETROL",
4	"size": 2.5
5	},
6	"make": "FORD",
7	"model": {
8	"base": "KA",
9	"trimLevel": "EDGE"
10	},
11	"registration": "YJ07 YPB",
12	"year": 2007
13	}

#SHOW TOP10 MAKES/MODELS

```
SELECT
  ATTRIBUTES:make::STRING AS MAKE, #NEED TO DEFINE THE DATA FORMATS
  ATTRIBUTES:model.base::STRING AS MODEL,
  COUNT(*) AS NUM_VEHICLES
FROM CURATION.ASSETS
```

```

WHERE TYPE = 'CAR'
GROUP BY MAKE, MODEL
ORDER BY NUM_VEHICLES DESC
LIMIT 10;

```

Row	MAKE	MODEL	NUM_VEHICLES
1	FORD	FIESTA	25
2	FORD	FOCUS	17
3	VAUXHALL	CORSA	13
4	VOLKSWAGEN	GOLF	13
5	VOLKSWAGEN	POLO	11
6	VAUXHALL	ASTRA	8

#show the avg(mid) year of all GG prefix registrations

```

SELECT
  MEDIAN(ATTRIBUTES:year::INTEGER)::INTEGER AS AVG_YEAR
FROM CURATION.ASSETS
WHERE
  TYPE = 'CAR'
  AND LEFT(ATTRIBUTES:registration::STRING,2) = 'GG';

```

Row	AVG_YEAR
1	2013

#CREATE A VIEW OF ALL car ASSETS

CREATE VIEW CALCULATION.CARS AS

```

SELECT
  ID,
  ATTRIBUTES,
  ATTRIBUTES:registration::STRING AS REGISTRATION,
  ATTRIBUTES:year::INTEGER AS YEAR,
  DATE_PART(YEAR,CURRENT_DATE()) - YEAR AS AGE,
  ATTRIBUTES:make::STRING AS MAKE,
  ATTRIBUTES:model.base::STRING AS MODEL,
  ATTRIBUTES:model.trimLevel::STRING AS TRIM_LEVEL,
  ATTRIBUTES:engine.fuel::STRING AS FUEL,
  ATTRIBUTES:engine.size::FLOAT AS ENGINE_SIZE,
  CREATED,
  LAST_UPDATED,
  CLIENT_ID
FROM CURATION.ASSETS
WHERE TYPE = 'CAR'
;

```

SELECT * FROM CALCULATION.CARS LIMIT 10;

Row	ID	ATTRIBUTES	REGISTRAT...	YEAR	AGE	MAKE	MODEL	TRIM_LEVEL	FUEL	ENGINE_SIZE	CREATED	LAST_UPDA...	CLIT
1	d3a464c7-57...	["engine": {"...	YJ07 YPB	2007	11	FORD	KA	EDGE	PETROL	2.5	1519986202	1519986202	fed
2	50ad3b99-9e...	["engine": {"...	OX13 JXX	2013	5	FORD	FIESTA	LX	DIESEL	1	1519986202	1519986202	06
3	5ac8b5b1-72...	["engine": {"...	MO16 FUD	2016	2	HONDA	JAZZ	SE	DIESEL	2	1519986202	1519986202	06
4	343d006e-4f...	["engine": {"...	FF63 WSA	2013	5	VAUXHALL	CORSA	SXI 16V	PETROL	1	1519986202	1519986202	7e9
5	7a89e1b9-0c...	["engine": {"...	MK59 JUW	2009	9	FORD	FOCUS	ZETEC	PETROL	3	1519986202	1519986202	ee9
6	c56bbadc-bb...	["engine": {"...	LC10 WDB	2010	8	TOYOTA	Missing	NULL	DIESEL	1	1519986202	1519986202	7e9

#DROP THE SANDBOX DATABASE

DROP DATABASE ZTS_DEV_ALEX;

11. ACCIDENTS HAPPEN -- UPDROP OUR SANDBOX DATABASE

SHOW DATABASES LIKE 'ZTS%';

SHOW DATABASE HISTORY LIKE 'ZTS%';

UNDROP DATABASE ZTS_DEV_ALEX;

#SWITCH TO PRODUCTION

USE ZTS;

#BREAK SOME DATA : HOW MANY CARS ARE THERE?

```
SELECT TYPE, COUNT(ID) AS NUM_ASSETS
FROM CURATION.ASSETS
WHERE TYPE = 'CAR'
GROUP BY TYPE'
```

#NOW ACCIDENTALLY DELETE THEM ALL

```
DELETE FROM CURATION.ASSETS WHERE TYPE = 'CAR';
```

#LOOK THEY'RE GONE

```
SELECT COUNT(ID) AS NUM_ASSETS
FROM CURATION.ASSETS
WHERE TYPE = 'CAR';
```

#LET'S GO BACK IN TIME TO 5 MINUTES AGO

```
SELECT COUNT(ID) AS NUM_ASSETS
FROM CURATION.ASSETS
AT (OFFSET >= -60*5)
WHERE TYPE = 'CAR';
```

#WE CAN USE THE TABLE AT A POINT IMMEDIATELY BEFORE WE RAN THE DELETE

```
SELECT COUNT(ID) AS NUM-ASSETS
FROM CURATION.ASSETS
BEFORE (STATEMENT => ")
WHERE TYPE = 'CAR'
;
```

#GO TO THE HISTORY AND FIND THE HISTORY QUERY ID

The screenshot shows the Snowflake History page. The top navigation bar includes 'snowflake', 'Databases', 'Warehouses', 'Worksheet', and 'History' (highlighted in yellow). Below the navigation bar, the page title is 'History > 12:12:36 PM for 1.2s'. The main content area is divided into two tabs: 'Details' (selected) and 'Profile'. The 'Details' tab shows the following information:

Status	Success
User	ZTS_ALEX
Warehouse	ZTS_ETL_WH
Start Time	12:12:36 PM
End Time	12:12:37 PM
Total Duration	1.2s
Scanned Bytes	60.5KB
Rows	814
Query ID	2c949ca9-0b71-4102-a541-c74f84d3d61f

The 'Query ID' is highlighted in blue. To the right of the 'Details' tab, the 'SQL Text' tab is visible, showing the following SQL statement:

```
1. DELETE FROM CURATION.ASSETS WHERE TYPE = 'CAR';
```

#SO THAT'S SELECT THOSE ROWS BACK INTO THE CURRENT TABLE

```
SELECT COUNT(ID) AS NUM-ASSETS
FROM CURATION.ASSETS
BEFORE (STATEMENT => '6c949ca9-0b71-410-a541-c74f84d3d61f')
WHERE TYPE = 'CAR'
;
```

#look they're back

```
SELECT COUNT(ID) AS NUM_ASSETS
FROM CURATION.ASSETS
WHERE TYPE = 'CAR'
;
```