

Write Less (Code) With More (New 26ai Features)

Oren Nakdimon

<https://linktr.ee/oren>

oren@db-oriented.com

<https://db-oriented.com>

Write Less (Code)
With More (New 26ai Features)

Write Less/Simpler (Code)
With More (New 26ai Features)

Write Less/Simpler/Better/... (Code) With More (New 26ai Features)

- Not a deep-dive session
- There are more... Many more...
- Features that give value without big efforts

<https://db-oriented.com>



SYML²



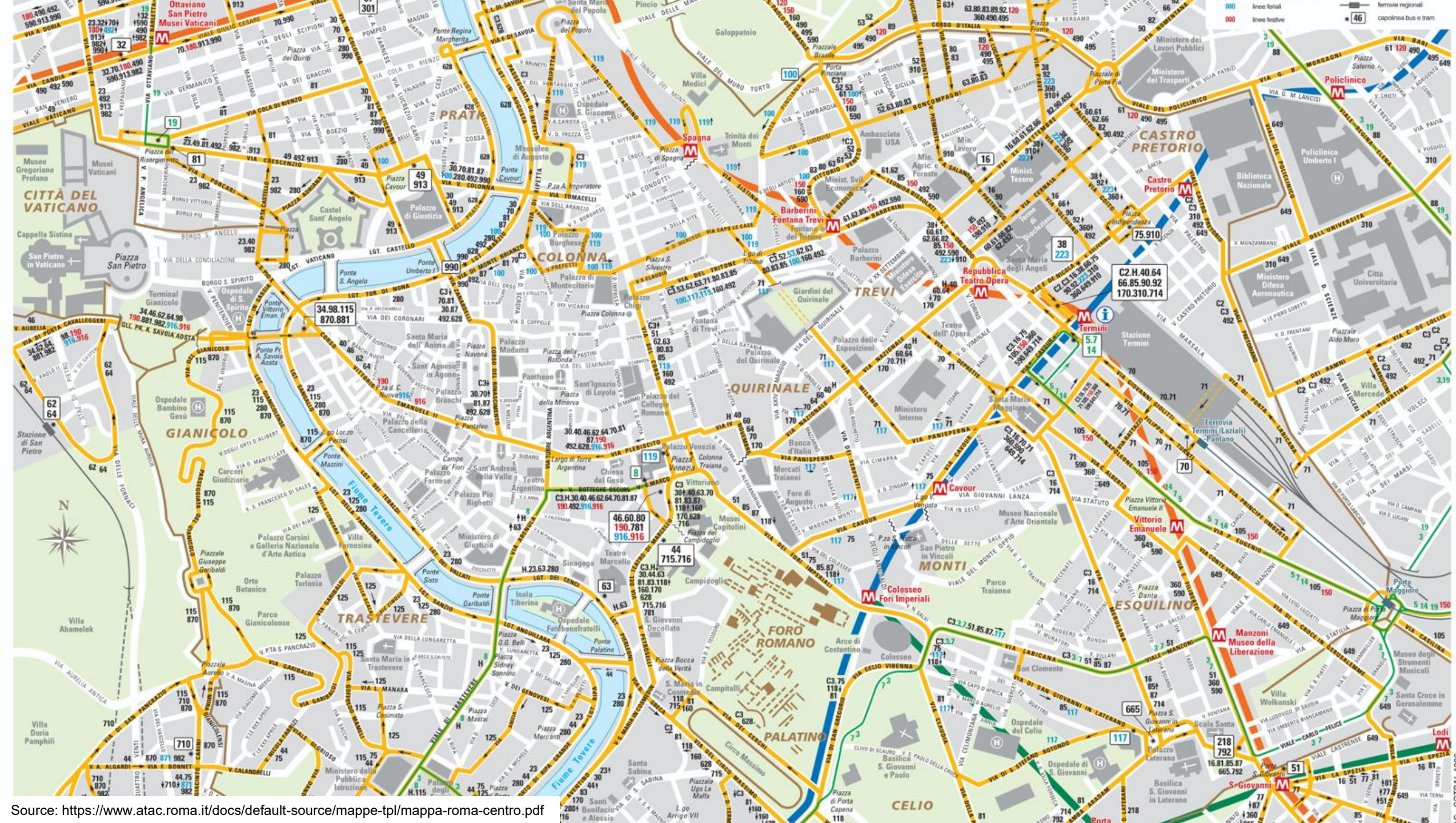
<https://linktr.ee/oren>

THINGS TO DO TODAY

Date 1993 COMPLETED

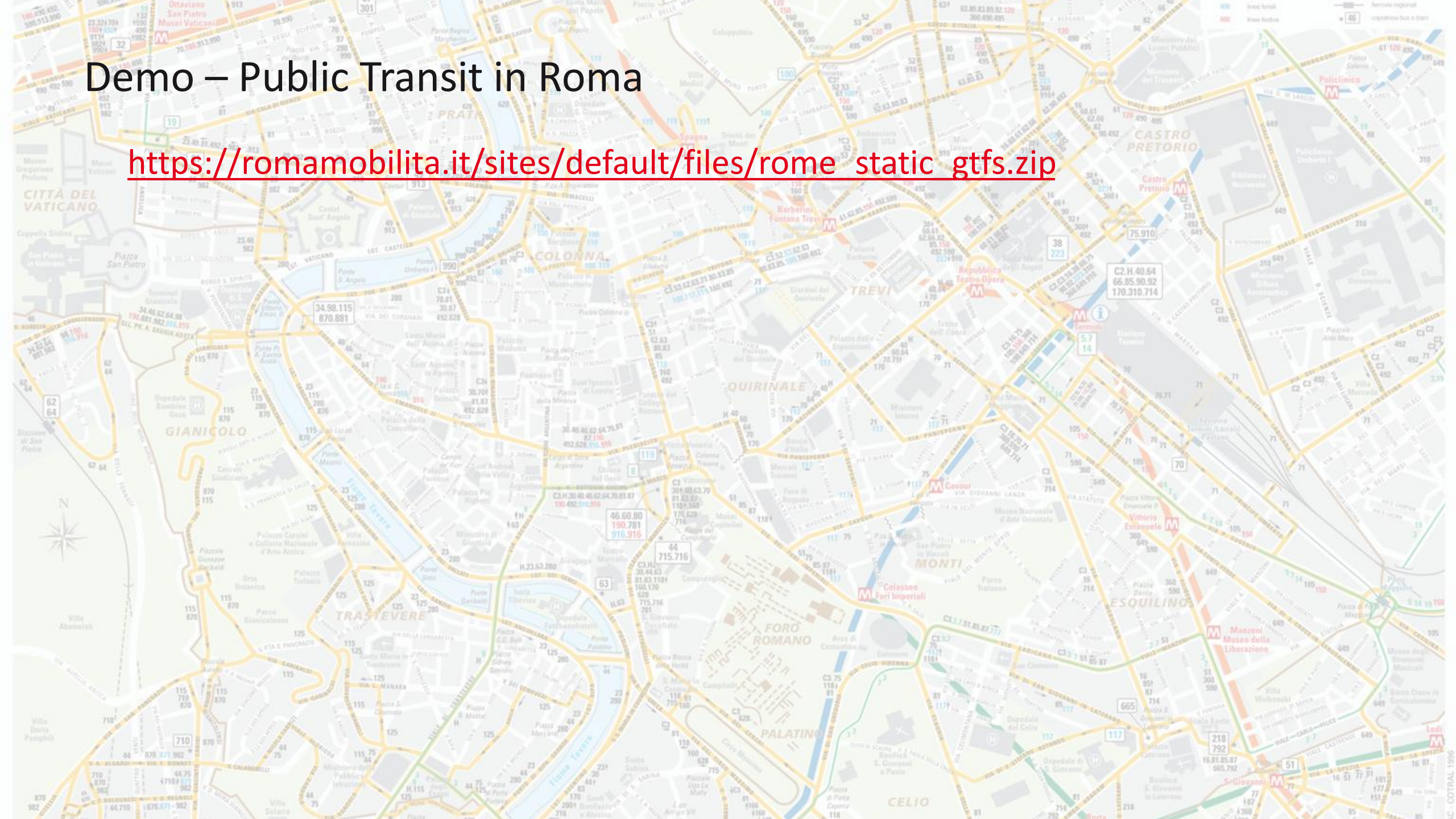
- 1) ☐
- 2) Start developing ☐
- 3) ☐
- 4) in ORACLE6 + ☐
- 5) SQL*Forms 3.0 ☐
- 6) +Oracle*CASE ☐
- 7) ☐
- 8) ☐
- 9) 5.0 ☐
- 10) ☐



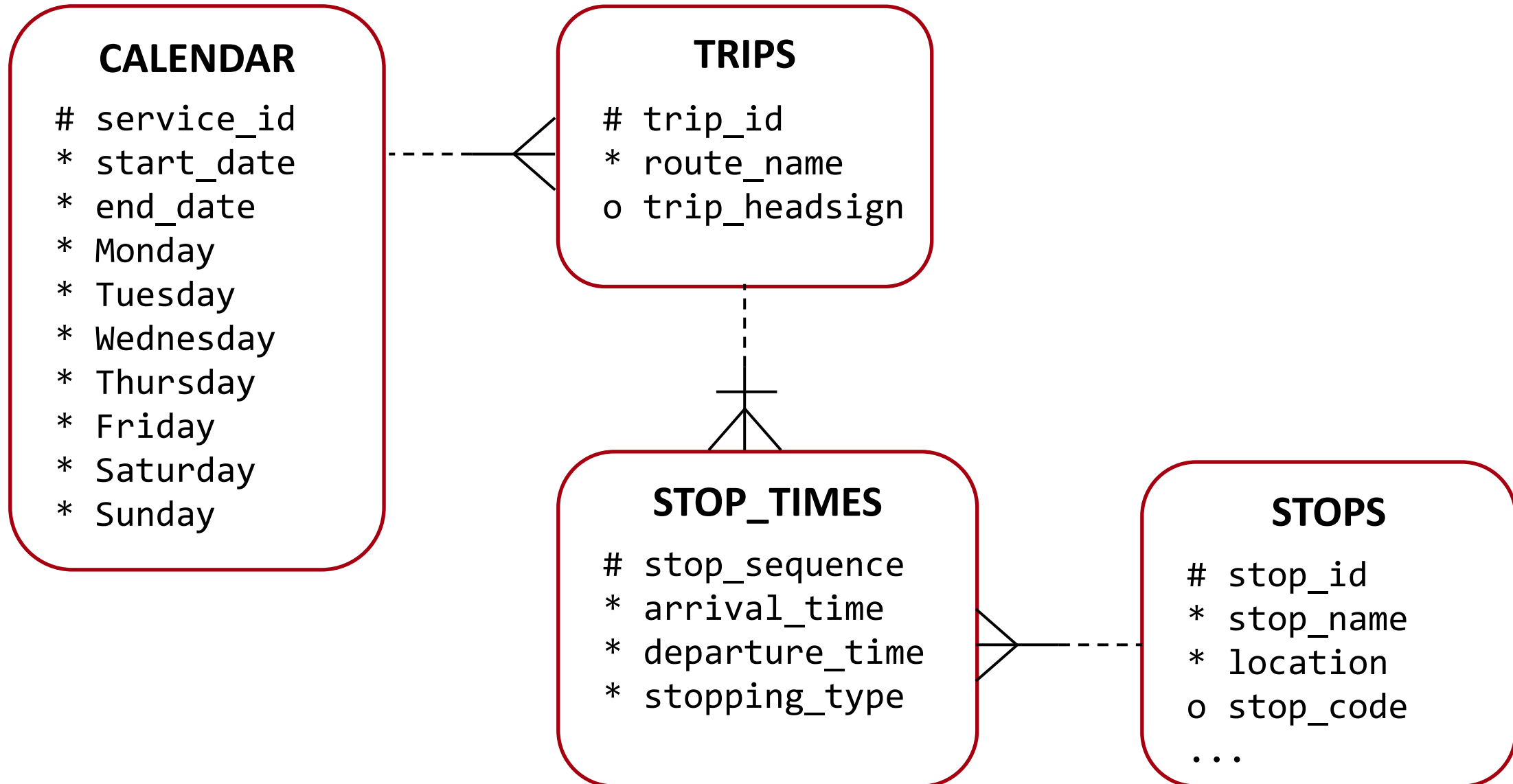


Demo – Public Transit in Roma

https://romamobilita.it/sites/default/files/rome_static_gtfs.zip



Demo – Data Model



```
spool install.log
```

```
drop table stop_times;  
drop table stops;  
drop table trips;  
drop table calendar;  
drop sequence stops_seq;
```

```
create table calendar (  
    service_id number(10) not null constraint calendar_pk primary key,
```

```
.  
. .
```

```
create table trips (  
    .
```

```
. .  
. .
```

```
spool off
```

```
C:\WLWM\scripts>find /i "ora-" install.log
```

```
----- INSTALL.LOG
```

```
ORA-00942: table or view does not exist
```

```
ORA-00942: table or view does not exist
```

```
spool install.log
```

```
drop table stop_times;  
drop table stops;  
drop table trips;  
drop table calendar;  
drop sequence stops_seq;
```

```
create table calendar (  
    service_id number(10) not null constraint calendar_pk primary key,
```

```
    .  
    .  
    .
```

```
create table trips (  
    .
```

```
    .  
    .  
    .
```

```
spool off
```

```
C:\WLWM\scripts>find /i "ora-" install.log
```

```
----- INSTALL.LOG
```

```
ORA-00942: table or view "DEMO_WLWM"."STOPS" does not exist
```

```
ORA-00942: table or view "DEMO_WLWM"."TRIPS" does not exist
```



26ai

```
spool install.log
```

```
drop table if exists stop_times;  
drop table if exists stops;  
drop table if exists trips;  
drop table if exists calendar;  
drop sequence if exists stops_seq;
```

```
create table calendar (  
    service_id number(10) not null constraint calendar_pk primary key,
```

```
.  
. .
```

```
create table trips (  
    .
```

```
.  
. .
```

```
spool off
```

26ai

```
C:\WLWM\scripts>find /i "ora-" install.log
```

```
----- INSTALL.LOG
```

```
C:\WLWM\scripts>
```

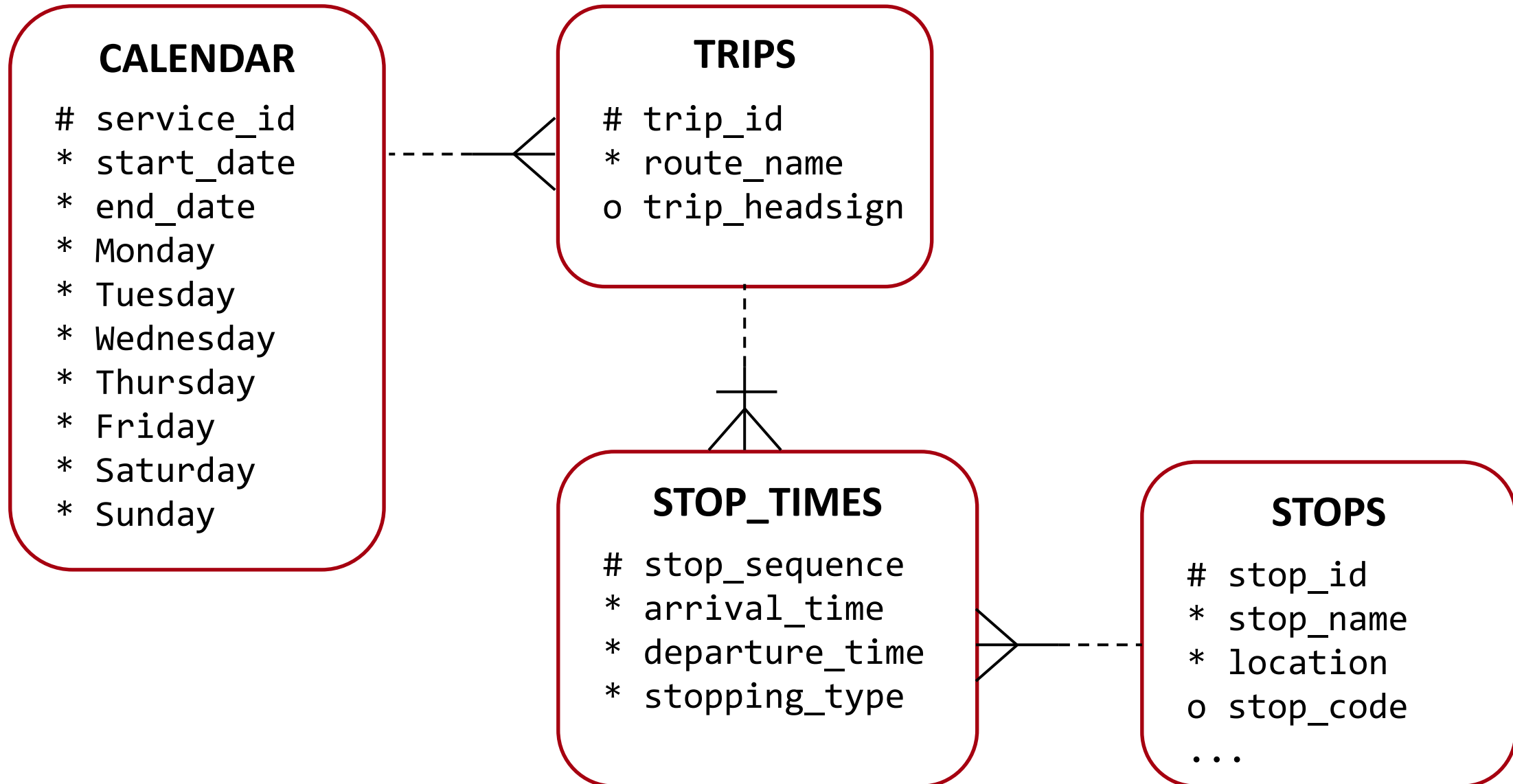
@oerr

SQL*Plus Enhancements



- SET ERROREDETAILS
- OERR
- And more...

Demo – Data Model



CALENDAR

```
# service_id
* start_date
* end_date
* Monday
* Tuesday
* Wednesday
* Thursday
* Friday
* Saturday
* Sunday
```

TRIPS

```
# trip_id
* route_name
o trip_headsign
```

STOP_TIMES

```
# stop_sequence
* arrival_time
* departure_time
* stopping_type
```

STOPS

```
# stop_id
* stop_name
* location
o stop_code
...
```



```
# service_id 123
* start_date 1/1/2026
* end_date    31/3/2026
* Monday      ✓
* Tuesday     ✓
* Wednesday   ✓
* Thursday    ✗
* Friday      ✓
* Saturday    ✗
* Sunday      ✗
```

```
create table calendar (  
    service_id number(10) not null constraint calendar_pk primary key,  
    --  
    monday      number(1) not null,  
    tuesday     number(1) not null,  
    wednesday   number(1) not null,  
    thursday    number(1) not null,  
    friday      number(1) not null,  
    saturday    number(1) not null,  
    sunday      number(1) not null,  
    --  
    start_date  date not null,  
    end_date    date not null  
);
```

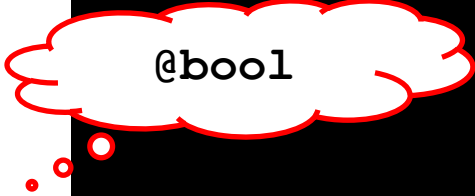
```
create table calendar (  
    service_id number(10) not null constraint calendar_pk primary key,  
    --  
    monday      number(1) not null,  
    tuesday     number(1) not null,  
    wednesday   number(1) not null,  
    thursday    number(1) not null,  
    friday      number(1) not null,  
    saturday    number(1) not null,  
    sunday      number(1) not null,  
    --  
    start_date  date not null,  
    end_date    date not null  
);
```

```
create table calendar (  
    service_id number(10) not null constraint calendar_pk primary key,  
    --  
    monday      number(1) not null check (monday in (0,1)),  
    tuesday     number(1) not null check (tuesday in (0,1)),  
    wednesday   number(1) not null check (wednesday in (0,1)),  
    thursday    number(1) not null check (thursday in (0,1)),  
    friday       number(1) not null check (friday in (0,1)),  
    saturday    number(1) not null check (saturday in (0,1)),  
    sunday      number(1) not null check (sunday in (0,1)),  
    --  
    start_date  date not null,  
    end_date    date not null  
);
```

```
create table calendar (  
    service_id number(10) not null constraint calendar_pk primary key,  
    --  
    monday      boolean not null,  
    tuesday     boolean not null,  
    wednesday   boolean not null,  
    thursday    boolean not null,  
    friday      boolean not null,  
    saturday    boolean not null,  
    sunday      boolean not null,  
    --  
    start_date  date not null,  
    end_date    date not null  
);
```



26ai



@bool

```
create table calendar (  
    service_id number(10) not null constraint calendar_pk primary key,  
    --  
    monday      boolean not null,  
    tuesday     boolean not null,  
    wednesday   boolean not null,  
    thursday    boolean not null,  
    friday      boolean not null,  
    saturday    boolean not null,  
    sunday      boolean not null,  
    --  
    start_date  date not null,  
    end_date    date not null  
);
```

```
create table calendar (  
    service_id number(10) not null constraint calendar_pk primary key,  
    --  
    monday      boolean not null,  
    tuesday     boolean not null,  
    wednesday   boolean not null,  
    thursday    boolean not null,  
    friday      boolean not null,  
    saturday    boolean not null,  
    sunday      boolean not null,  
    --  
    start_date  date not null  
        constraint cal_start_chk check (start_date = trunc(start_date)),  
    end_date    date not null  
        constraint cal_end_chk   check (end_date = trunc(end_date))  
);
```

```
create table calendar (  
    service_id number(10) not null constraint calendar_pk primary key,  
    --  
    monday      boolean not null,  
    tuesday     boolean not null,  
    wednesday   boolean not null,  
    thursday    boolean not null,  
    friday      boolean not null,  
    saturday    boolean not null,  
    sunday      boolean not null,  
    --  
    start_date  date not null  
        constraint cal_start_chk check (start_date = trunc(start_date)),  
    end_date    date not null  
        constraint cal_end_chk   check (end_date = trunc(end_date))  
);
```

```
create domain date_with_no_time as  
    date  
    constraint date_with_no_time_chk check (  
        date_with_no_time = trunc(date_with_no_time)  
    );
```



26ai

```
create table calendar (  
    service_id number(10) not null constraint calendar_pk primary key,  
    --  
    monday      boolean not null,  
    tuesday     boolean not null,  
    wednesday   boolean not null,  
    thursday    boolean not null,  
    friday      boolean not null,  
    saturday    boolean not null,  
    sunday      boolean not null,  
    --  
    start_date   date_with_no_time not null,  
    end_date     domain date_with_no_time not null  
);
```

```
create domain date_with_no_time as  
date  
constraint date_with_no_time_chk check (  
    date_with_no_time = trunc(date_with_no_time)  
);
```

@domain

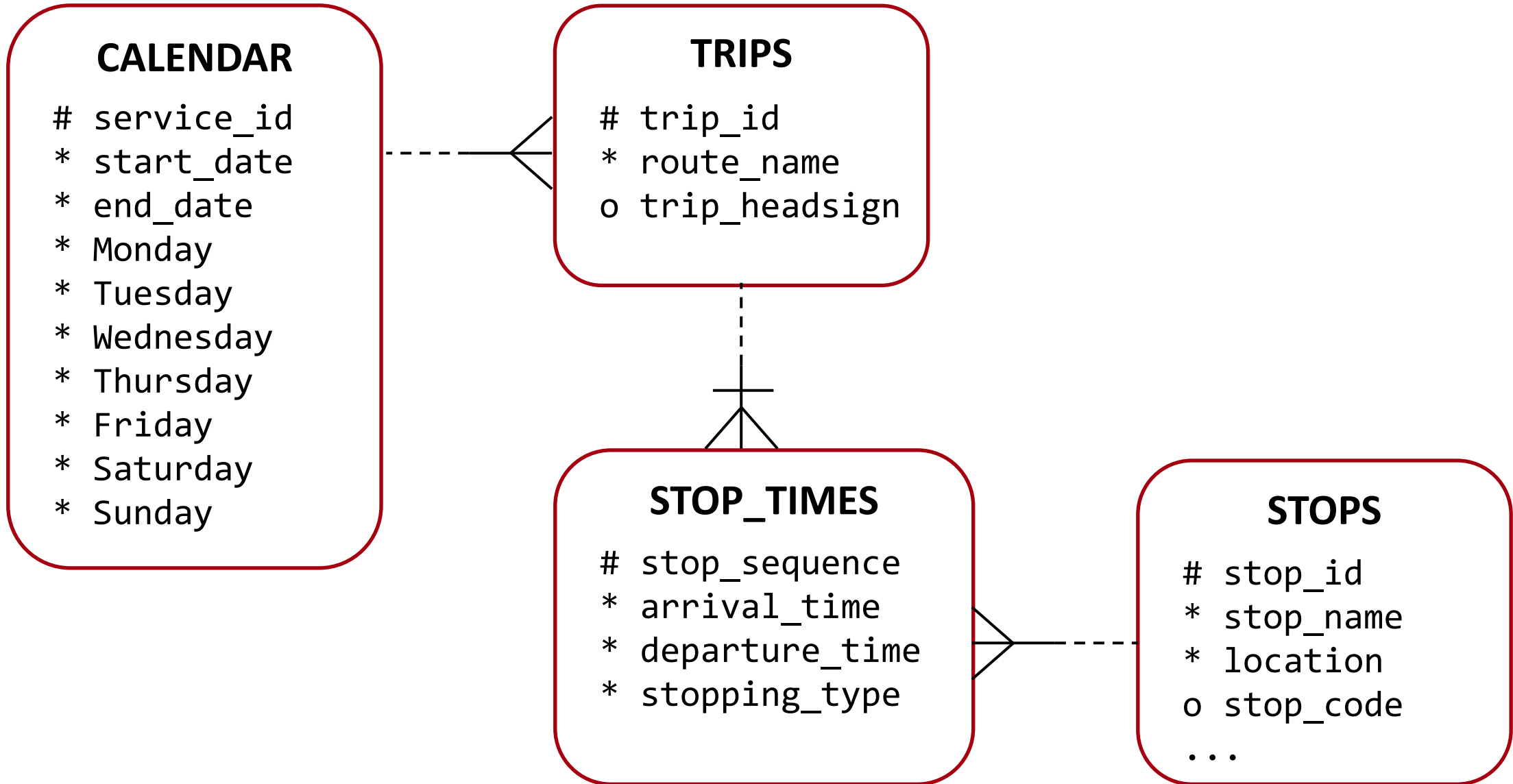
```

create domain "SYS"."CREDIT_CARD_NUMBER_D" as
  varchar2(20)
  check (regexp_like(credit_card_number_d, '^[\0-9]{12,19}$'))
  display case when length(credit_card_number_d)=15
    then substr(credit_card_number_d,1,4) || ' ' ||
         substr(credit_card_number_d,5,6) || ' ' ||
         substr(credit_card_number_d,11,5)
  when length(credit_card_number_d)=16
    then substr(credit_card_number_d,1,4) || ' ' ||
         substr(credit_card_number_d,5,4) || ' ' ||
         substr(credit_card_number_d,9,4) || ' ' ||
         substr(credit_card_number_d,13,4)
  when length(credit_card_number_d)=14
    then substr(credit_card_number_d,1,4) || ' ' ||
         substr(credit_card_number_d,5,6) || ' ' ||
         substr(credit_card_number_d,11,4)
  else credit_card_number_d
end;

```



@domain2



CALENDAR

```
# service_id
* start_date
* end_date
* Monday
* Tuesday
* Wednesday
* Thursday
* Friday
* Saturday
* Sunday
```

TRIPS

```
# trip_id
* route_name
o trip_headsign
```

SQL> desc trips

Name	Null?	Type
TRIP_ID	NOT NULL	VARCHAR2(30)
ROUTE_NAME	NOT NULL	VARCHAR2(100)
SERVICE_ID	NOT NULL	NUMBER(10)
TRIP_HEADSIGN		VARCHAR2(100)

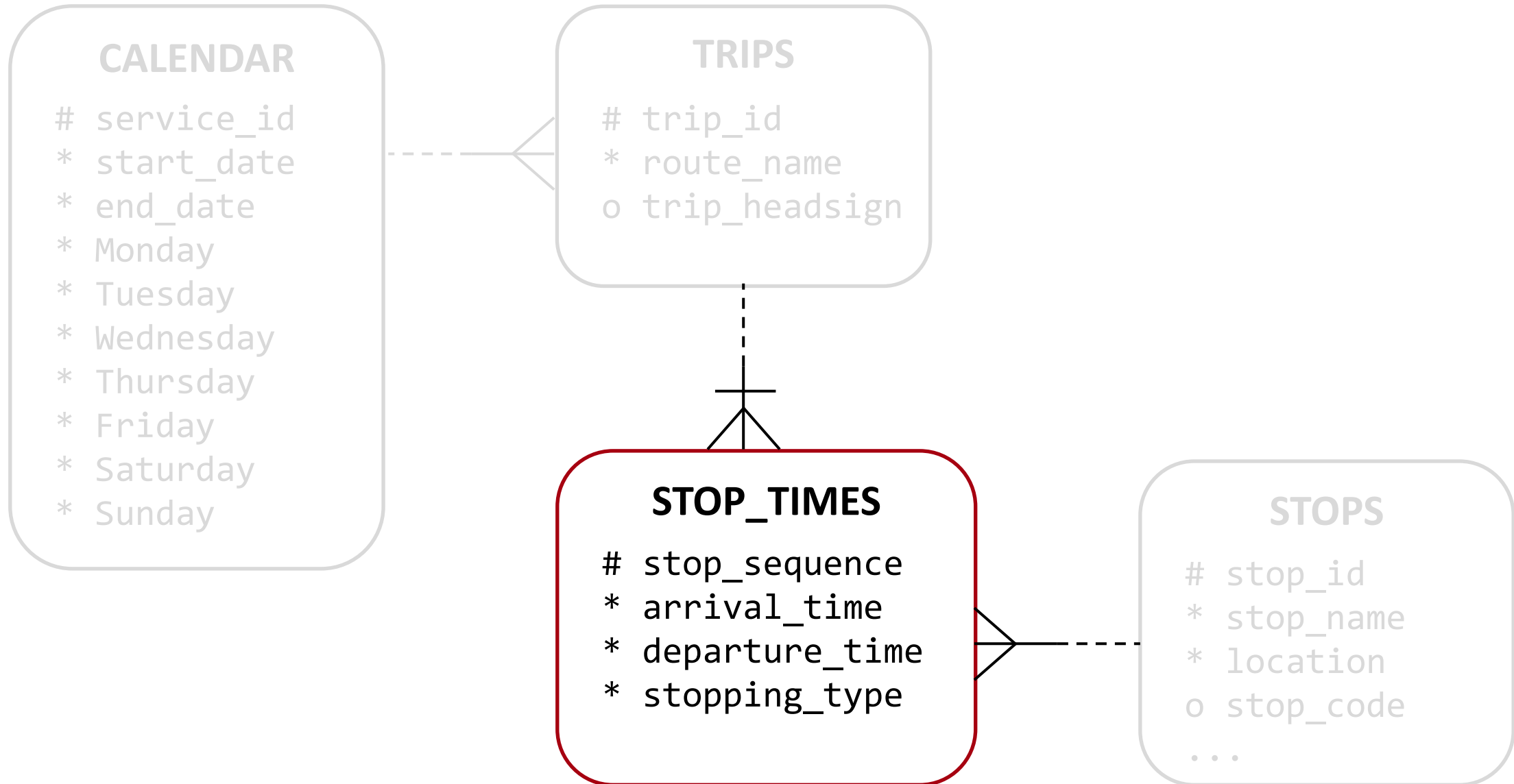
STOP TIMES

SQL> desc stops

Name	Null?	Type
STOP_ID	NOT NULL	NUMBER(10)
STOP_CODE		VARCHAR2(10)
STOP_NAME	NOT NULL	VARCHAR2(100)
LOCATION	NOT NULL	MDSYS.SDO_GEOMETRY

STOPS

```
# stop_id
* stop_name
* location
o stop_code
...
```



```
create table stop_times (  
    trip_id not null  
        constraint stop_times_fk_trips references trips(trip_id),  
    stop_sequence number(3) not null,  
    --  
    constraint stop_times_pk primary key (trip_id, stop_sequence),  
    --  
    arrival_time varchar2(8) not null,  
    departure_time varchar2(8) not null,  
    stop_id not null  
        constraint stop_times_fk_stops references stops(stop_id),  
    stopping_type varchar2(100) not null  
  
);
```

```
create table stop_times (  
    trip_id not null  
        constraint stop_times_fk_trips references trips(trip_id),  
    stop_sequence number(3) not null,  
    --  
    constraint stop_times_pk primary key (trip_id, stop_sequence),  
    --  
    arrival_time varchar2(8) not null,  
    departure_time varchar2(8) not null,  
    stop_id not null  
        constraint stop_times_fk_stops references stops(stop_id),  
    stopping_type varchar2(100) not null  
);
```



**Pickup
Dropoff
Both**

Constraints: Single Column in a Single Row Checks

```
create table stop_times (  
    trip_id not null  
        constraint stop_times_fk_trips references trips(trip_id),  
    stop_sequence number(3) not null,  
    --  
    constraint stop_times_pk primary key (trip_id, stop_sequence),  
    --  
    arrival_time varchar2(8) not null,  
    departure_time varchar2(8) not null,  
    stop_id not null  
        constraint stop_times_fk_stops references stops(stop_id),  
    stopping_type varchar2(100) not null  
        constraint stopping_type_chk check (  
            stopping_type in ('Pickup', 'Dropoff', 'Both'))  
);
```

```
create table stop_times (  
    trip_id not null  
        constraint stop_times_fk_trips references trips(trip_id),  
    stop_sequence number(3) not null,  
    --  
    constraint stop_times_pk primary key (trip_id, stop_sequence),  
    --  
    arrival_time varchar2(8) not null,  
    departure_time varchar2(8) not null,  
    stop_id not null  
        constraint stop_times_fk_stops references stops(stop_id),  
    stopping_type varchar2(100) not null  
        constraint stopping_type_chk check (  
            stopping_type in ('Pickup', 'Dropoff', 'Both'))  
);
```

**Departure cannot be
before arrival**

Constraints: Multiple Columns in a Single Row Checks

```
create table stop_times (  
    trip_id not null  
        constraint stop_times_fk_trips references trips(trip_id),  
    stop_sequence number(3) not null,  
    --  
    constraint stop_times_pk primary key (trip_id, stop_sequence),  
    --  
    arrival_time varchar2(8) not null,  
    departure_time varchar2(8) not null,  
    stop_id not null  
        constraint stop_times_fk_stops references stops(stop_id),  
    stopping_type varchar2(100) not null  
        constraint stopping_type_chk check (  
            stopping_type in ('Pickup', 'Dropoff', 'Both')),  
    --  
    constraint stop_times_arr_dep_chk check (departure_time >= arrival_time)  
);
```

Cross-Row Checks ???

TRIP_ID	STOP_SEQUENCE	ARRIVAL_TIME	DEPARTURE_TIME	STOP_ID
1#3-2	1	12:32:00	12:32:00	26205
1#3-2	2	12:37:00	12:38:00	26207
1#3-2	3	12:40:00	12:40:00	26209
1#3-2	4	12:39:00	12:46:30	26172
1#3-2	5	12:54:00	12:54:00	29232

Cross-Row Checks ???

TRIP_ID	STOP_SEQUENCE	ARRIVAL_TIME	DEPARTURE_TIME	STOP_ID
1#3-2	1	12:32:00	12:32:00	26205
1#3-2	2	12:37:00	12:38:00	26207
1#3-2	3	12:40:00	12:40:00	26209
1#3-2	4	12:39:00	12:46:30	26172
1#3-2	5	12:54:00	12:54:00	29232

Sorry, you cannot go back in time

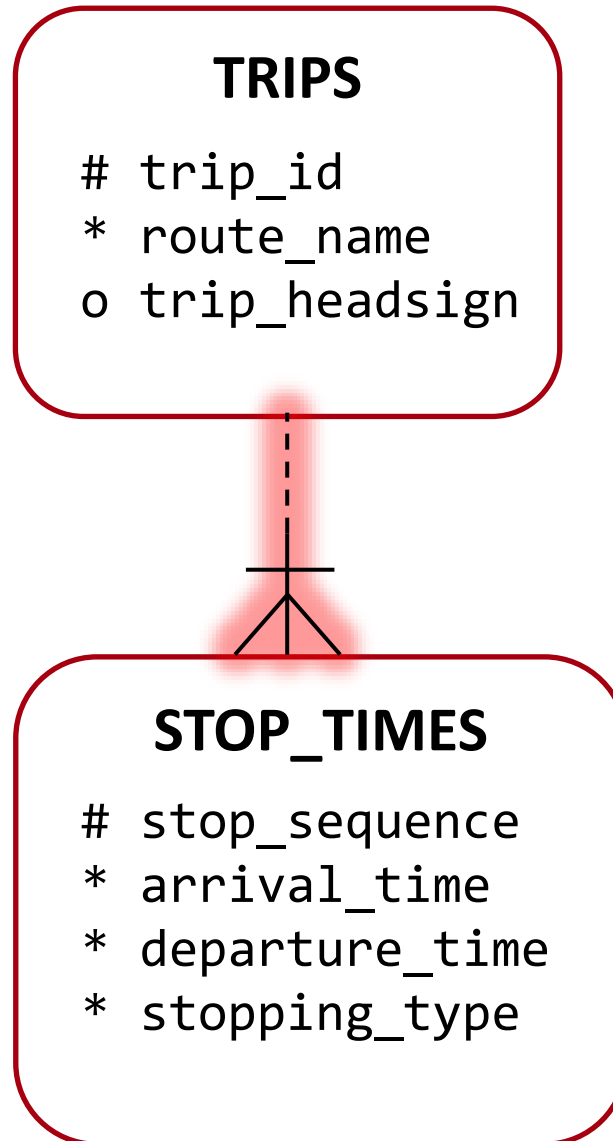
SQL Assertions: Cross-Row Checks



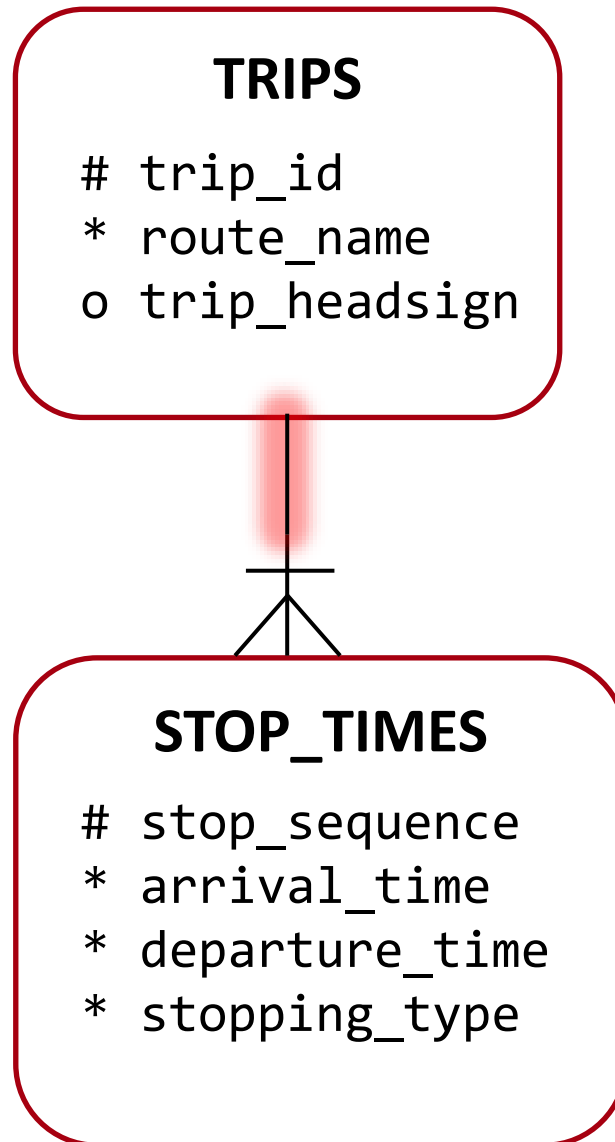
TRIP_ID	STOP_SEQUENCE	ARRIVAL_TIME	DEPARTURE_TIME	STOP_ID
1#3-2	1	12:32:00	12:32:00	26205
1#3-2	2	12:37:00	12:38:00	26207
1#3-2	3	12:40:00	12:40:00	26209
1#3-2	4	12:39:00	12:46:30	26172
1#3-2	5	12:54:00	12:54:00	29232

```
create assertion stop_times_not_going_back_in_time
check (not exists (
  select null
  from stop_times st1
  where exists (
    select null
    from stop_times st2
    where st2.trip_id = st1.trip_id
    and st2.stop_sequence > st1.stop_sequence
    and st2.arrival_time < st1.departure_time)));
```

Cross-Table Checks



Cross-Table Checks



SQL Assertions: Cross-Table Checks

26ai

```
create assertion trips_must_have_stop_times
  check (not exists (
    select null
    from trips t
    where not exists (
      select null
      from stop_times st
      where st.trip_id = t.trip_id)))

deferrable
initially deferred;
```

Existential Expression

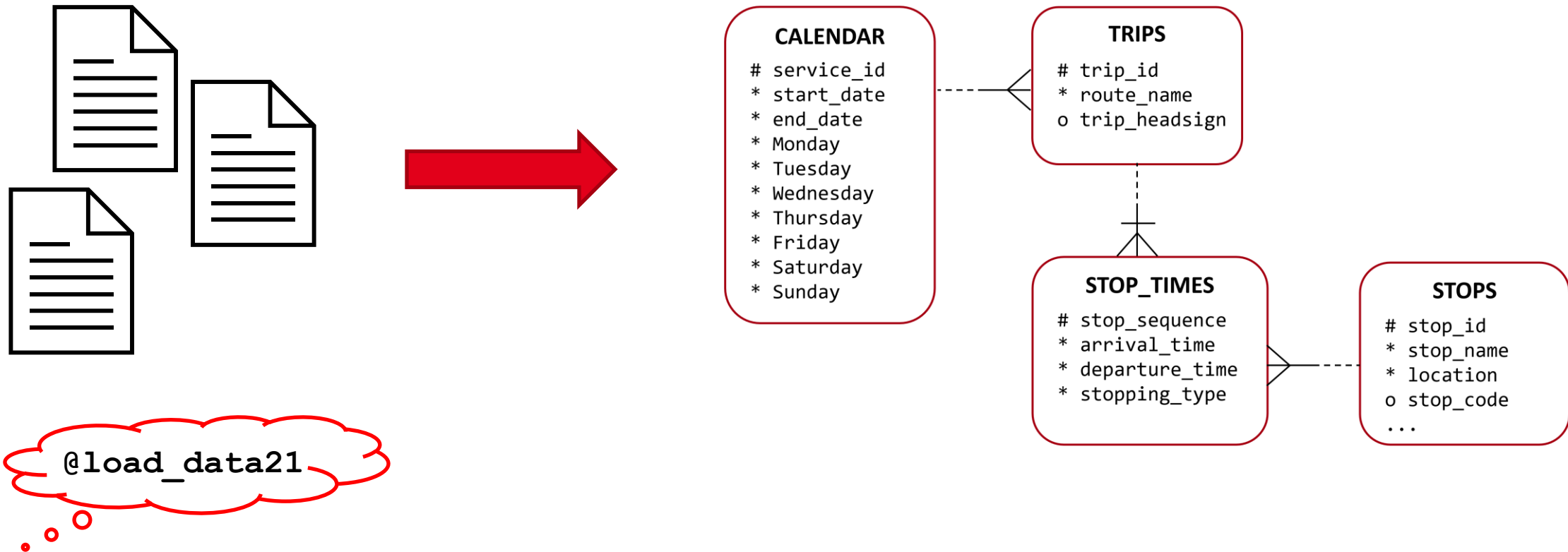
Universal Expression

@assert1

```
create assertion trips_must_have_stop_times2
  check (
    all (select * from trips) t
    satisfy (exists (
      select null
      from stop_times st
      where st.trip_id = t.trip_id)))

deferrable
initially deferred;
```

Reload (in a single transaction)



```
insert /*+ append */ into stops
(stop_code,
 stop_name,
 location )
select stop_code,
       stop_name,
       mdsys.sdo_geometry(2001,4326,mdsys.sdo_point_type(
                           stop_lon, stop_lat, null), null, null)
from   stops_x external modify (location ('stops_2.txt'));
```

ORA-12838: cannot read/modify an object after modifying it in parallel

```
insert into stop_times
(trip_id,stop_sequence,arrival_time,departure_time,stop_id)
select st.trip_id,
       st.stop_sequence,
       st.arrival_time,
       st.departure_time,
       s.stop_id
from stop_times_x st,
     stops s
where s.stop_code=st.stop_id;
```

ORA-12838: cannot read/modify an object after modifying it in parallel

```
insert /*+ append */ into stops
(stop_code,
 stop_name,
 location )
select stop_code,
 stop_name,
 mdsys.sdo_geometry(2001,4326,mdsys.sdo_point_type(
 stop_lon, stop_lat, null), null, null)
from stops_x external modify (location ('stops_2.txt'));
```

Unrestricted direct loads

26ai

Unrestricted parallel DML

@load_data

```
insert into stop_times
(trip_id,stop_sequence,arrival_time,departure_time,stop_id)
select st.trip_id,
 st.stop_sequence,
 st.arrival_time,
 st.departure_time,
 s.stop_id
from stop_times_x st,
 stops s
where s.stop_code=st.stop_id;
```

```
select c.*  
from   calendar c  
where  trunc(sysdate) between c.start_date and c.end_date  
and    decode(to_char(sysdate, 'fmDay', 'nls_date_language=english'),  
              'Sunday',      c.sunday,  
              'Monday',      c.monday,  
              'Tuesday',     c.tuesday,  
              'Wednesday',   c.wednesday,  
              'Thursday',    c.thursday,  
              'Friday',      c.friday,  
              'Saturday',    c.saturday);
```

```
select c.*  
from   calendar c  
where  trunc(sysdate) between c.start_date and c.end_date  
and    decode(to_char(sysdate, 'fmDay', 'nls_date_language=english'),  
              'Sunday',    c.sunday,  
              'Monday',    c.monday,  
              'Tuesday',   c.tuesday,  
              'Wednesday', c.wednesday,  
              'Thursday',  c.thursday,  
              'Friday',    c.friday,  
              'Saturday',  c.saturday);
```

```
select c.*
from   calendar c
where  trunc(sysdate) between c.start_date and c.end_date
and    decode(to_char(sysdate, 'fmDay', 'nls_date_language=english'),
              'Sunday',    c.sunday,
              'Monday',    c.monday,
              'Tuesday',   c.tuesday,
              'Wednesday', c.wednesday,
              'Thursday',  c.thursday,
              'Friday',    c.friday,
              'Saturday',  c.saturday);
```

```
select c.*
from   calendar c
where  trunc(sysdate+1) between c.start_date and c.end_date
and    decode(to_char(sysdate+1, 'fmDay', 'nls_date_language=english'),
              'Sunday',    c.sunday,
              'Monday',    c.monday,
              'Tuesday',   c.tuesday,
              'Wednesday', c.wednesday,
              'Thursday',  c.thursday,
              'Friday',    c.friday,
              'Saturday',  c.saturday);
```

```
select c.*
from   calendar c
where  trunc(date'2026-08-29') between c.start_date and c.end_date
and    decode(to_char(date'2026-08-29', 'fmDay', 'nls_date_language=english'),
              'Sunday',    c.sunday,
              'Monday',    c.monday,
              'Tuesday',   c.tuesday,
              'Wednesday', c.wednesday,
              'Thursday',  c.thursday,
              'Friday',    c.friday,
              'Saturday',  c.saturday);
```

Reusable Queries



Views

Reusable Queries
+
Different Inputs



Views

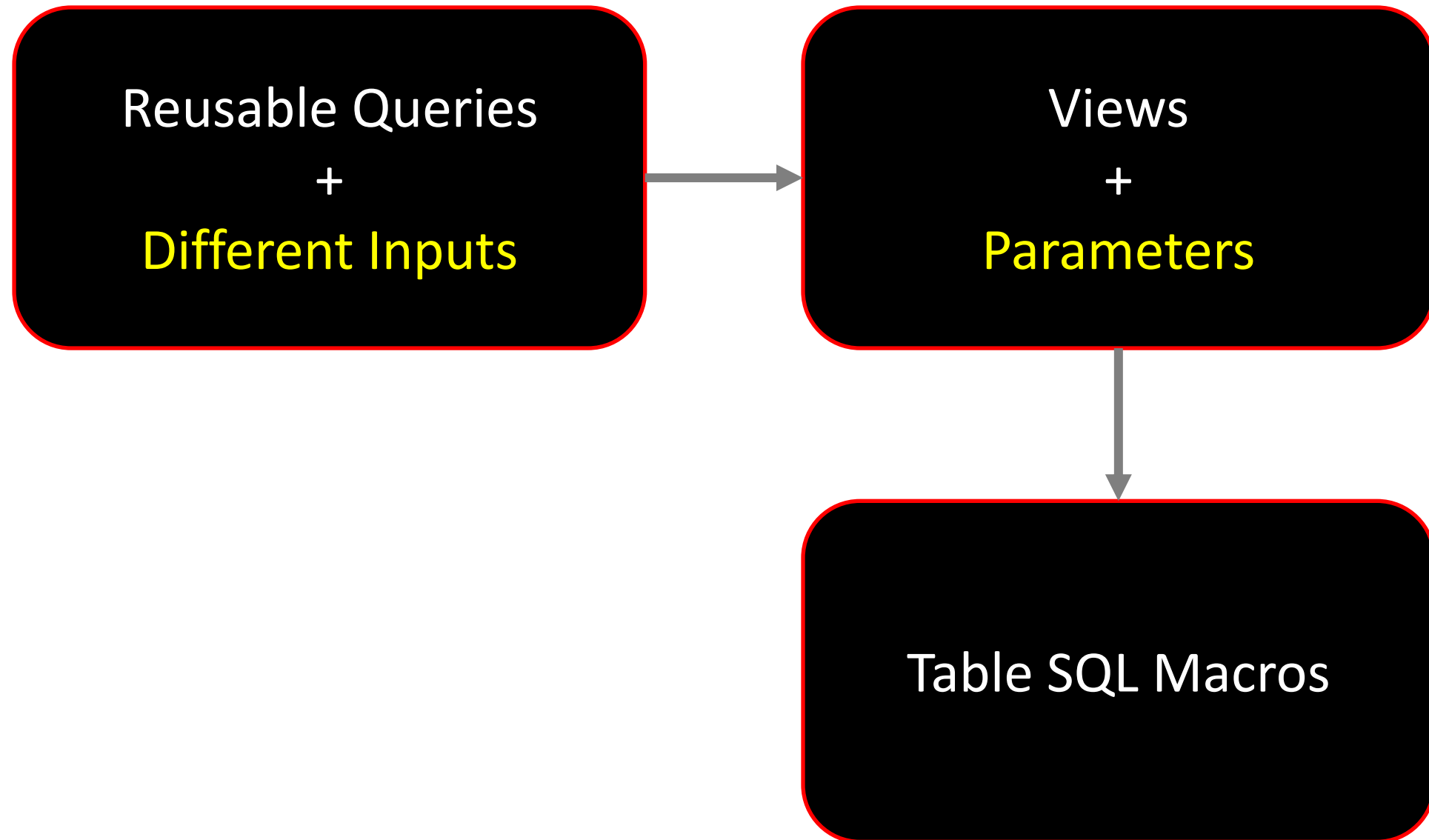


Table SQL Macro – Parameterized Views



21c



19.7

```
create or replace package body gtfs_sm as

    function services(i_date in date default sysdate)
        return varchar2 sql_macro(table) is
    begin
        return q'<
select c.service_id
from   calendar c
where  trunc(i_date) between c.start_date and c.end_date
and    decode(to_char(i_date, 'fmDay', 'nls_date_language=english'),
              'Sunday',      c.sunday,
              'Monday',      c.monday,
              'Tuesday',     c.tuesday,
              'Wednesday',   c.wednesday,
              'Thursday',    c.thursday,
              'Friday',      c.friday,
              'Saturday',    c.saturday)
>';
    end services;
end gtfs_sm;
```

Table SQL Macro – Parameterized Views

21c19.7

```
create or replace package body gtfs_sm as

    function services(i_date in date default sysdate)
        return varchar2 sql_macro(table) is
    begin
        return q'<
select c.service_id
from   calendar c
where  trunc(i_date) between c.start_date and c.end_date
and    decode(to_char(i_date, 'fmDay', 'nls_date_language=english'),
                'Sunday',    c.sunday,
                'Monday',    c.monday,
                'Tuesday',   c.tuesday,
                'Wednesday', c.wednesday,
                'Thursday',  c.thursday,
                'Friday',    c.friday,
                'Saturday',  c.saturday)
>';
        end services;
    end gtfs_sm;
```

Table SQL Macro – Parameterized Views

21c

19.7

```
create or replace package body gtfs_sm as

  function services(i_date in date default sysdate)
    return varchar2 sql_macro(table) is
  begin
    return q'<
select c.service_id
from   calendar c
where  trunc(i_date) between c.start_date and c.end_date
and    decode(to_char(i_date, 'fmDay', 'nls_date_language=english'),
              'Sunday',      c.sunday,
              'Monday',      c.monday,
              'Tuesday',     c.tuesday,
              'Wednesday',   c.wednesday,
              'Thursday',    c.thursday,
              'Friday',      c.friday,
              'Saturday',    c.saturday)
    end services;
end gtfs_sm;
```

@sm1

Better Error Messages



26ai

```
select route_name,count(*)  
from trips  
group by trip_id  
      *
```

ERROR at line 1:
ORA-00979: not a GROUP BY expression

```
select route_name,count(*)  
from trips  
group by trip_id  
      *
```

ERROR at line 1:
ORA-00979: "ROUTE_NAME": must appear in the GROUP
BY clause or be used in an aggregate function

ORA-03162: "T"."ROUTE_NAME": must appear in the GROUP BY clause or be used in an aggregate function as 'group_by_position_enabled' is FALSE

ORA-11534: check constraint (DEMO_WLWM.SYS_C008967) involving column START_DATE due to domain constraint DEMO_WLWM.DATE_WITH_NO_TIME_CHK of domain DEMO_WLWM.DATE_WITH_NO_TIME violated

ORA-00001: unique constraint (DEMO_WLWM.STOP_TIMES_PK) violated on table
DEMO_WLWM.STOP_TIMES columns
(TRIP_ID, STOP_SEQUENCE)

ORA-03301: (ORA-00001 details) row with column values (TRIP_ID:'itoug#1', STOP_SEQUENCE:3)
already exists

```

SELECT "A1"."ROUTE_NAME" "ROUTE_NAME",
       COUNT(*) "NUMBER_OF_TRIPS"
FROM   (SELECT "A3"."SERVICE_ID" "SERVICE_ID"
        FROM   (SELECT "A4"."SERVICE_ID" "SERVICE_ID"
                 FROM   "DEMO_WLWM"."CALENDAR" "A4"
                 WHERE  TRUNC(SYSDATE) >= "A4"."START_DATE"
                 AND    TRUNC(SYSDATE) <= "A4"."END_DATE"
                 AND    DECODE(TO_CHAR(SYSDATE, 'fmDay', 'nls_date_language=english'),
                              'Sunday', "A4"."SUNDAY",
                              'Monday', "A4"."MONDAY",
                              'Tuesday', "A4"."TUESDAY",
                              'Wednesday', "A4"."WEDNESDAY",
                              'Thursday', "A4"."THURSDAY",
                              'Friday', "A4"."FRIDAY",
                              'Saturday', "A4"."SATURDAY") = TRUE) "A3") "A2",
       "DEMO_WLWM"."TRIPS" "A1"
WHERE  "A1"."SERVICE_ID" = "A2"."SERVICE_ID"
GROUP BY "A1"."ROUTE_NAME"
ORDER BY COUNT(*)

```

```

select t.route_name,count(*) number_of_trips
from gtfs_sm.services(sysdate) s, trips t
where t.service_id = s.service_id
group by t.route_name
order by number_of_trips

```

THANK YOU 😊

Oren Nakdimon

www.db-oriented.com

✉ oren@db-oriented.com

X @DBoriented