



Из Oracle в Postgres и обратно

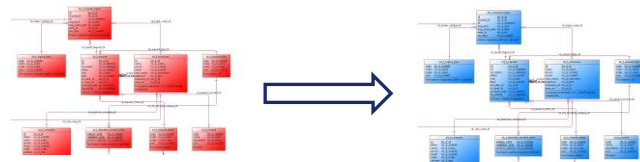
Александр Любушкин
Владимир Сенченко

Миграция структур данных

Миграция программ на языках СУБД

Миграция данных

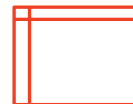
Миграция клиентских программ
и интерфейсов



PL/SQL



PL/pgSQL



Наши решения ранее
представленные на PgConf



Миграция структур данных

Миграция программ на языках СУБД

Миграция данных

Миграция клиентских программ
и интерфейсов

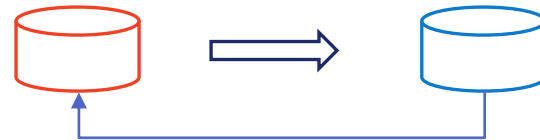


Миграция структур данных

Миграция программ на языках СУБД

Миграция данных

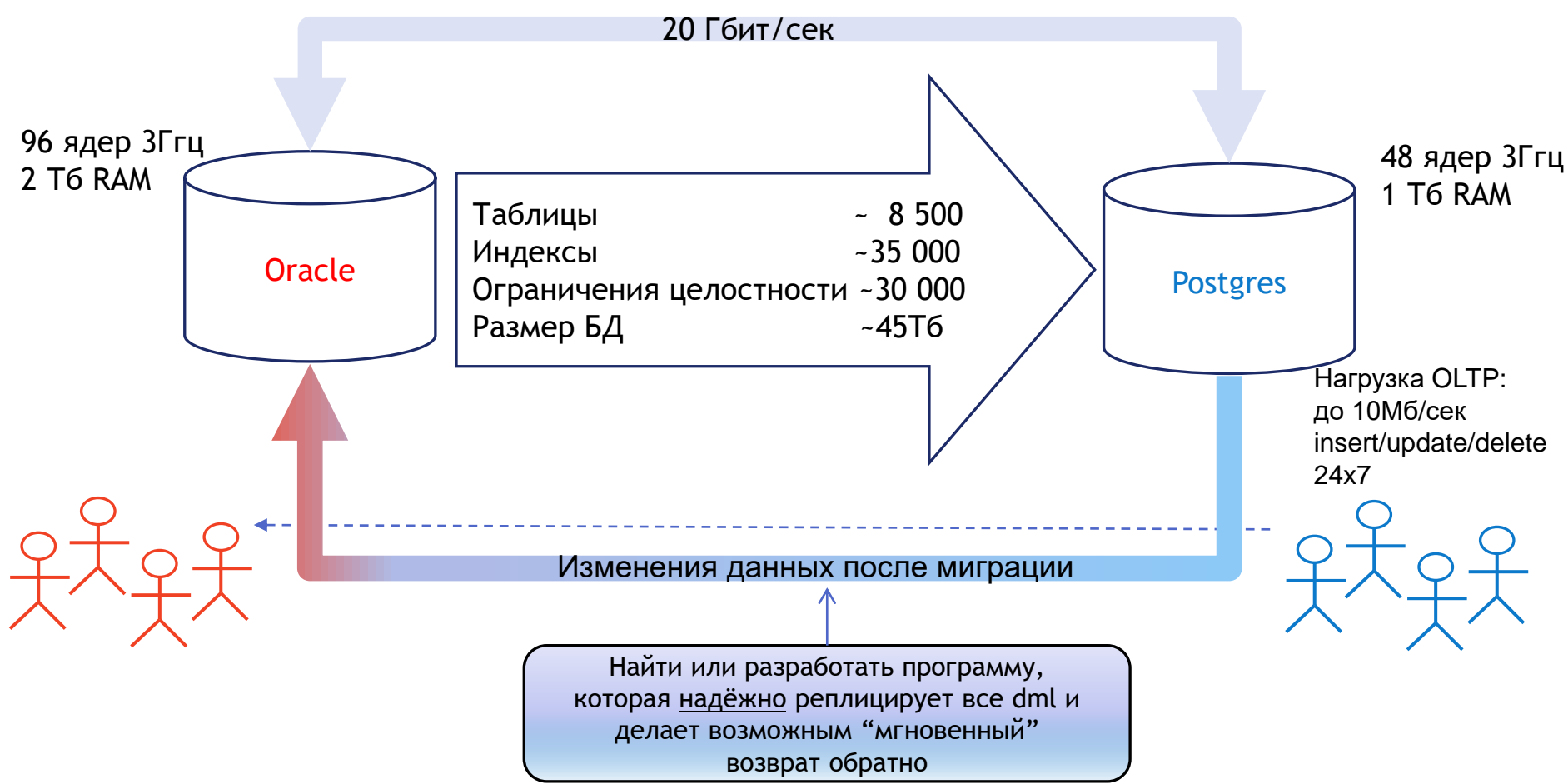
Миграция клиентских программ
и интерфейсов

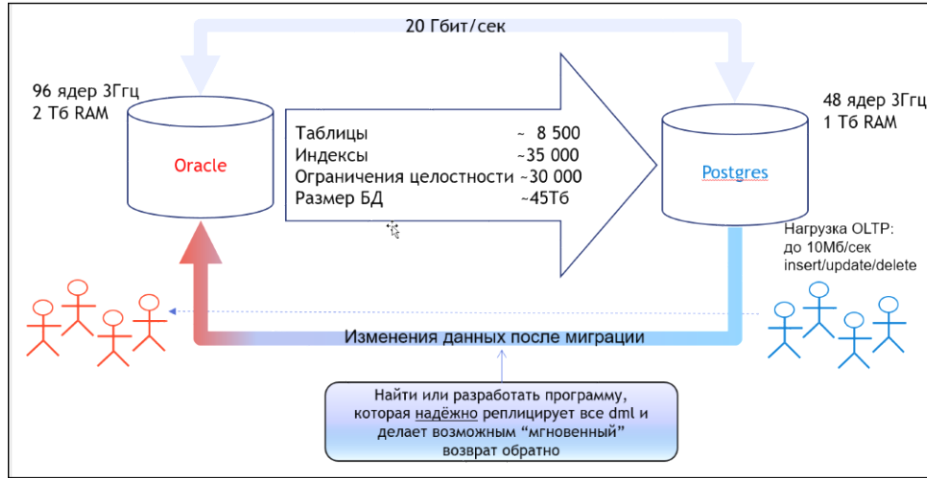


И вот, один уважаемый заказчик
захотел вот это...
Зачем ???

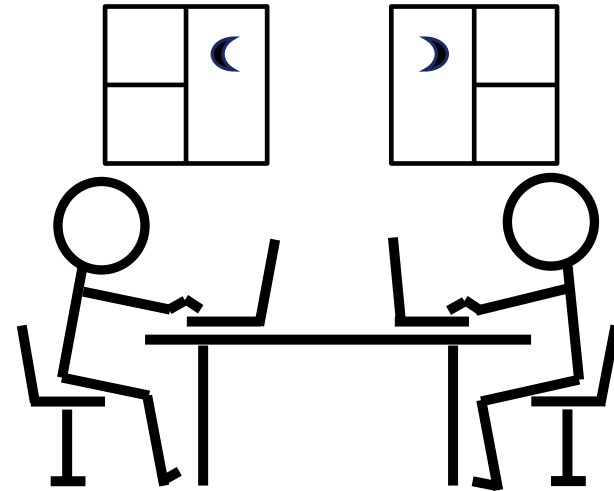


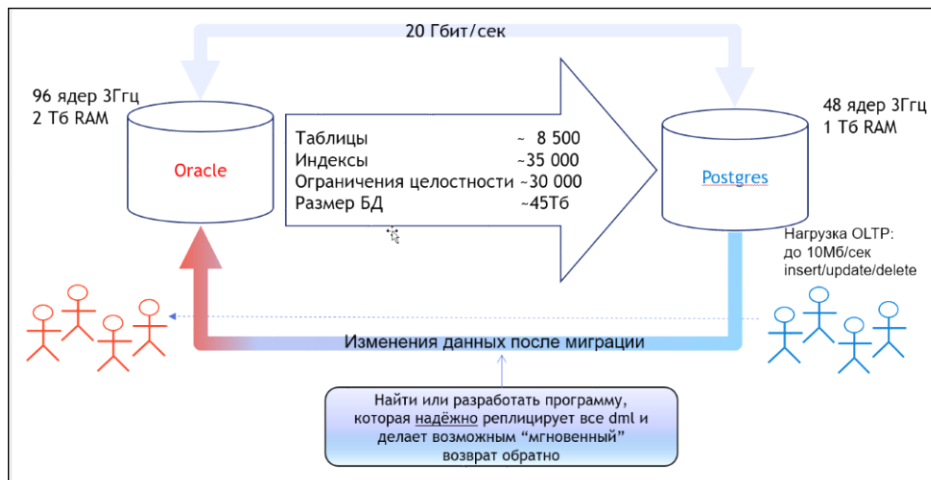




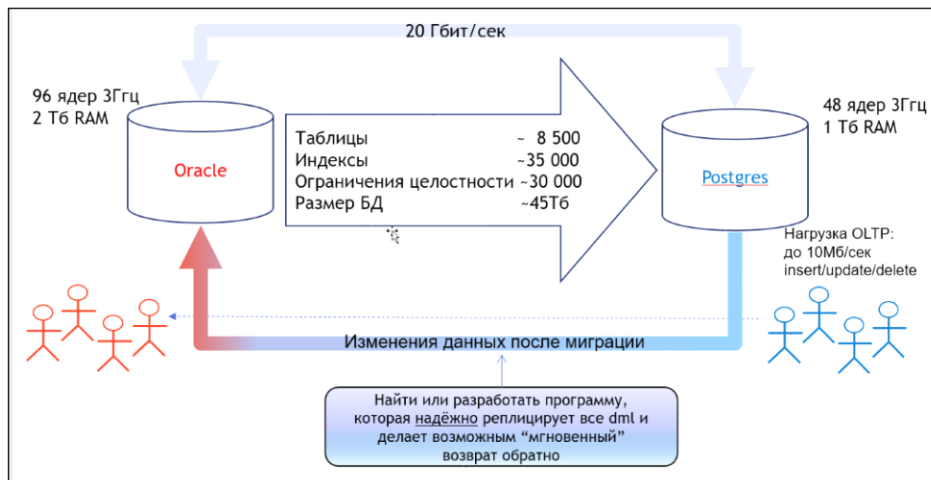


- 2 инженера (без опыта логической репликации из Postgres)
- 2 календарных месяца





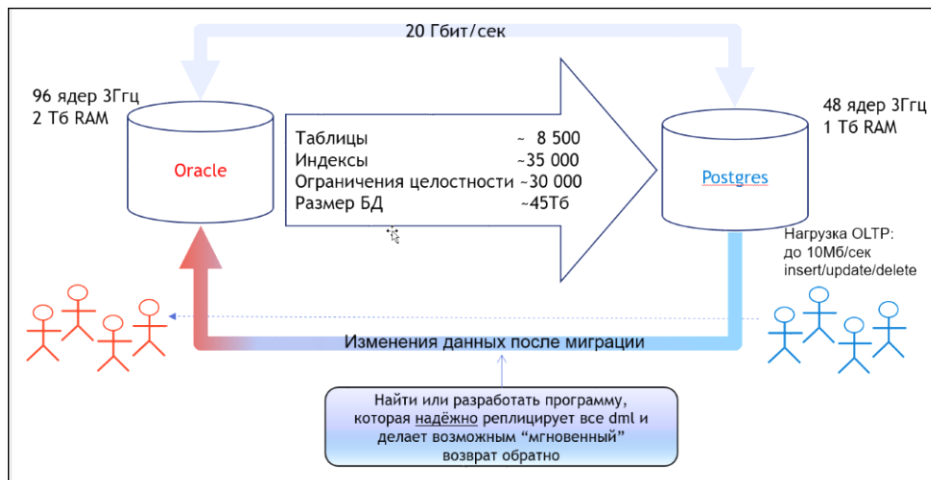
- Асимметрия схем
 - имена
 - состав столбцов
 - состав таблиц
 - типы данных
- Таблицы без ключей
- Таблицы со столбцом rowid
- Не конвенциональные имена
- Множество секционированных
- CLOB+BLOB>1Гб



- Асимметрия схем
 - имена
 - состав столбцов
 - состав таблиц
 - типы данных
- Таблицы без ключей
- Таблицы со столбцом rowid
- Не конвенциональные имена
- Множество секционированных
- CLOB+BLOB>1Гб

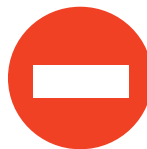
- Oracle Golden Gate
- Debezium
- Quest Shareplex





- Асимметрия схем
- - имена
- - состав столбцов
- - состав таблиц
- - типы данных
- Таблицы без ключей
- Таблицы со столбцом rowid
- Не конвенциональные имена
- Множество секционированных
- CLOB+BLOB > 1Гб

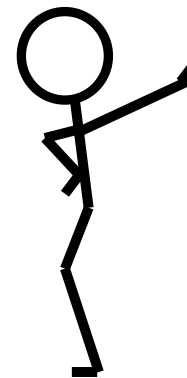
- Oracle Golden Gate
- Debezium
- Quest Shareplex



Java logical replication API:

<https://jdbc.postgresql.org/documentation/server-prepare/>

???

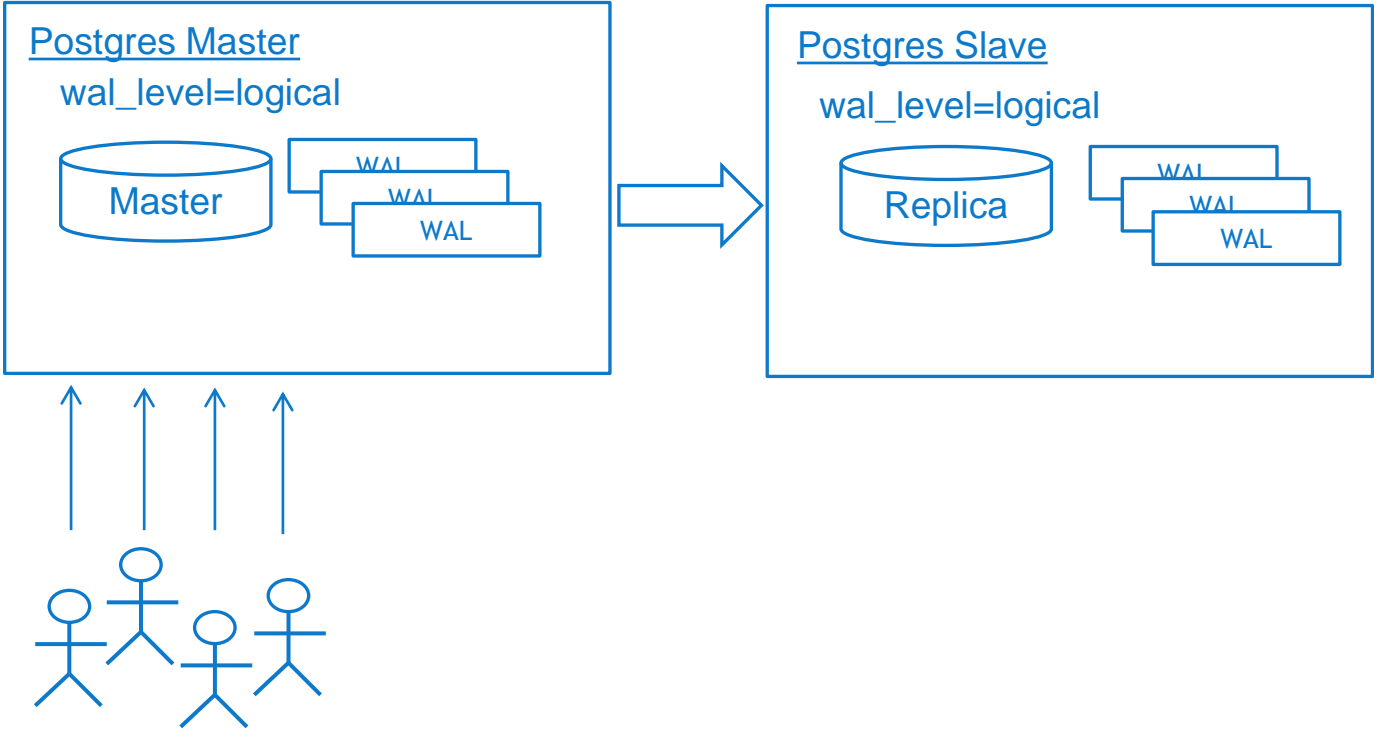


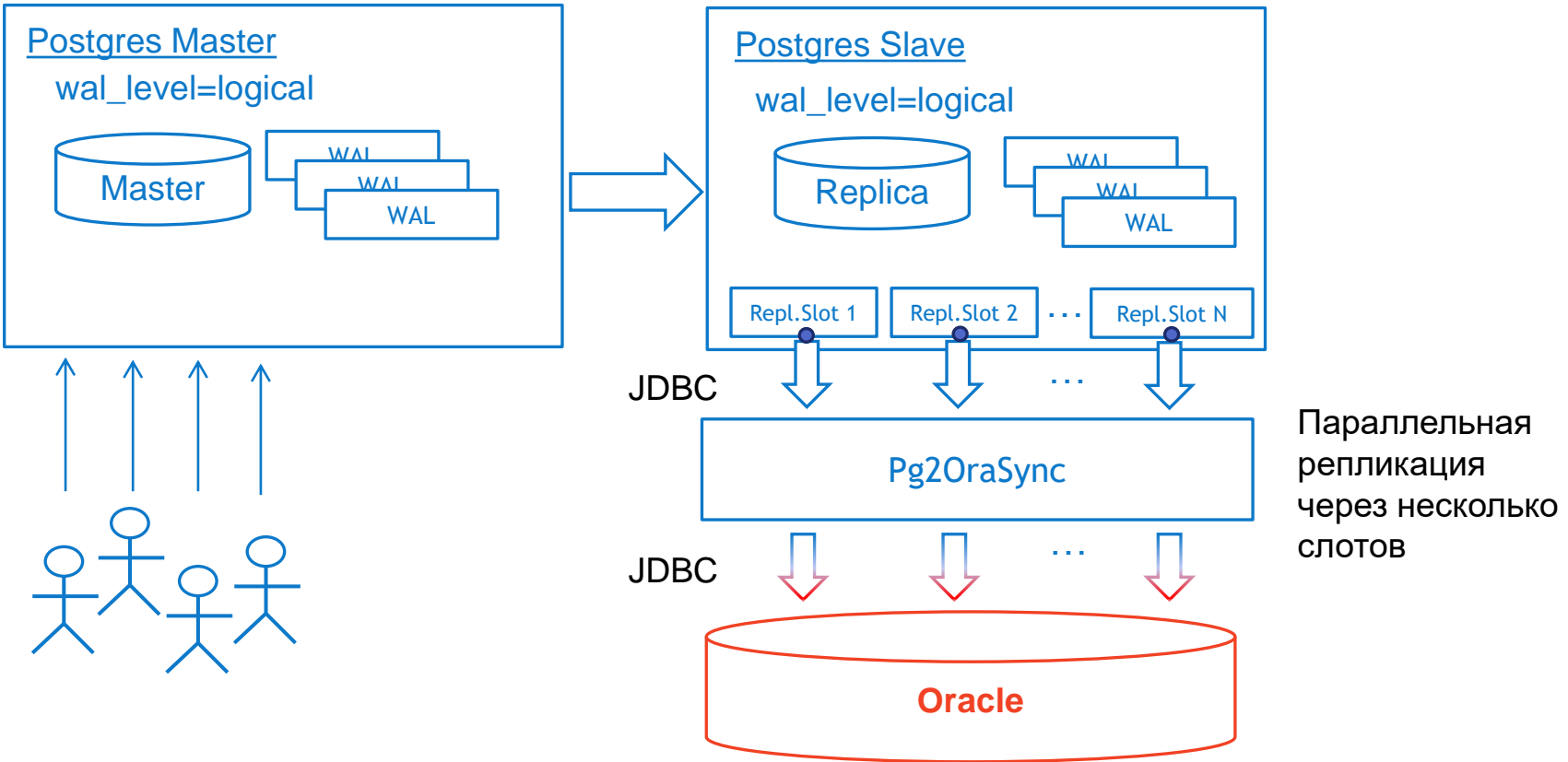
Example 9.14. Full example of logical replication

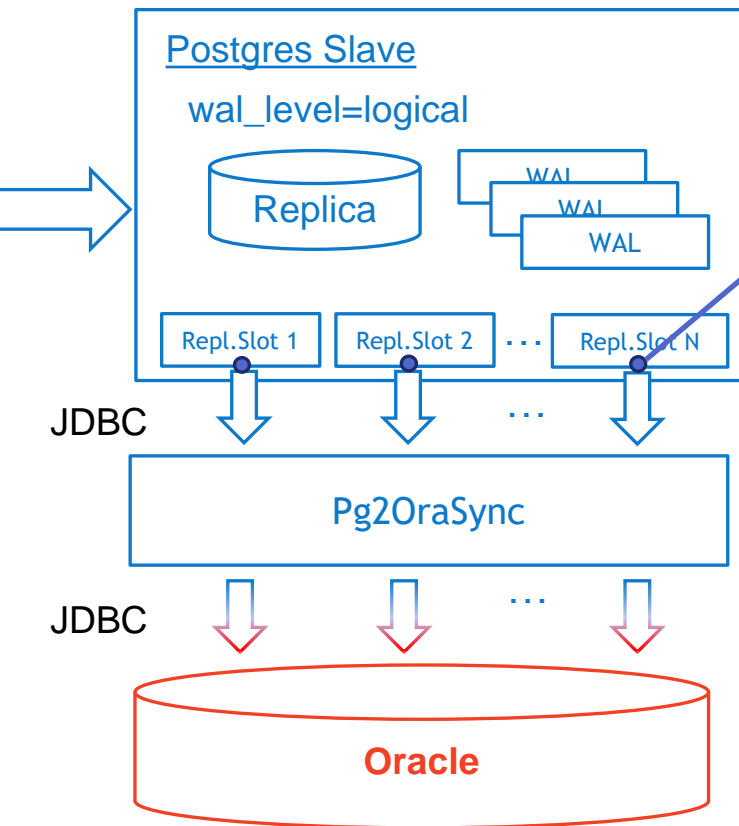
```
String url = "jdbc:postgresql://localhost:5432/test";
Properties props = new Properties();
PGProperty.USER.set(props, "postgres");
PGProperty.PASSWORD.set(props, "postgres");
PGProperty.ASSUME_MIN_SERVER_VERSION.set(props, "9.4");
PGProperty.REPLICATION.set(props, "database");
PGProperty.PREFER_QUERY_MODE.set(props, "simple");

Connection con = DriverManager.getConnection(url, props);
PGConnection replConnection = con.unwrap(PGConnection.class);

replConnection.getReplicationAPI()
    .createReplicationSlot()
    .logical()
    .withSlotName("demo_logical_slot")
    .withOutputPlugin("test_decoding")
    .make();
```







Plugin логического декодирования.

- postgres-decoderbufs <https://github.com/debezium/postgres-decoderbufs>
- wal2json github.com/eulerto/wal2json
- test_decoding
- **pgoutput**

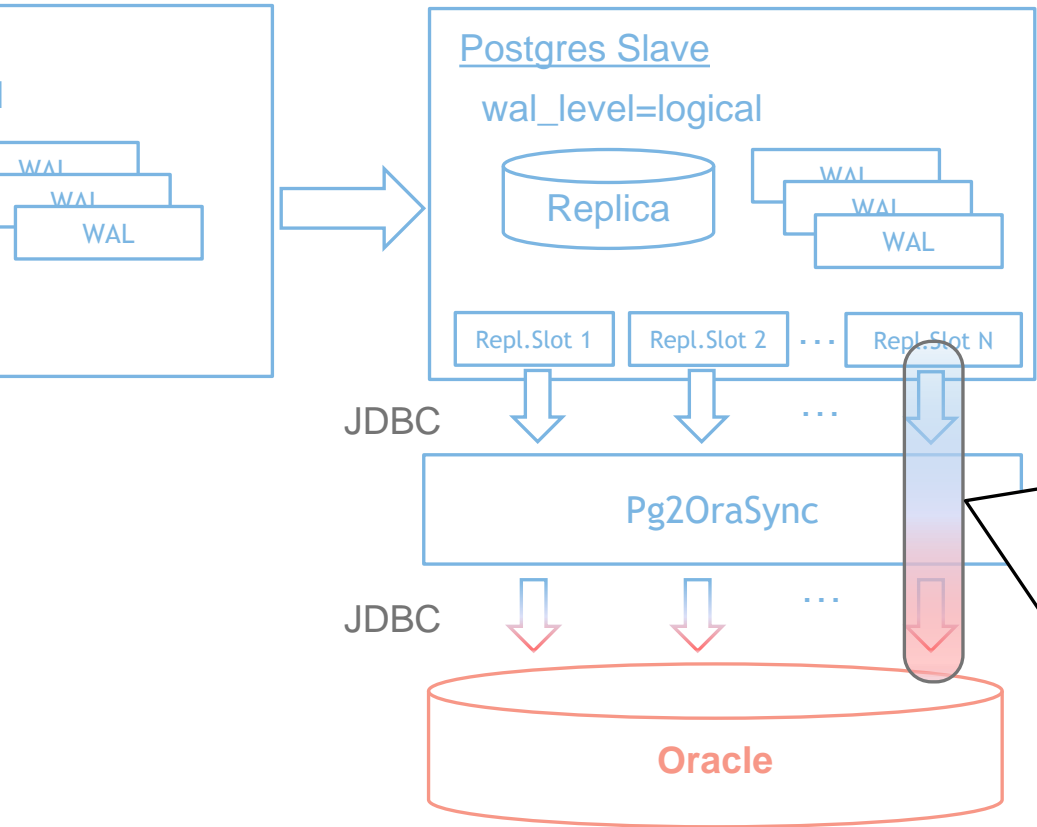
← Пред Наверх | Глава 57. Клиент-серверный протокол Начало След.

57.9. Форматы сообщений логической репликации

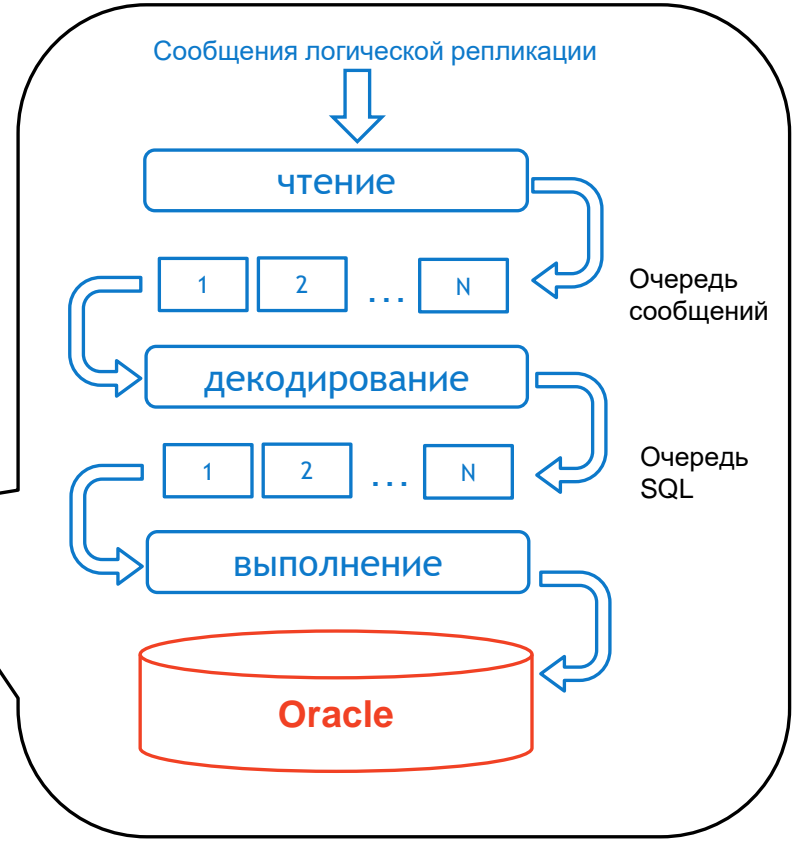
В этом разделе подробно описывается формат каждого сообщения логической репликации. Эти сообщения или выдвоятся через SQL-интерфейс слота репликации или передаются процессом walsender. Когда их передаёт walsender, они помещаются внутрь WAL-сообщений протокола репликации, описанных в Разделе 57.4, и в общем следуют тому же потоку сообщений, что и сообщения физической репликации.

```

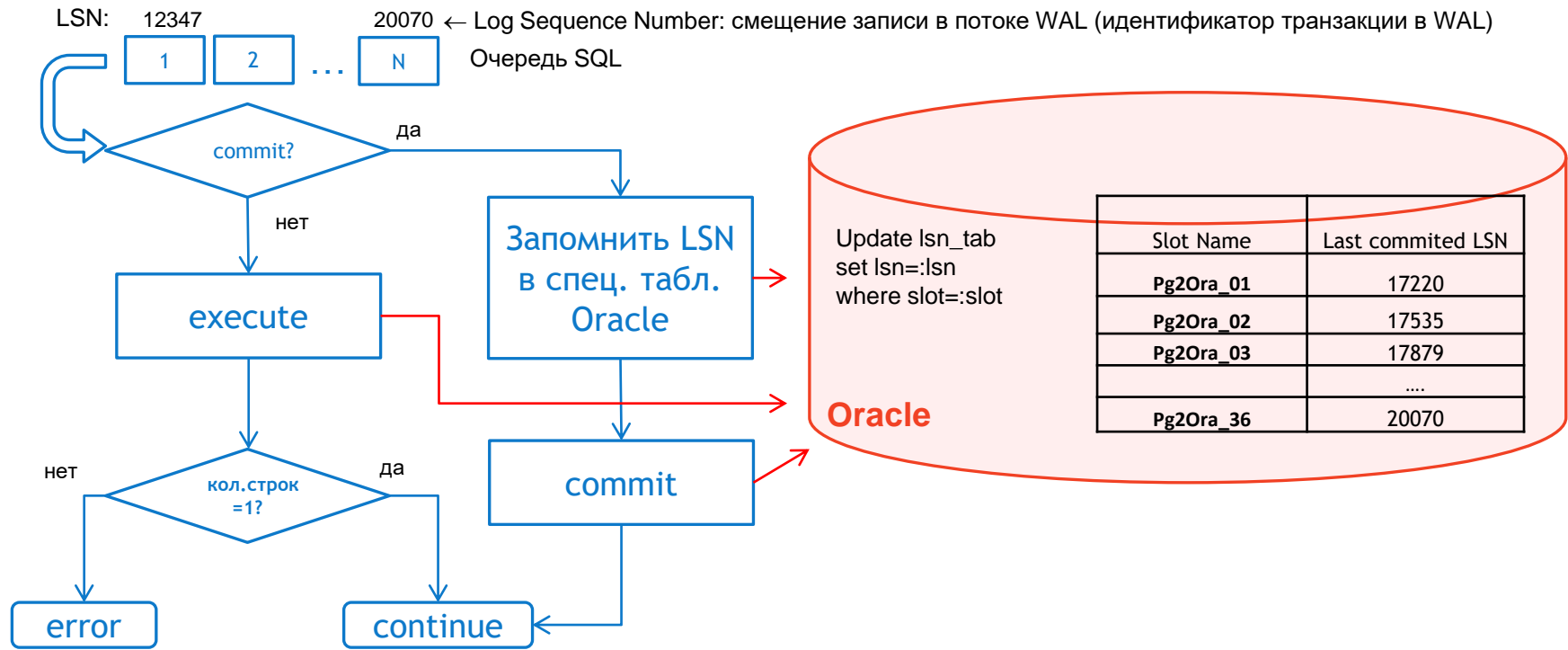
Begin
  Byte1('B')
    Указывает, что это начальное сообщение.
  Int64 (XLogRecPtr)
    Окончательный LSN транзакции.
  Int64 (TimestampZ)
    Время фиксации транзакции. Значение задаётся в микросекундах, прошедших с начала эпохи PostgreSQL (2000-01-01).
  Int64 (TransactionId)
    Идентификатор транзакции.
Сообщение
  Byte1('M')
    Указывает, что это сообщение логического декодирования.
  Int64 (TransactionId)
    Идентификатор транзакции (только для транзакций, передаваемых потоками). Это поле присутствует со 2-й версии протокола.
  Int8
    Флаги, либо 0, если флагов нет, либо 1, если сообщение логического декодирования передано как транзакционное.
  Int64 (XLogRecPtr)
    LSN сообщения логического декодирования.
  
```



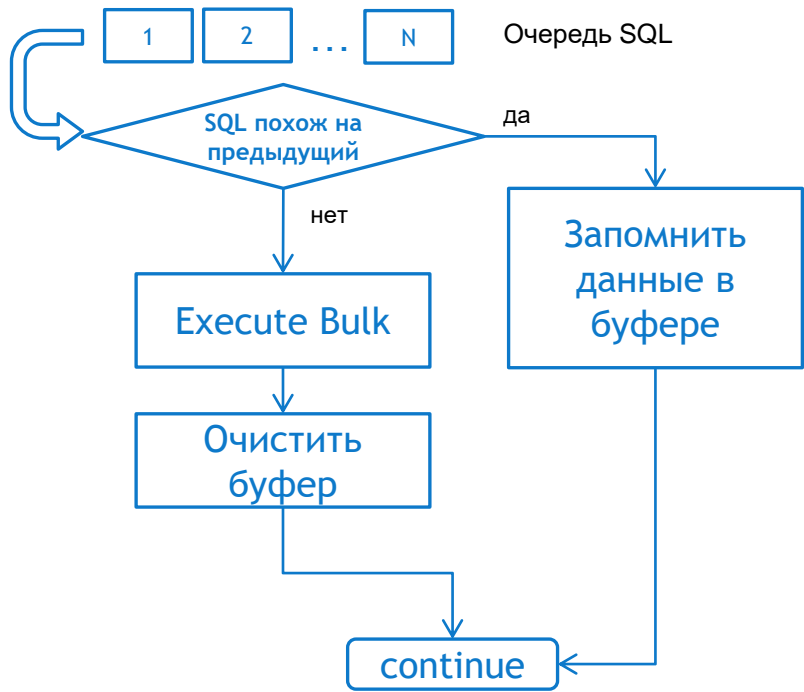
Внутреннее устройство



16 Ora->Pg->Ora Pg2OraSync – репликация без потерь



- Возможность продолжить репликацию после сбоя
- Контроль количества обновлённых и удалённых записей

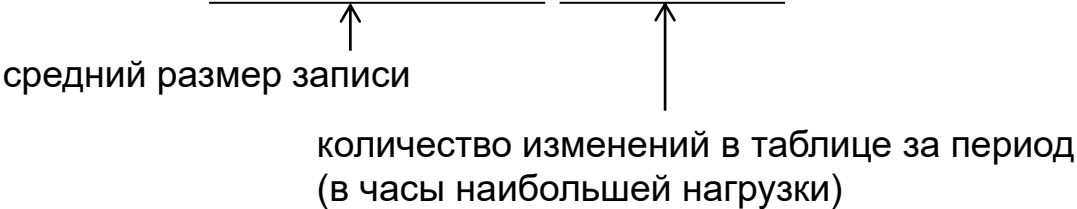


```
update table1  
set notes = 'note '||id  
where id between 1 and 100000;
```

Иногда целесообразно для таблиц подверженных массовым операциям выделить отдельные слоты

Slot	Publication	Tables
Pg2Ora_01	Pg2Ora_01	table_1, table_4, table_8
Pg2Ora_02	Pg2Ora_02	table_2, table_5
Pg2Ora_03	Pg2Ora_03	table_10
Pg2Ora_04	Pg2Ora_04	table_3, table_6
Pg2Ora_05	Pg2Ora_05	table_7

Критерий распределения: $avg_row_length * dml_count$



Подсчёт количеств изменений в таблицах исходной системы на Oracle по расписанию.

Расписание: за 1 час до старта dbms_stats

```

SELECT start_time - to_dsinterval('000 01:00:00') datetime_tz
FROM dba_autotask_schedule      ts,
     dba_scheduler_wingroup_members wgm,
     dba_autotask_client        atc
where ts.WINDOW_NAME = wgm.WINDOW_NAME
      and wgm.WINDOW_GROUP_NAME = atc.WINDOW_GROUP
      and atc.CLIENT_NAME = 'auto optimizer stats collection'
      and atc.STATUS = 'ENABLED'
order by start_time FETCH FIRST 7 ROWS ONLY

```

DATETIME_TZ
01.10.24 21:00:00 +03:00
02.10.24 21:00:00 +03:00
03.10.24 21:00:00 +03:00
04.10.24 21:00:00 +03:00
05.10.24 05:00:00 +03:00
06.10.24 05:00:00 +03:00
07.10.24 21:00:00 +03:00

Извлечение: за 1 час до старта dbms_stats

```

select table_owner, table_name,
       inserts+updates+deletes
       as dml_count
from dba_tab_modifications
where table_owner in ('...', '...')

```

TABLE_OWNER	TABLE_NAME	DML_COUNT
FWRK1	IN_T_ELEM	2411
FWRK1	IN_T_ELEM_ATTR	4223
FWRK1	IN_T_SUBSYSTEM_ELEM	3003
FWRK2	IT_T_CONTR_WEBUSER	1220
FWRK2	IT_T_WEBUSERPROP	1804
FWRK3	IT_T_WEBUSERS	1205
FWRK3	IV_T_INVOICEF_UNAFFECTED	18033
FWRK3	MR_T_BEHAVIOUR	1817
FWRK3	RD_T_ACTION_QUEUE	1201
FWRK3	SY_T_DATA_LOG	41409
FWRK3	SY_T_USER_TAB_COLUMN	608
FWRK3	TR_T_ARCHIVE_QUEUE	70207
FWRK3	TR_T_CDR_DATA_QUEUE	13801
FWRK3	TR_T_PHONE_NUMBERS	10202

Разбиение наиболее нагруженных таблиц на части

```
select pg_drop_replication_slot(rs.slot_name) from pg_replication_slots rs where rs.slot_name like 'pg2ora_%';

select pg_create_logical_replication_slot('pg2ora_01','pgoutput');
drop publication if exists pg2ora_01;
create publication pg2ora_01 for table
  fwork."gl_t_egrul$"      where mod(id,5)=0,
  fwork.ev_t_action       where mod(id,5)=0,
  fwork.it_t_radius_v2_log where mod(id,5)=0,
  fwork.bl_t_fineday      where mod(id,5)=0,
  fwork.ad_t_audit        where mod(id,5)=0
with ( publish_via_partition_root=true );

select pg_create_logical_replication_slot('pg2ora_02','pgoutput');
drop publication if exists pg2ora_02;
create publication pg2ora_02 for table
  fwork."gl_t_egrul$"      where mod(id,5)=1,
  fwork.ev_t_action       where mod(id,5)=1,
  fwork.it_t_radius_v2_log where mod(id,5)=1,
  fwork.bl_t_fineday      where mod(id,5)=1,
  fwork.ad_t_audit        where mod(id,5)=1
with ( publish_via_partition_root=true );

select pg_create_logical_replication_slot('pg2ora_03','pgoutput');
drop publication if exists pg2ora_03;
create publication pg2ora_03 for table
  fwork."gl_t_egrul$"      where mod(id,5)=2,
  fwork.ev_t_action       where mod(id,5)=2,
  fwork.it_t_radius_v2_log where mod(id,5)=2,
  fwork.bl_t_fineday      where mod(id,5)=2,
  fwork.ad_t_audit        where mod(id,5)=2
with ( publish_via_partition_root=true );
```

- Настройка преобразования типов данных Postgres->Oracle в т.ч. собственными функциями
- Настройка отображения схема Postgres->схема Oracle
- Настройка отображения таблица Postgres->таблица Oracle
- Исключение отдельных столбцов из репликации (например rowid)
- Постобработка изменений в Oracle

http://localhost:8080/ восторенный web-сервер для отображения хода репликации

Start time	01.10.2024 12:02:02	JVM Memory			PG WALs	Refresh Interval				Format HR	Stop Incoming	Stop Programm	Write Nowait	Debug Off	GC	
Current time	01.10.2024 12:04:40	used	1.868G	free	857.898M	1.141G	0	1	3							5
Running time	0.00:02:37	total	2.726G	max	8.590G											

#	Slot	Status	W	Incomming Msg			Msg Queue		Sql Queue		Outgoing Msg			PG LO bytes	Speed
				Count	Commits	Bytes	Count	Bytes	Count	Bytes	Count	Commits	Bytes		
1	pg2ora_01	Active		0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
2	pg2ora_02	Active		0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
3	pg2ora_03	Active				501.455K	0.000	0.000	0.000	0.000	5.419K	9.000	501.455K	0.000	0.000
4	pg2ora_04	Error				5.242M	0.000	0.000	600.000	336.645K	8.431K	14.000	4.905M	0.000	0.000
5	pg2ora_05	Active				2.789M	0.000	0.000	1.039K	94.765K	30.266K	50.000	2.694M	0.000	14.423K
6	pg2ora_06	Active				5.163M	0.000	0.000	0.000	0.000	15.653K	26.000	5.163M	0.000	0.000
7	pg2ora_07	Active				0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
8	pg2ora_08	Active				259.823K	0.000	0.000	0.000	0.000	4.849K	215.000	259.823K	0.000	0.000
9	pg2ora_09	Active				5.010M	0.000	0.000	2.946K	246.934K	46.292K	354.000	4.763M	0.000	13.548K
10	pg2ora_10	Active		110.321K	385.000	12.441M	0.000	0.000	64.098K	7.229M	46.223K	161.000	5.212M	0.000	52.589K
11	pg2ora_11	Active		44.649K	255.000	3.948M	0.000	0.000	0.000	0.000	44.649K	255.000	3.948M	0.000	0.000
12	pg2ora_12	Active		162.408K	547.000	12.740M	0.000	0.000	112.978K	8.772M	49.430K	164.000	3.968M	0.000	50.063K
13	pg2ora_13	Active		83.665K	392.000	11.317M	0.000	0.000	39.535K	5.467M	44.130K	201.000	5.850M	0.000	24.084K
14	pg2ora_14	Active		138.647K	394.000	7.741M	0.000	0.000	85.257K	4.709M	53.390K	149.000	3.032M	0.000	26.308K
15	pg2ora_15	Active		52.772K	263.000	3.534M	0.000	0.000	3.864K	272.439K	48.908K	247.000	3.261M	0.000	47.064K
16	pg2ora_16	Active		44.210K	280.000	4.901M	0.000	0.000	0.000	0.000	44.210K	280.000	4.901M	0.000	0.000
17	pg2ora_17	Active		119.720K	399.000	14.702M	0.000	0.000	74.736K	9.108M	44.984K	150.000	5.594M	0.000	147.232K
18	pg2ora_18	Active		76.044K	329.000	8.466M	0.000	0.000	30.099K	3.425M	45.945K	191.000	5.041M	0.000	25.223K
19	pg2ora_19	Active		57.621K	232.000	9.422M	0.000	0.000	11.163K	1.840M	46.458K	184.000	7.581M	0.000	81.226K
20	pg2ora_20	Active		54.290K	308.000	4.965M	0.000	0.000	8.901K	841.643K	45.389K	252.000	4.123M	0.000	23.950K
				1.060M	4.471K	113.142M	0.000	0.000	435.216K	42.344M	624.626K	2.902K	70.798M	0.000	505.710K

[4] slot pg2ora_04 Rollback:
ERROR: update count=0
Table:fwrk1.ad_t_log
Id=12007
Evt= register
Note = reg01012

+ Уведомление об ошибках по e-mail @ и telegram

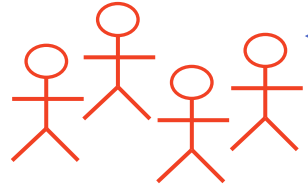


Автоматизация подготовки к репликации и возврата на Oracle

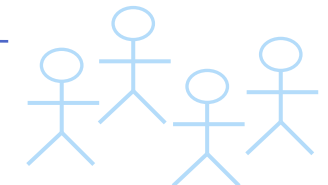
- Управление ограничениями целостности в Oracle
 - Отключение внешних ключей и триггеров перед репликацией и включение их при возврате
- Управление последовательностями в Oracle
 - Установка текущих значений из Postgres в Oracle
- Выявление объектов рискованной репликации
 - Таблицы без ключей
 - Разница в составе столбцов
 - Столбцы “rowid”
 - Уникальные ключи с nullable – столбцами
 - alter table replica identity full/using index ...
- Функциональное и нагрузочное тестирование репликации
 - Проверка реплицируемости данных всех таблиц
 - Циклическая генерация dml для групп таблиц в нескольких потоках
- Создание слотов и публикаций с равномерной нагрузкой

24 Ora->Pg->Ora

Pg2OraSync Результаты



Возврат на Oracle ~ 1ч



Что нам мешало?

- Не работает **with** (publish_via_partition_root=**true**)
 - В сообщениях всё равно указана секция
 - Identity full для root не действует на существующие секции
- Секционированные таблицы с rowid
 - ...primary key(row_id, sec_key);
- Timestamp
 - Oracle точность долей секунды – 9 зн.
 - Postgres – 6 зн.
- Timestamp with time zone


```
select to_char(current_timestamp,'yyyy-mm-dd hh24:mi:ss.ff6 tzh:tzm') union all
select to_char(current_timestamp - '100 year'::interval,'yyyy-mm-dd hh24:mi:ss.ff6 tzh:tzm') union all
select to_char(current_timestamp - '105 year'::interval,'yyyy-mm-dd hh24:mi:ss.ff6 tzh:tzm') union all
select to_char(current_timestamp - '106 year'::interval,'yyyy-mm-dd hh24:mi:ss.ff6 tzh:tzm') union all
select to_char(current_timestamp - '1000 year'::interval,'yyyy-mm-dd hh24:mi:ss.ff6 tzh:tzm')
```

2025-02-05 17:25:44.439360	+03:00
1925-02-05 17:25:44.439360	+02:00
1920-02-05 17:25:44.439360	+03:00
1919-02-05 17:25:44.439360	+03:31
1025-02-05 17:25:44.439360	+02:30
- Служба информационной безопасности
- Поздно получили стенды для тестирования

Направления развития

- Поточковая логическая репликация
- Один слот в PG ->Разделение на сеансы =>Oracle
- Конфигурирование “на лету”

Когда это полезно?

- Снятие рисков и страхов “что делать если в Postgres всё будет плохо ?”
- Интеграция с системами оставшимися на Oracle (CDC Postgres->Oracle)
- “Наш старый добрый Oracle питал данными BI-системы!”
- Ещё одна проверка качества миграции с возможностью повторить

Вопросы ???

E-mail:

forstelecom@fors.ru

lui@fors.ru

Сайт:

<http://lui.fors.ru>

<http://forstelecom.ru>

Телефон:

+7 495 798 9838

доб. 240, 267, 239, 212

Мобильный:

+7 903 505-9567

+7 903 505-9564



Телеком

Форс