



# BRAVIANT

Calling Remote Functions  
Using FDW

Hettie Dombrovskaya  
Braviant Holdings

**MARCH 2019**

## USING FDW WE CAN:

Map tables, views, other foreign tables, even queries. But can we map a function?...

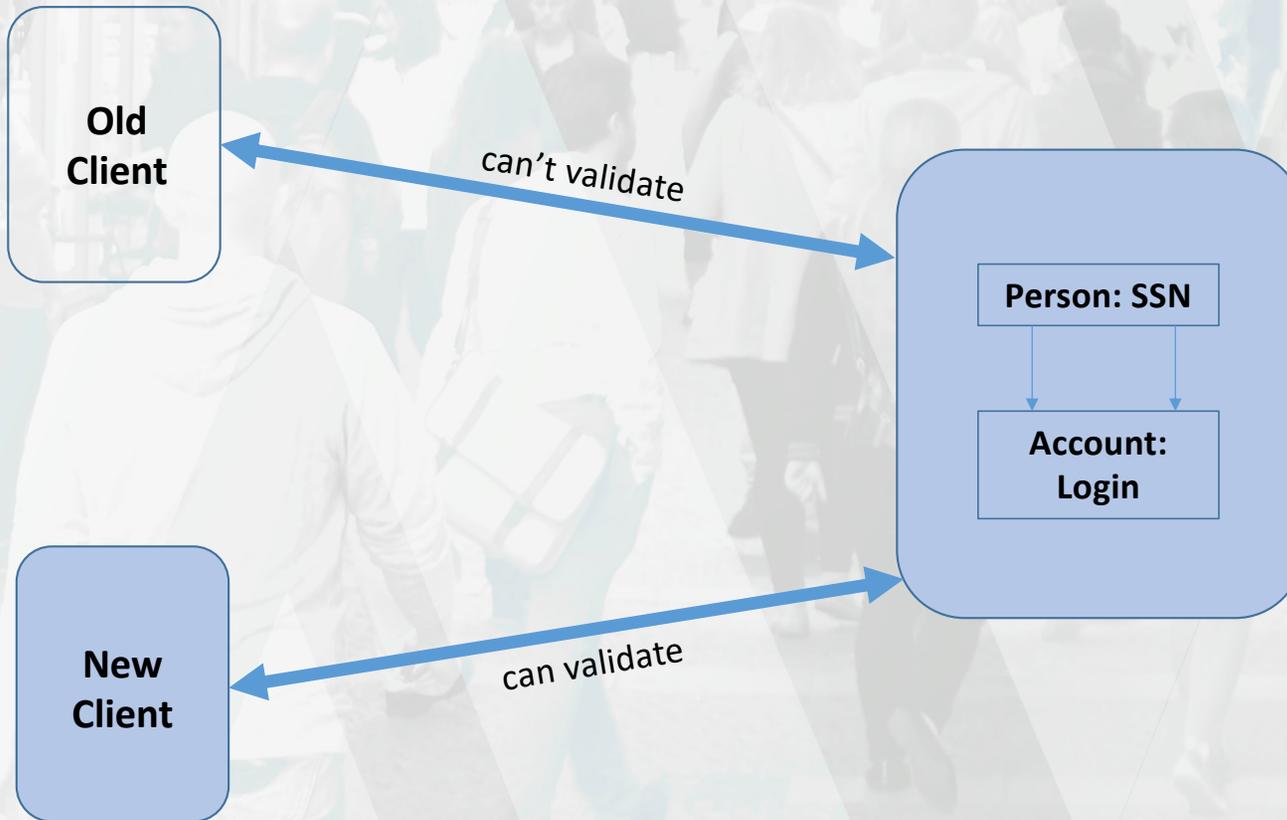
---

Technically speaking, we can't, but... when there is a will, there is away, and I **LOVE** functions!



**WHY IN THE WORLD  
YOU MAY NEED IT???**

# CASE STUDY 1: IDENTITY VALIDATION





**OUR SOLUTION:**

**We can create a trigger on the application table, which will call the remote function to validate identity... if we can call it!**

# HERE IS HOW

On the IDENTITY side:

```
CREATE OR REPLACE VIEW
identity.user_account_select_fdw_tests_v AS
SELECT u.username,
u.brand,
identity.user_account_get_j(u.username, u.brand) AS
user_account_json
FROM identity.user_account u
WHERE now()<@effective and now()<@ asserted;
```

# HERE IS HOW

On the RECEIVING side:

```
CREATE FOREIGN TABLE
bsf_prod_identity_fdw.user_account_select_fdw_tests_v(
  username text,
  brand text,
  user_account_json text)
SERVER "bsf_prod_server"
OPTIONS (
  schema_name 'identity',
  table_name 'user_account_select_fdw_tests_v');
```

# AND NOW GUESS WHAT

We can run it remotely!

```
SELECT * FROM  
bsf_prod_identity_fdw.user_account_select_fdw_tests_v  
  WHERE username='cordelialrlanc@yahoo.com' and  
         brand='balance';
```

```
SELECT * FROM  
bsf_prod_identity.user_account_ssn_select_fdw_tests_v  
  WHERE ssn='699034651';
```

# CASE STUDY 2: UTILIZING REMOTE INDEXES

## custom\_fields\_entity

```
id integer
entity_type text
custom_field_id integer
entity_id integer
custom_field_value text
```

## loan\_entity

```
id integer
user_account_id integer
```

loan\_id

## loan\_settings\_entity

```
id integer
loan_id integer
loan_status_id integer
loan_amount numeric
```

entity\_id

```
entity_type='loan_settings'
custom_field_id=53
```

## WHAT DOES NOT WORK HERE?

```
SELECT custom_field_value AS servicer
FROM loan_entity l JOIN
loan_settings s ON l.id=s.loan_id AND id=p_loan_id
LEFT OUTER JOIN
custom_field__entity cfls ON cfls.custom_field_id=53
AND entity_type='loan_settings'
AND cfls.entity_id=s.id
```

- Remote site – MySQL
- Indexes are not being used – need to substitute specific value

## OUR SOLUTION

```
create or replace function loan_servicer (p_loan_id int,  
p_loan_settings_id int)  
returns setof text as  
$BODY$\nbegin\nreturn query ( select servicer from (select ls.loan_id,  
(select cfls_53.custom_field_value  
from custom_field_entity cfls  
where cfls.custom_field_id=53  
and entity_type='loan_settings' and  
entity_id=p_loan_settings_id ) as servicer  
from loan_settings_entity ls where ls.loan_id=p_loan_id. )a);\nend;\n$BODY$\nlanguage plpgsql;
```

## OUR SOLUTION

Incorporating function into the view:

```
CREATE VIEW loan_settings_custom_v
AS
SELECT loan_id,
       Id,
       (SELECT loan_servicer FROM loan_servicer(loan_id,id )) AS
       servicer FROM
loan_settings_entity;
```

We map this view and then execute the function remotely

```
SELECT servicer FROM lp_fdw. loan_settings_custom_v (123456);
```



**QUESTIONS, DISCUSSIONS,  
AND WHAT ELSE!**