

Business logic with PostgreSQL and Python

Using functions and triggers and PostgreSQL magic with plpy.



Knock knock, who's there?



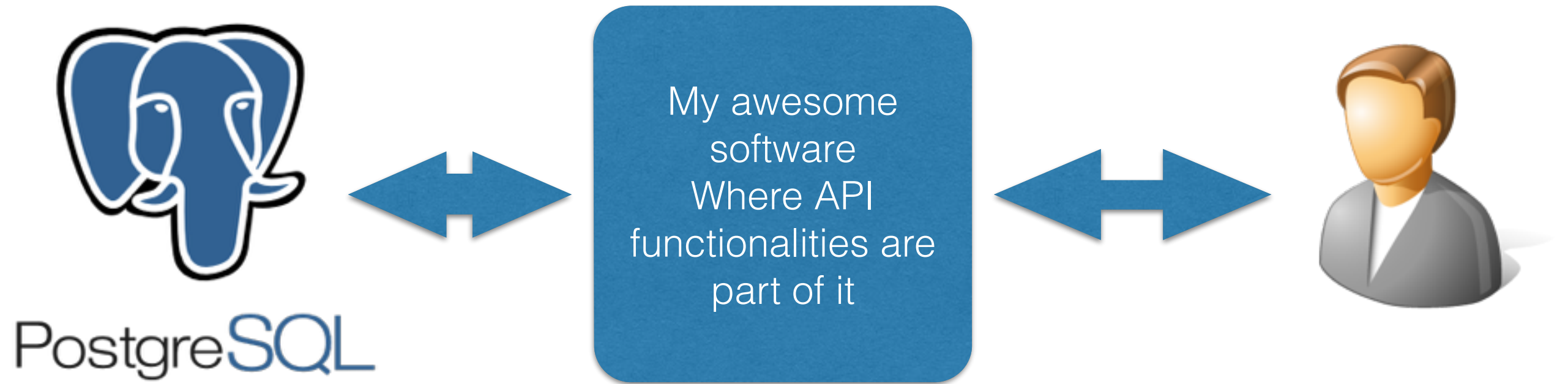
Hubert Piotrowski
Senior systems architect



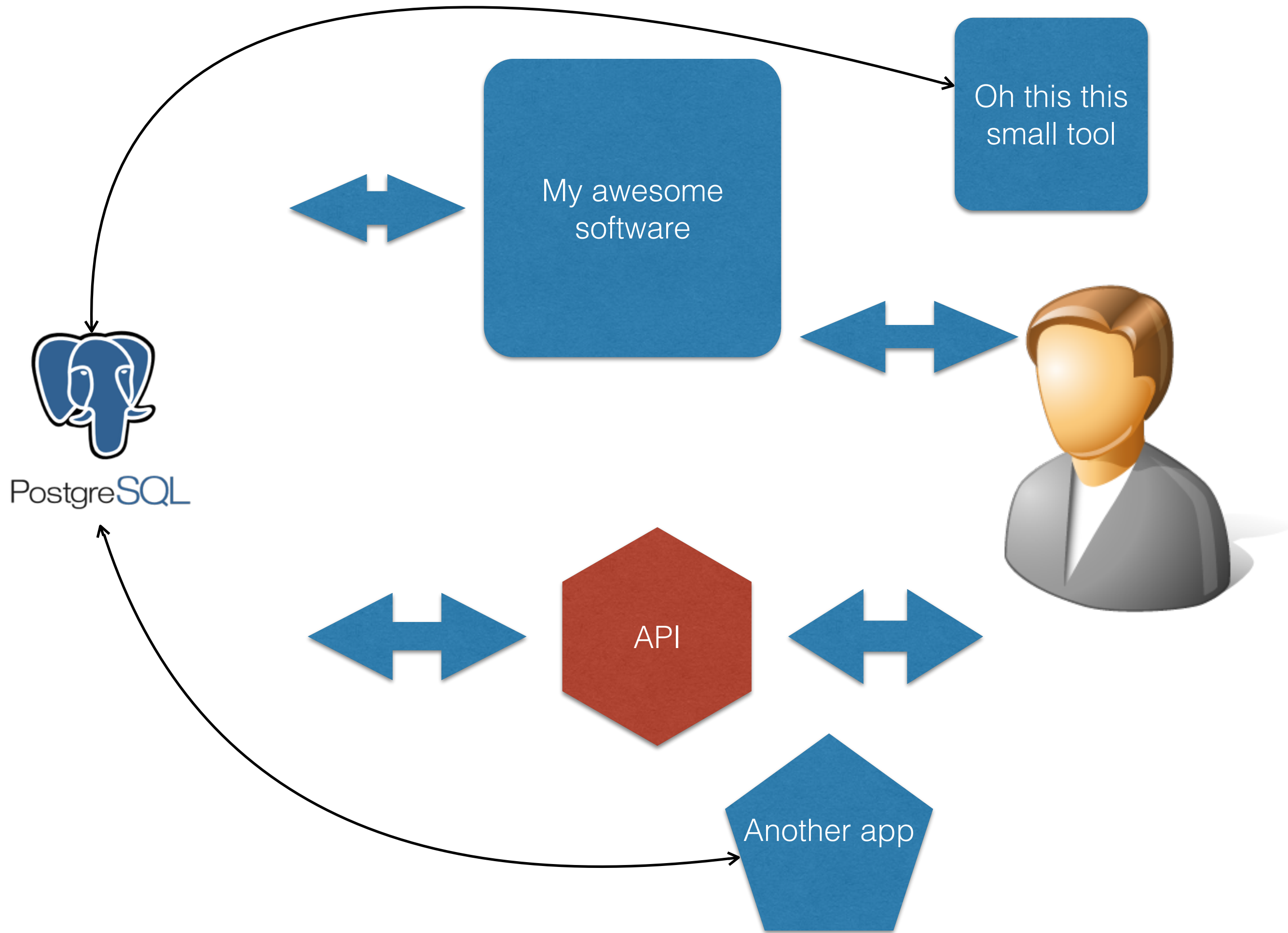
How to build API

How to build API

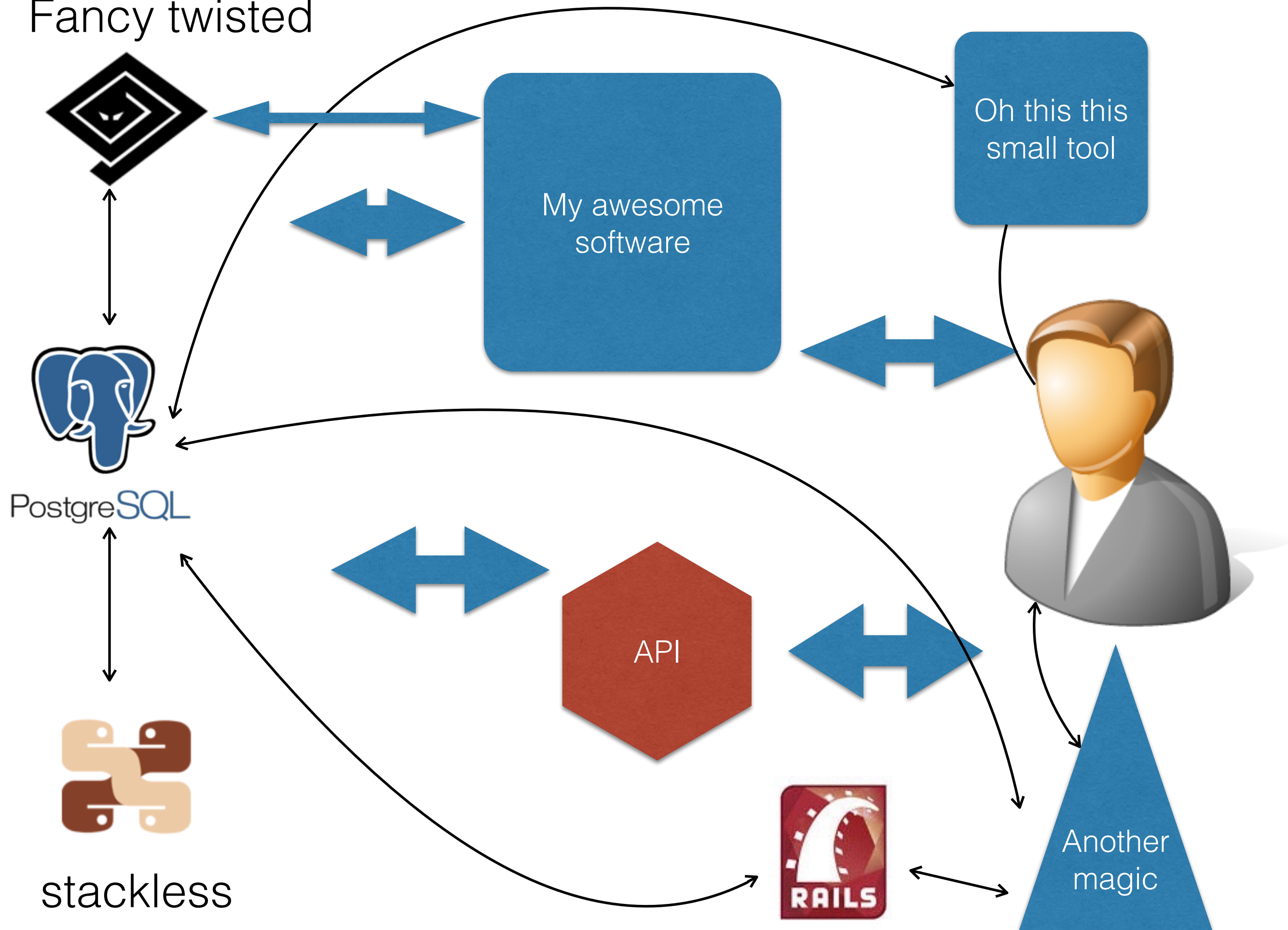
Perfect API



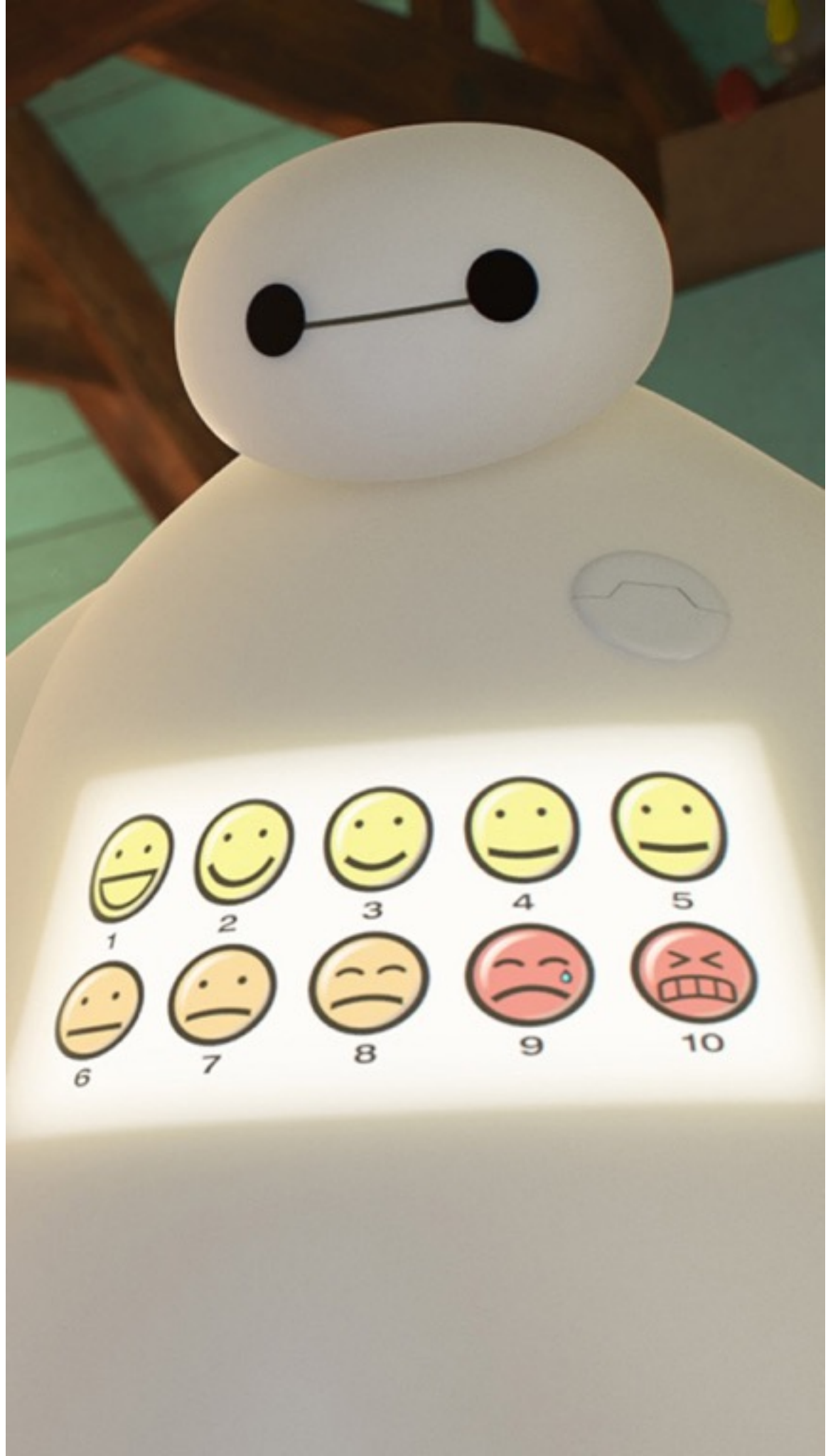
Welcome to the
Reality

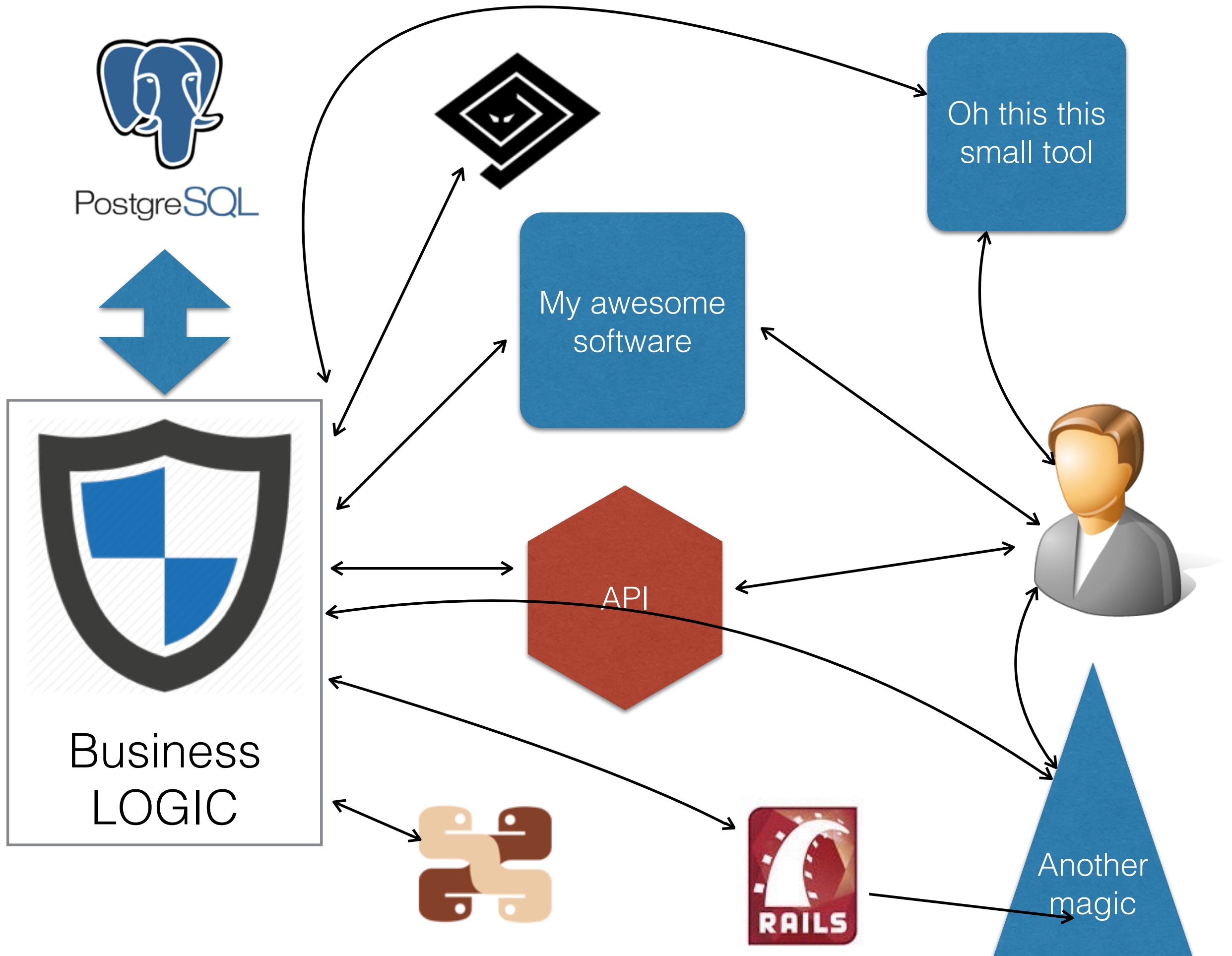


Fancy twisted



HELP?





What is so great about PostgreSQL

- Dynamic data typing
- Procedural functions (many languages)
- Horizontal scaling
- Plugins
- Triggers
- Granularity for access privileges

Data types

- Standard: Strings, float, inter, etc.
- JSON
- Dynamic (composite)
- Arrays

Arrays

```
pay_by_quarter integer ARRAY[4]
```

Or, if no array size is to be specified:

```
pay_by_quarter integer ARRAY
```

Arrays

```
INSERT INTO sal_emp  
VALUES ('Bill',  
'{10000, 10000, 10000, 10000}',  
'{"meeting", "lunch"}, {"training", "presentation"}');
```

And get it

```
SELECT name FROM sal_emp WHERE pay_by_quarter[1] <> pay_by_quarter[2];  
  
name  
-----  
Carol  
(1 row)
```

Still arrays

```
SELECT schedule[1:2][2] FROM sal_emp WHERE name = 'Bill';
```

schedule

```
-----  
{{meeting,lunch},{training,presentation}}  
(1 row)
```

```
SELECT * FROM sal_emp WHERE 10000 = ANY (pay_by_quarter);
```


Composite

```
CREATE TYPE inventory_item AS (  
  name      text,  
  supplier_id integer,  
  price     numeric  
);
```

```
CREATE TABLE on_hand (  
  item    inventory_item,  
  count   integer  
);
```

```
INSERT INTO on_hand VALUES (ROW('fuzzy dice', 42, 1.99), 1000);
```

Procedures

plPythonu(?) one second

```
CREATE FUNCTION merge_fields(t_row table1) RETURNS text AS $$  
DECLARE  
    t2_row table2%ROWTYPE;  
BEGIN  
    SELECT * INTO t2_row FROM table2 WHERE ... ;  
    RETURN t_row.f1 || t2_row.f3 || t_row.f5 || t2_row.f7;  
END;  
$$ LANGUAGE plpgsql;  
  
SELECT merge_fields(t.*) FROM table1 t WHERE ... ;
```

Better formatting

```
CREATE FUNCTION merge_fields(t_row table1)
RETURNS text AS
$$
DECLARE
    t2_row table2%ROWTYPE;
BEGIN
    SELECT * INTO t2_row FROM table2 WHERE ... ;
    RETURN t_row.f1 || t2_row.f3 || t_row.f5 || t2_row.f7;
END;
$$
LANGUAGE plpgsql;

SELECT merge_fields(t.*) FROM table1 t WHERE ... ;
```

Let there be light...and there was light



Compile PostgreSQL 9.4 with Python 2.7.10

Python compiled with special flags

- PYTHONHOME
- PYTHONPATH
- PYTHON2K
- PYTHONOPTIMIZE
- PYTHONDEBUG
- PYTHONVERBOSE
- PYTHONCASEOK
- PYTHONDONTWRITEBYTECODE
- PYTHONIOENCODING
- PYTHONUSERBASE

Compile python

```
[hubert@ThePit]~/stuff/Python-2.7.10% ./configure --PREFIX=/opt/py --enable-shared
```

Compilation takes some time...

Once it's done... we need easy_install and other python modules

```
postgres@9.4.3 ➤ wget https://bootstrap.pypa.io/ez_setup.py -O - | /opt/py/bin/python
--2015-06-05 23:37:54-- https://bootstrap.pypa.io/ez_setup.py
Resolving bootstrap.pypa.io... 103.245.222.175
Connecting to bootstrap.pypa.io|103.245.222.175|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 11432 (11K) [text/x-python]
Saving to: "STDOUT"

100%[=====>] 11,432

2015-06-05 23:37:54 (1.36 GB/s) - written to stdout [11432/11432]

Extracting in /tmp/tmp0hnlk9
```

issues?

```
py ➤ bin/python2.7  
bin/python2.7: error while loading shared libraries: libpython2.7.so.1.0: cannot open  
shared object file: No such file or directory
```

simple fix

```
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/py/lib
```


Compile PostgreSQL

```
postgresql-9.4.3 ➤ ./configure --prefix=/opt/pgsql \
—with-python PYTHON=/opt/py/bin/python
postgresql-9.4.3 ➤ make
postgresql-9.4.3 ➤ make install
```

Compile essential contrib data

```
contrib ➤ cd ~/stuff/postgresql-9.4.3/contrib
contrib ➤ make
contrib ➤ make install
```

Initialize postgresql data space

```
~ ➤ mkdir /opt/pg_data
~ ➤ /opt/pgsql/bin/initdb /opt/pg_data
~ ➤ /opt/pgsql/bin/postmaster -p 5432 -D /opt/pg_data/
```

```
~ ➤ /opt/pgsql/bin/psql -h localhost -l
```

```
          List of databases
  Name      | Owner   | Encoding | Collate  | Ctype    | Access privileges
-----+-----+-----+-----+-----+-----+-----
 postgres   | hubert  | UTF8     | en_US.UTF-8 | en_US.UTF-8 |
 template0   | hubert  | UTF8     | en_US.UTF-8 | en_US.UTF-8 | =c/hubert      +
              |         |          |             |             | hubert=CTc/hubert
 template1   | hubert  | UTF8     | en_US.UTF-8 | en_US.UTF-8 | =c/hubert      +
              |         |          |             |             | hubert=CTc/hubert
(3 rows)
```

initialize database

```
~ ➤ /opt/pgsql/bin/createdb -h localhost -E utf8 pie
```

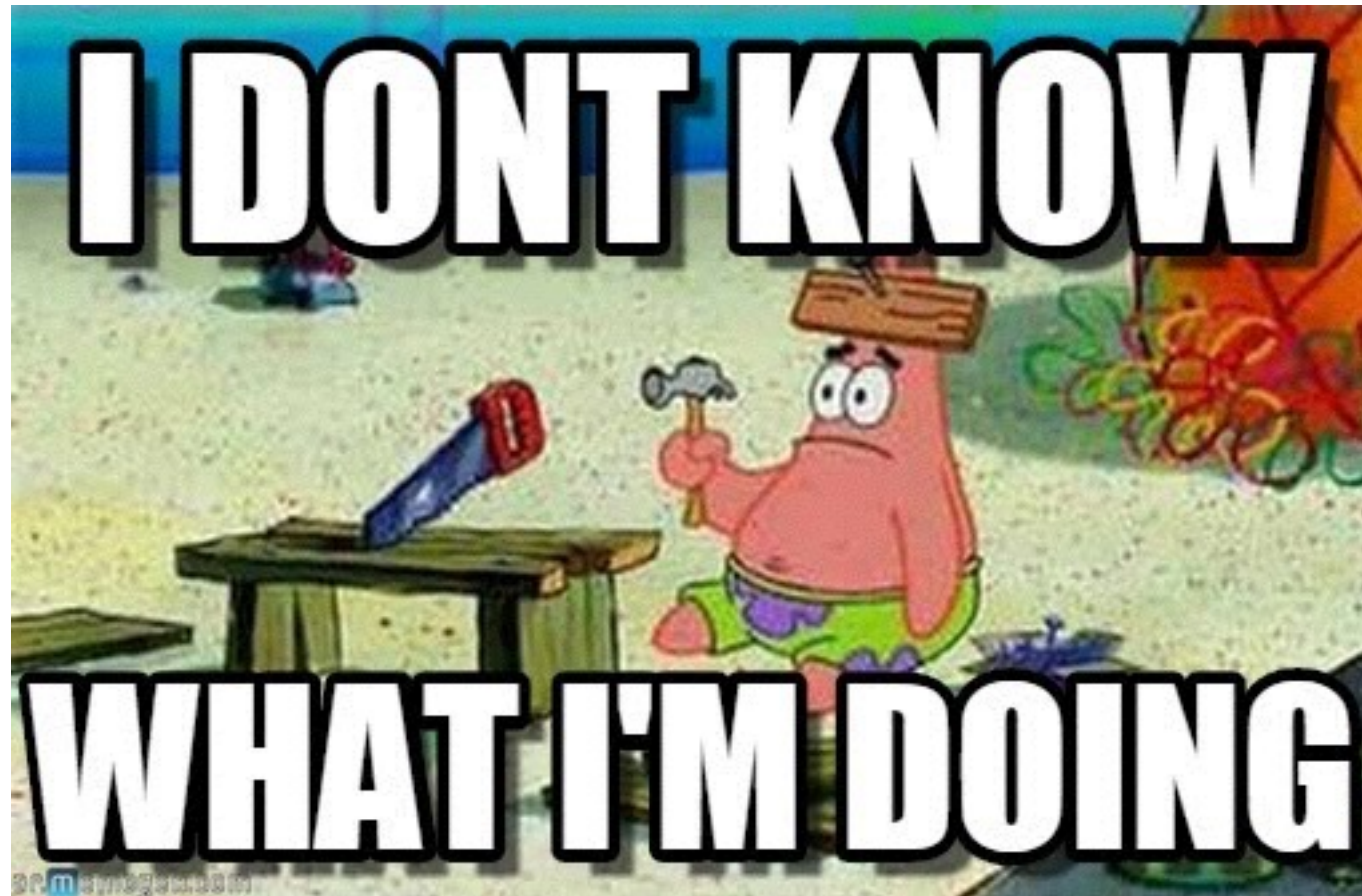
Create language plpython in database

```
~ ➤ /opt/pgsql/bin/postmaster -p 5432 -D /opt/pg_data/  
LOG: database system was shut down at 2015-06-07 17:29:05 SGT  
LOG: database system is ready to accept connections  
LOG: autovacuum launcher started
```

```
~ ➤ /opt/pgsql/bin/createlang -h localhost -d pie plpythonu
```

```
pie=# select * from pg_extension ;  
  extname  | extowner | extnamespace | extrelocatable | extversion | extconfig |  
extcondition  
-----+-----+-----+-----+-----+-----+-----  
plpgsql   |      10 |      11 | f           |      1.0   |           |  
plpythonu |      10 |      11 | f           |      1.0   |           |  
(2 rows)
```

Let uncle Google help you



didn't find the answer?

**Probably the issue that you have means that
what you try to achieve is not possible or...**

doesn't make sense

First **plPy** function



Hello world

```
create or replace function logic.hello_world()  
returns void as  
$$  
""  
  
Code code code code  
""  
  
print "hello world"  
$$  
LANGUAGE plpythonu VOLATILE;
```

Hello world

```
create or replace function logic.hello_world()  
returns void as  
$$  
""  
  
Code code code code  
""  
  
plpy.info("hello world")  
  
$$  
LANGUAGE plpythonu VOLATILE;
```


Triggers



- can be fired **before** or **after** (*instead of*)
- upon action: insert, update or delete
- trigger fires **function** (also by using **when** condition)
- trigger **function** can but don't have to manipulate data
- multiple triggers on the same table

Let's see some action

Hungry for more?



<https://github.com/darkman66/pyconsg2015>



darkman66

tumblr. <http://my-twisted-code.tumblr.com>