Comet web applications with Python, Django & Orbited

@ PyCon Italia Qu4ttro
http://www.pycon.it/

Massimo Scamarcia
http://www.webright.it/

How to push real-time data to web browsers?

Comet web applications with Python, Django & Orbited

AJAX Polling

- Client: sends AJAX request.
- Server:
  - Data available: sends response.
  - No data: sends empty response.
- Client: sends AJAX request again.

Comet web applications with Python, Django & Orbited

Latency, bandwidth, memory usage and scaling issues!

What about Comet?

Not a technology in itself

Long Polling, IFRAME stream, HTMLFile, XHR Streaming, HTML5 SSE

It's an hack!

Still waiting for an unanimous way to do it.

Comet web applications with Python, Django & Orbited

Long Polling

- Client: sends AJAX request.
- Server: holds the request.
    - Data available: sends response.
    - Timeout reached: empty response.
- Client: sends AJAX request again.

Comet web applications with Python, Django & Orbited

Http Push (HTTP Streaming*)

- Client: sends HTTP request.
- Server: doesn't terminate the connection.
- Server: sends data when available.
- Connection closed:
    - Data is queued.
    - Client reconnects.

[*] http://ajaxpatterns.org/HTTP_Streaming

Comet web applications with Python, Django & Orbited

- HTTP Push: better for heavily-loaded apps.
  - Not cross-browser.
- Long Polling: cross-browser and easy.
  - Bandwidth usage.
  - Too many connections.
- Async Python servers* are better for both:
  - Twisted, Tornado, Dieselweb, Eventlet, Concurrence, Circuits, Gevent, Cogen.

[*] http://nichol.as/asynchronous-servers-in-python

What about Orbited?

Comet web applications with Python, Django & Orbited

He's here to solve problems!

Comet web applications with Python, Django & Orbited

- Based on Twisted.
- STOMP* (ActiveMQ, RabbitMQ), IRC and XMPP protocol support.
- Ready-to-use and cross-browser JavaScript client code.
- You don't have to reinvent the wheel!

[*] http://stomp.codehaus.org/

Using Django and Orbited

Example app developed for PyConIt4:

http://pyconquiz.webright.it/

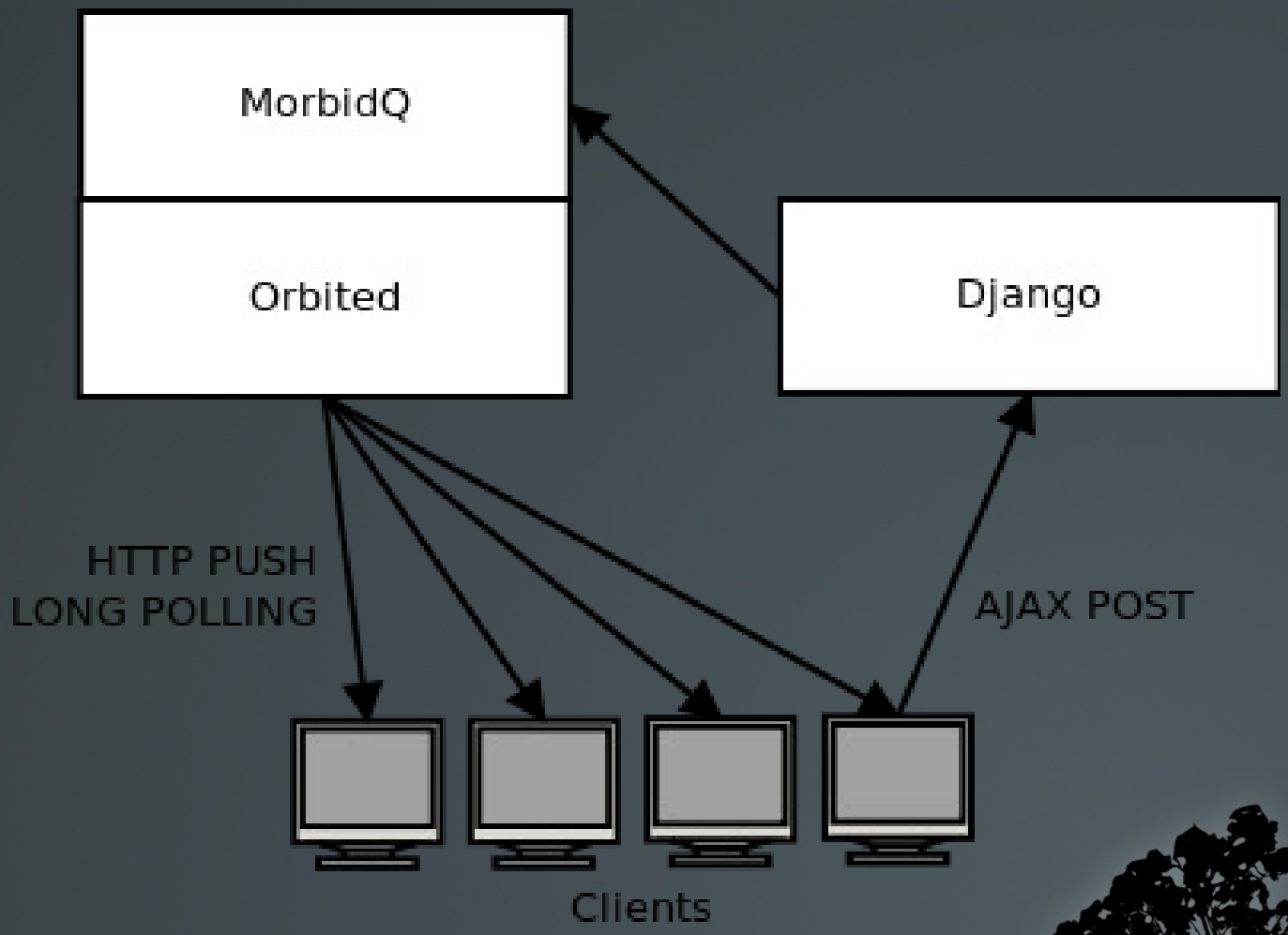
Comet web applications with Python, Django & Orbited

Django is used for:

- Application logic.
- Template rendering.
- User auth and registration.
- Handling AJAX requests.
- Sending messages to STOMP server.

...Orbited and jQuery do the rest!

Comet web applications with Python, Django & Orbited



Comet web applications with Python, Django & Orbited

orbited.cfg

```
[global]
#reactor=select
#reactor=kqueue
reactor = epoll
proxy.enabled = 1
session.ping_interval = 300

[listen]
http://localhost:8080
stomp://localhost:61613

[access]
* -> localhost:61613
```

settings.py

```
# Available using context
# processor or templatetag

ORBITED_HOST = 'localhost'
ORBITED_PORT = 8080

STOMP_HOST = 'localhost'
STOMP_PORT = 61613
```

Comet web applications with Python, Django & Orbited

Django base template (<head /> tag)

```
<script>document.domain = document.domain;</script>
<script src="{{ ORBITED_MEDIA_URL }}Orbited.js"></script>
<script type="text/javascript">
Orbited.settings.port = {{ ORBITED_PORT }};
Orbited.settings.hostname = "{{ ORBITED_HOST }}";
Orbited.settings.streaming = true;

TCPSocket = Orbited.TCPSocket;
</script>

<script src="{{ ORBITED_MEDIA_URL }}JSON.js"></script>
<script
src="{{ ORBITED_MEDIA_URL }}protocols/stomp/stomp.js"></script>
```

Comet web applications with Python, Django & Orbited

```
<script>
$(document).ready(function() {
    stomp = new STOMPClient();
    stomp.onopen = function() { debug('Connected.');};
    stomp.onclose = function(c) { debug(Lost Connection, Code: '+c);};
    stomp.onerror = function(error){ debug("Error: " + error);};
    stomp.onerrorframe = function(frame){ debug("Error: " + frame.body);};
    stomp.onconnectedframe = function() {
        {% if game %}
        stomp.subscribe('/games/{{ game.id }}/status');
        stomp.subscribe('/games/{{ game.id }}/players');
       {% else %}// subscribe to other channels...{% endif %}
    };
    stomp.onmessageframe = function(frame){
        // frame.headers.destination for channel, frame.body for JSON data
        destpatterns.dispatch(frame);
    };
    stomp.connect('{{ STOMP_HOST }}', {{ STOMP_PORT }});
});
</script>
```

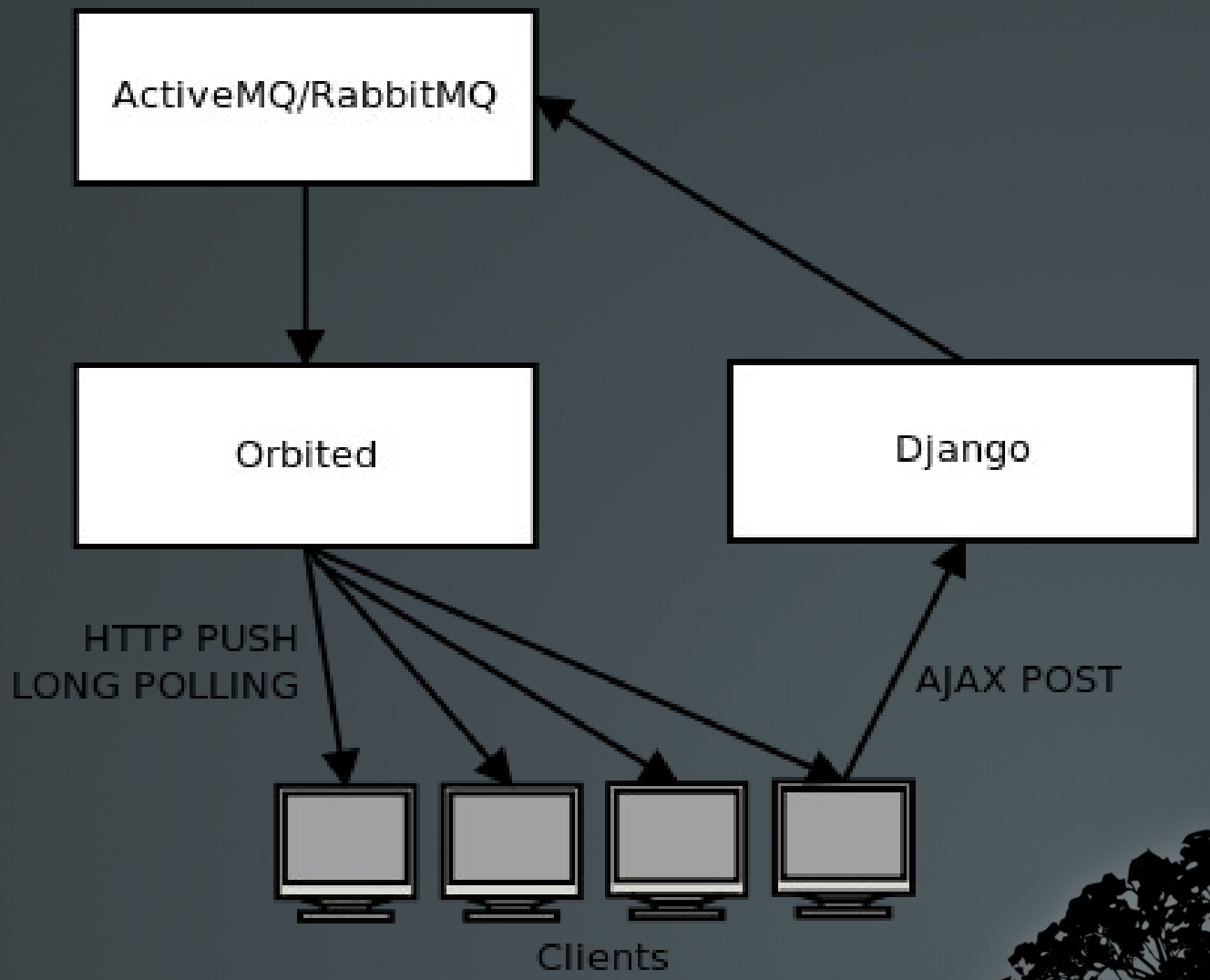
Comet web applications with Python, Django & Orbited

Django view

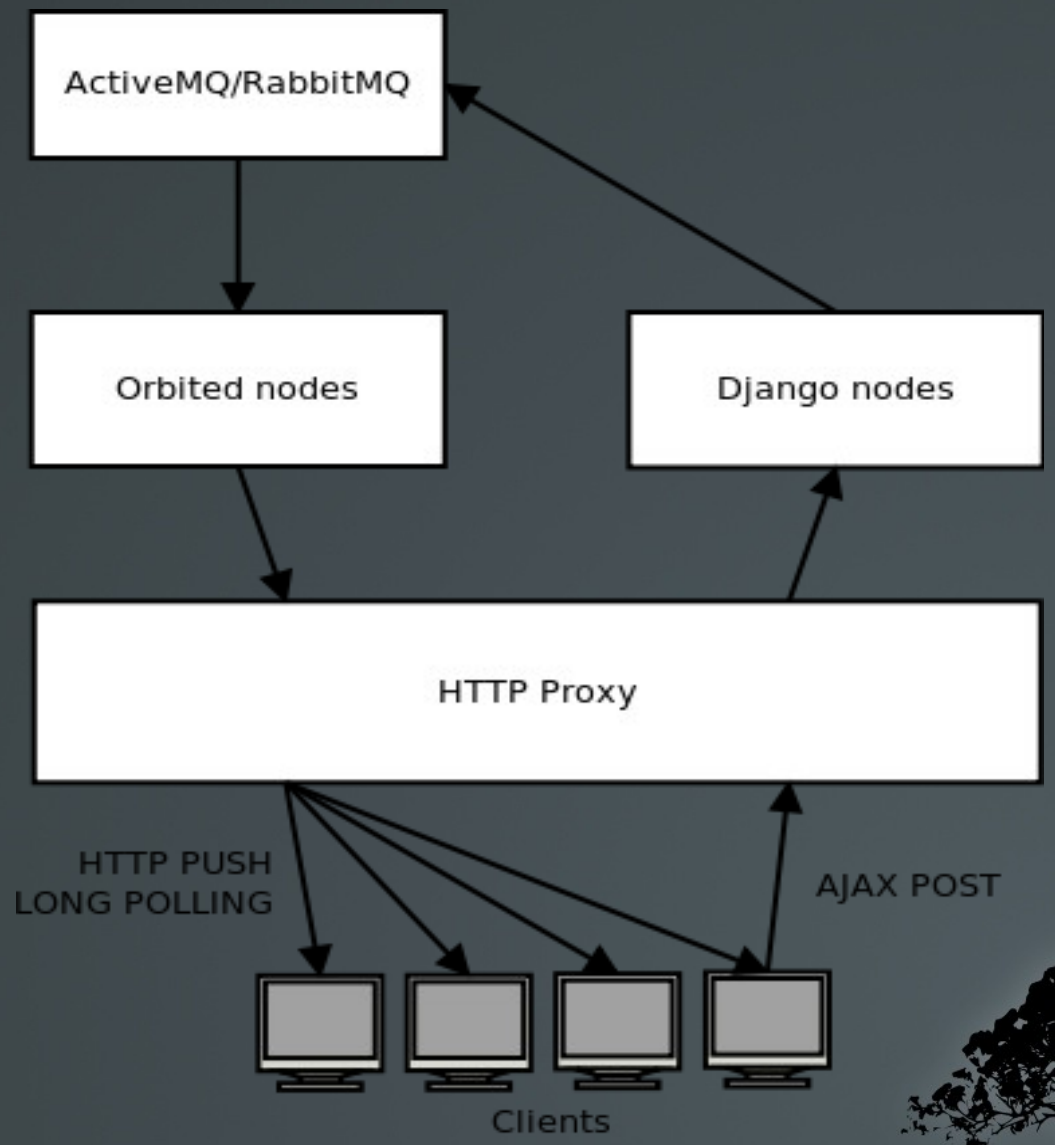
```
@require_POST
@login_required
def game_start(request, game_id):
    game = get_object_or_404(Game, id=game_id)
    if request.user != game.author:
        return HttpResponseForbidden()
    try:
        # start() method set game start_time and send JSON
        # to ”/games/{{ game.id }}/status” using stomp.py
        game.start()
    except GameError, e:
        return JSONResponse({'error': str(e)})
    return JSONResponse({})
```

Does it scale?

Comet web applications with Python, Django & Orbited



Comet web applications with Python, Django & Orbited



Thank you

and have fun with the quiz game!