Application architecture

Lars Holm Nielsen
CERN

Invenio User Group Workshop 2017, Garching bei München

INVENIO

MLZ
Heinz Maier-Leibnitz Zentrum

```python
from flask import Blueprint, Flask, request

# Blueprint
bp = Blueprint('bp', __name__)

@bp.route('/')
def my_user_agent():
    # Executing inside request context
    return request.headers['User-Agent']

# Extension
class MyExtension(object):
    def __init__(self, app=None):
        if app:
            self.init_app(app)

    def init_app(self, app):
        app.config.setdefault('MYCONF', True)

# Application
app = Flask(__name__)
ext = MyExtension(app)
app.register_blueprint(bp)
```

Application interfaces

- **Webserver:** WSGI
- **Background worker:** Celery
- **Command-line interface:** Click

Application assembly

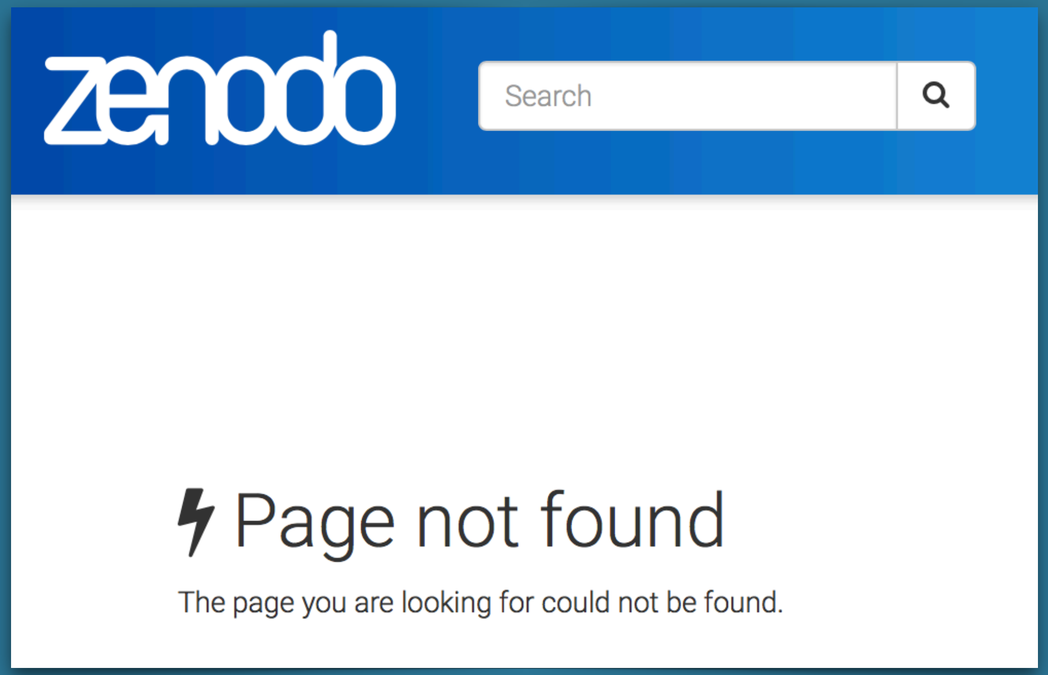
```python
from flask import Flask, Blueprint

# Module 1
bp1 = Blueprint(__name__, 'bp1')
@bp1.route('/')
def hello():
    return 'Hello'

# Module 2
bp2 = Blueprint(__name__, 'bp1')
@bp2.route('/')
def world():
    return 'World'

# Application factory
def create_app():
    app = Flask(__name__)
    app.register_blueprint(bp1)
    app.register_blueprint(bp2)
    return app
```

- **Application creation**
- **Configuration loading**
- **URL converter loading**
- **Extension loading**
- **Blueprints loading**

Entry points

```
# setup.py
setup(
    # ...
    entry_points={
        'invenio_config.module': [
            'mysite = mysite.config',
        ],
    }
)
```

WSGI: UI vs REST




```
{
    message: "The requested URL was not found on the serve
    spelling and try again.",
    status: 404
}
```