

# TEMPLATE

Django

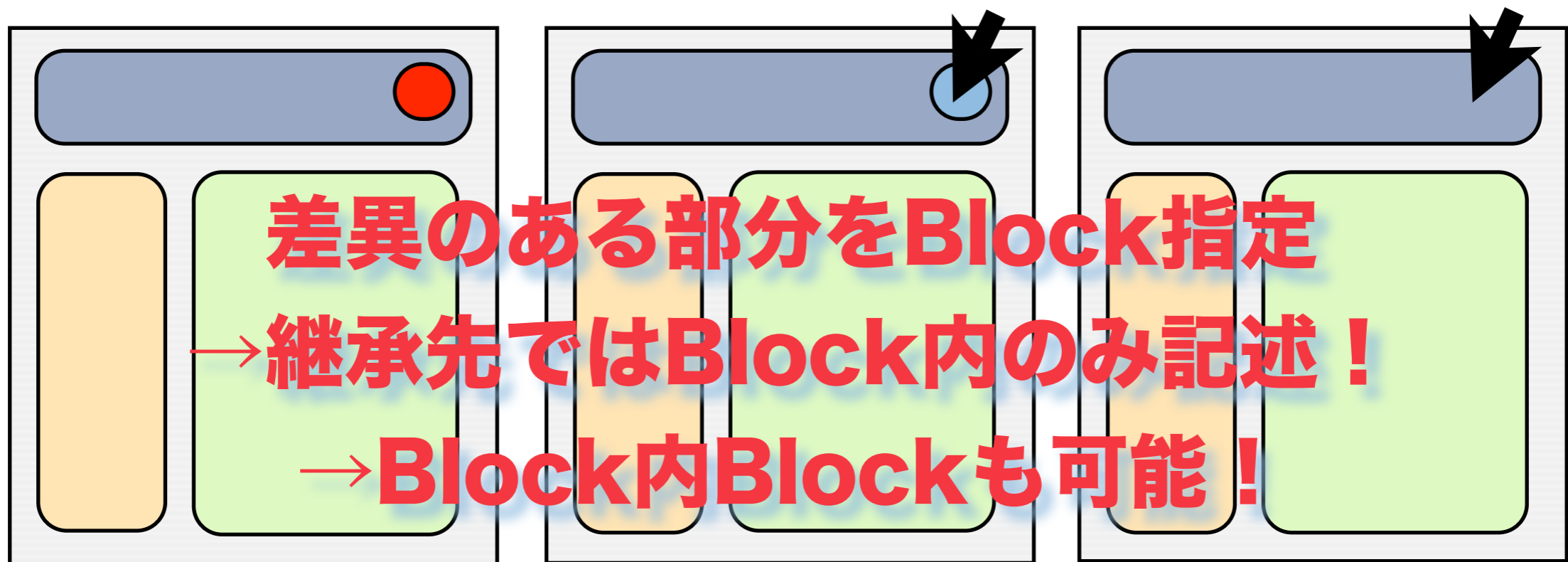
# Pythonic

- 非XML型
  - 実行が早い (kidの4倍、cheetahの6倍だとか)
  - HTMLとテンプレート言語の区別が容易
  - 簡単

```
{% for entry in entry_list %}  
<div class="entry">  
  <h1>{{ entry.title|capfirst }}</h1>  
  <p>{{ entry.body }}</p>  
</div>  
{% endfor %}
```

# Extends

- Include型ではない
  - オーサリングツールは使えない
  - 気分はPhotoshopのレイヤー。概念は理解してもらえそう
  - 問題はデザイナーに実行環境を強制できるか



# O/R Mapper

Django  
(最新開発MR版)

# 基本

- DRY
- 余計なことはしない
  - SQL効率重要
- テーブル自動作成
- 一応リバーースも可
- SQLAlchemyに似てる (インスパイア)

# コード

```
from project.app.models import Model

list = Model.objects.filter(**kwargs)[:10]

obj1 = Model(**kwargs)
obj1.save() // 明示的に保存

obj2 = Model.objects.get(pk=x)
obj2.y = z
obj2.save() //明示的に保存
```

# 対応DB

- PostgreSQL(8系)
- MySQL(5系はまだいまいち)
- SQLite(3系)
- MSSQL(有志によるコントリビュート)
- Oracle (有志によるコントリビュート)

# FORM自動生成

Django

# 管理画面

- Modelの情報を元に自動で生成される
- N対Nとかの対応までこなせる
- 見た目がイイ！
- バリデートに引っかかったフィールド  
毎に色・エラーメッセージが付く
- 国際化も簡単

# Model/Validator

- Modelの型定義がvalidatorに使用される
- こまかいvalidatorもmodelに定義
  - 分散させない

# generic view

- CRUDのロジック
- URLMapperの記述だけでロジック側は大抵完成する
- 面倒なページング処理も自動
- TemplateのscaffoldをModelから生成するヘルパーも有志により作成されている

# AJAX

## Django

# まだ無い！

- 管理画面から対応していく予定
- 有志がDojoに対応した物を作成済
- 議論中

# DOJOToolkit

- 巨大なJavaScriptライブラリ
- 圧縮して送られてくる (格好いい)
- 動的ロードをする
- SafariでもWYSIWYGエディタ
- Ajaxも . . .

本当に必要か？

足回

Django

# 実行環境

- Python2.3以降(True/Falseが必要)
- FastCGIかmod\_python3.x
- デーモン起動かhttpdインスタンスが必要  
→ つらい

# 使えそうなところ

- Python-hosting.com  
(djangoproject.jp)
- Dreamhost (はまり度高し)
- railsplayground.com (求む人柱)
- VPS (7は無くなった・・・)

# 仕事の場合

- サーバは気にしなくてよさそう
- memcacheとかが凄く簡単に使える
- スラド効果も怖くない、らしい
- さいごー