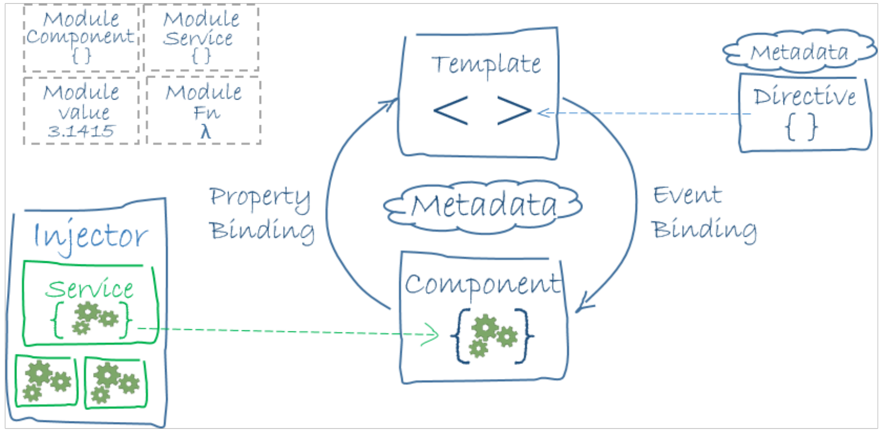
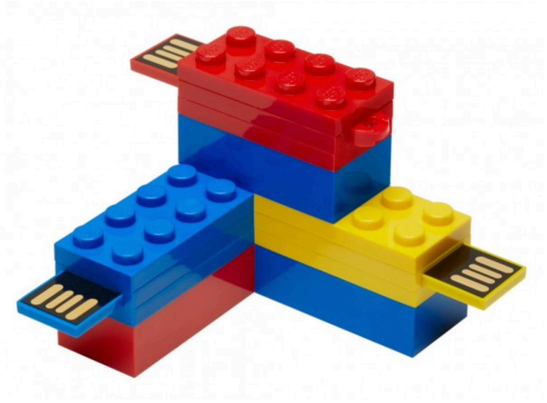
Angular - recap Day 1

Yesterday

- Introduction
- Modules and Components
- Template Syntax

Recap

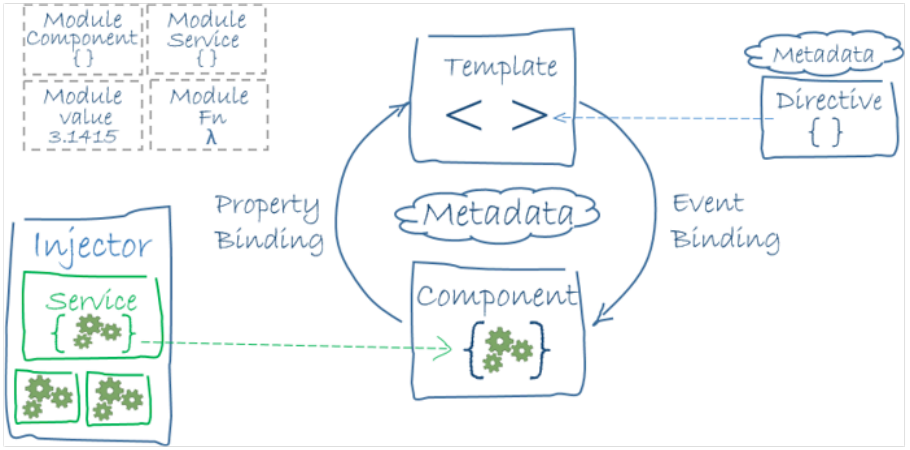




ilionx
Your partner in digital business

What does Angular do?

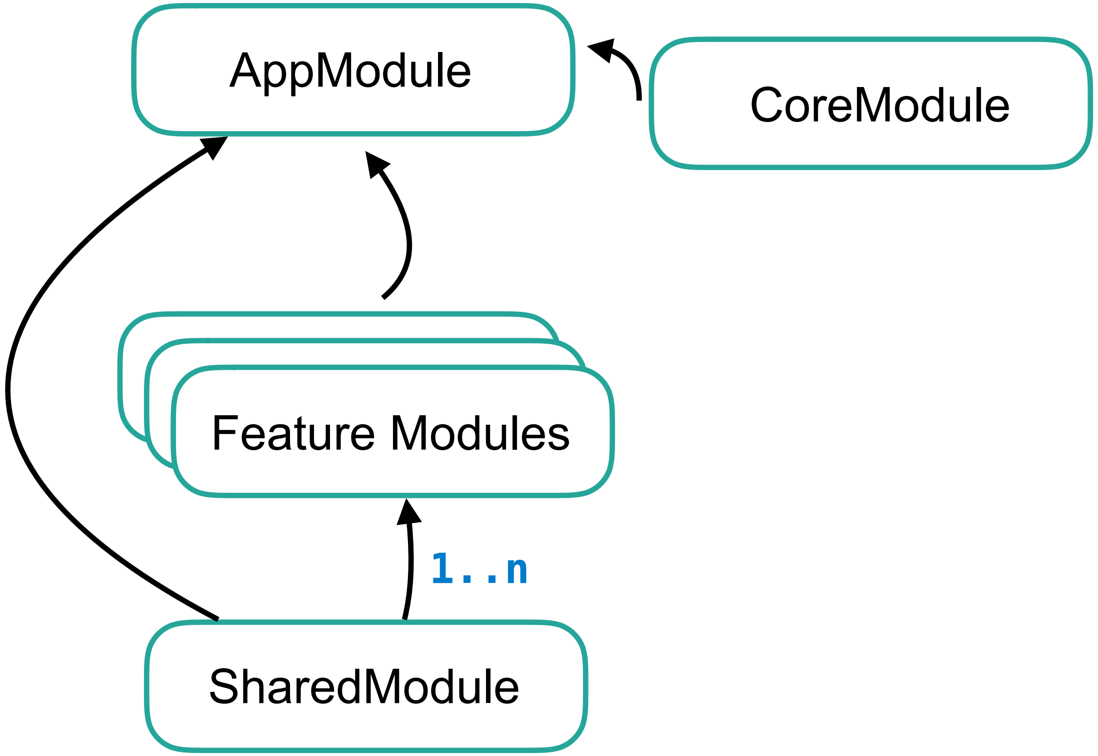
- Bind Classes <-> Templates
- Run Change Detection
- Dependency Injection
- Provide First Class Modules



ilionx
Your partner in digital business

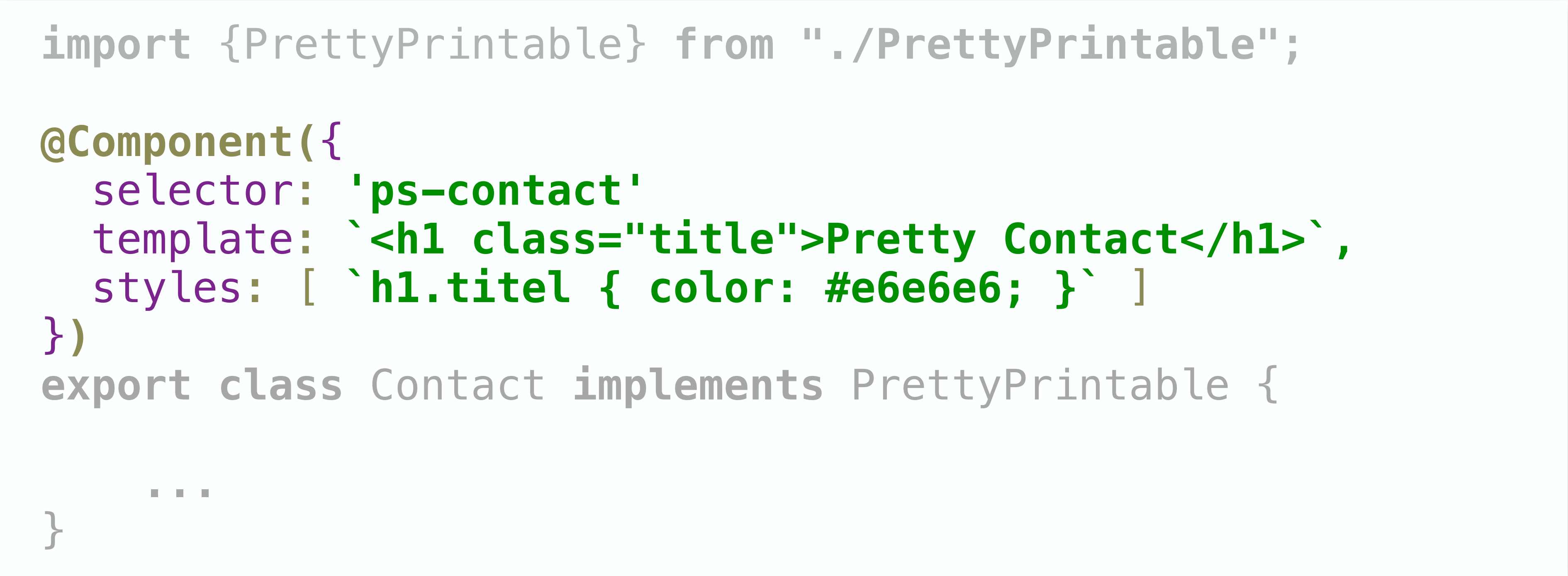
Modules

Modules divide the application into functionality



ilionx
Your partner in digital business

@Component

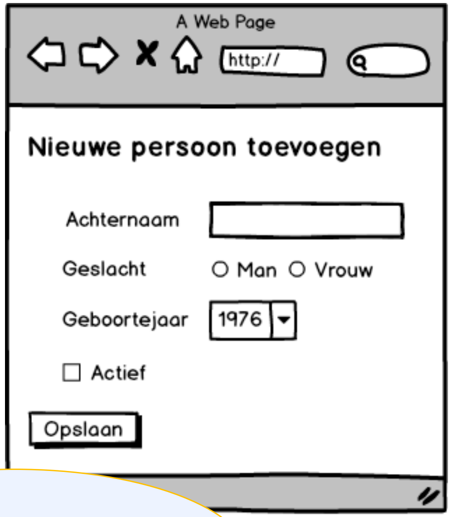


```
import {PrettyPrintable} from "./PrettyPrintable";

@Component({
  selector: 'ps-contact'
  template: `<h1 class="title">Pretty Contact</h1>`,
  styles: [ `h1.titel { color: #e6e6e6; }` ]
})
export class Contact implements PrettyPrintable {

    ...
}
```

Directives



amiq - www.balsamiq.com

component

```
<... *ngFor="...">
<... *ngIf="...">
```

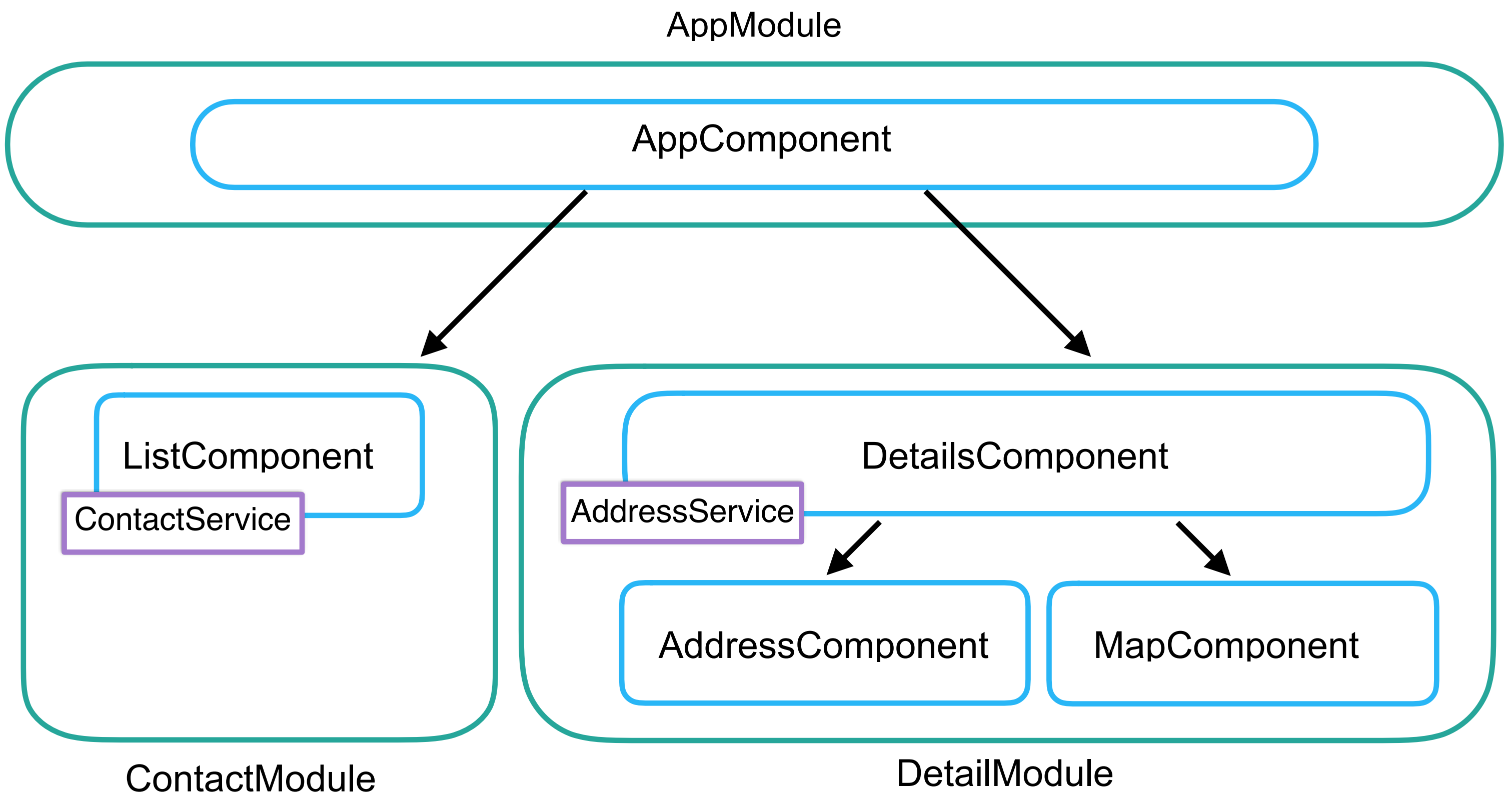
structural directive

```
<... [ngClass]="…">
```

attribute directive

Trees

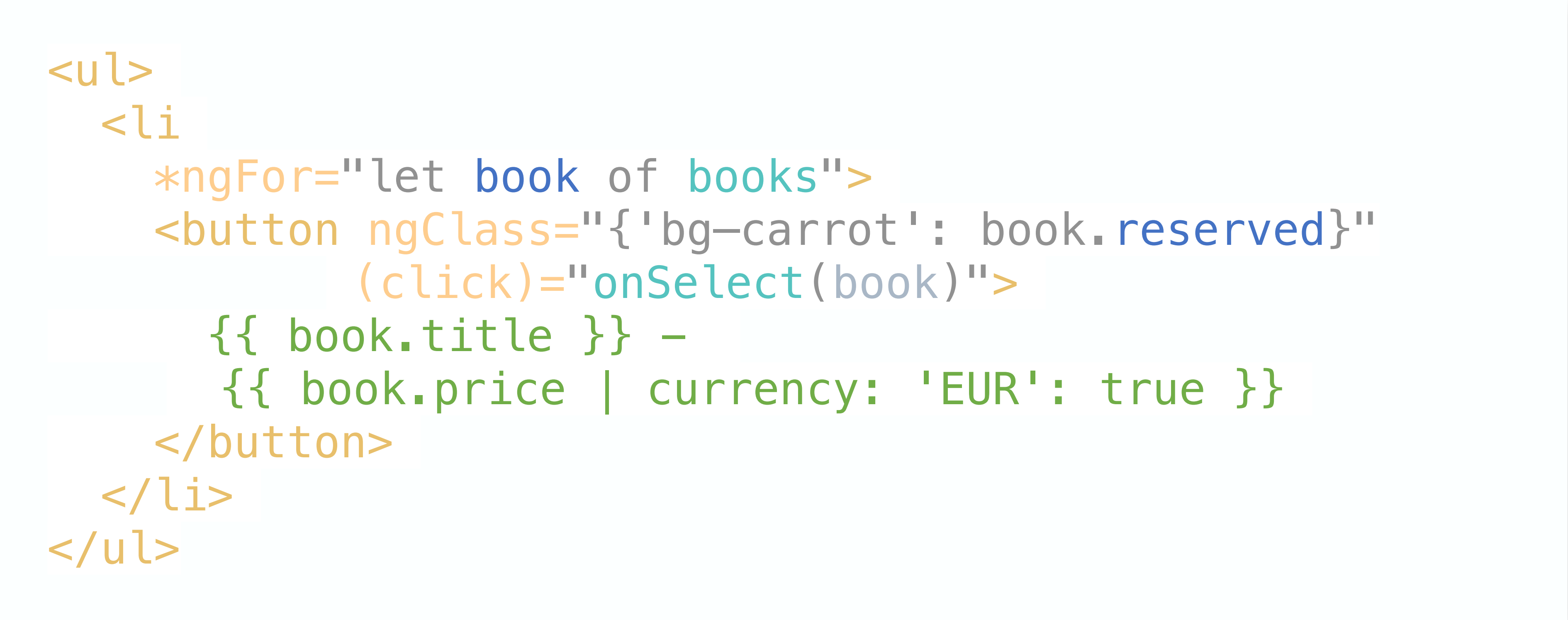
The trees are used for, among other, ChangeDetection



Template syntax toolbox

- {{ interpolation }}
- | pipes
- * structural directives
- attribute directives
- [property] and (event) binding

Template Syntax



```
<ul>
  <li
    *ngFor="let book of books">
    <button ngClass="{'bg-carrot': book.reserved}"
            (click)="onSelect(book)">
      {{ book.title }} -
      {{ book.price | currency: 'EUR': true }}
    </button>
  </li>
</ul>
```

Your partner in digital business

Today

- Dependency Injection And HTTP
- Testing
- Forms

Tomorrow

- Routing
- RxJS
- Material Design Library