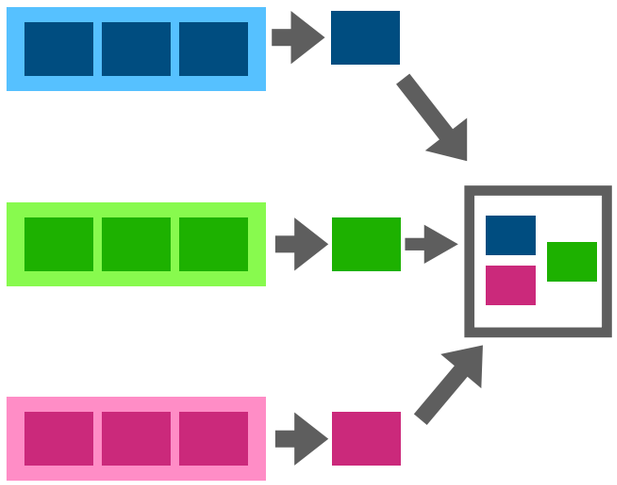
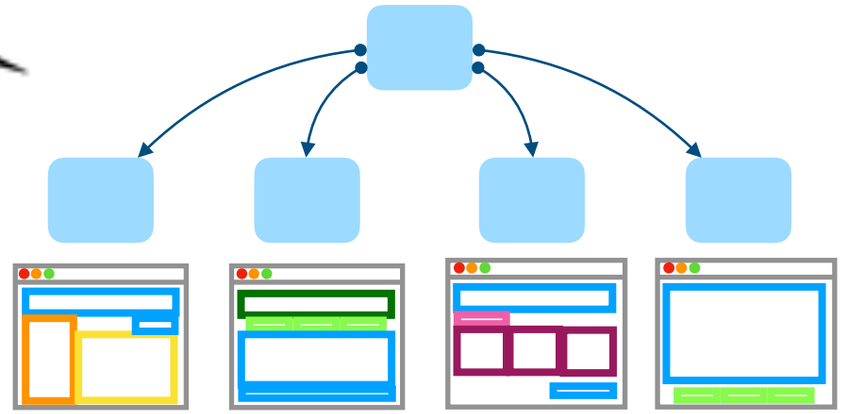
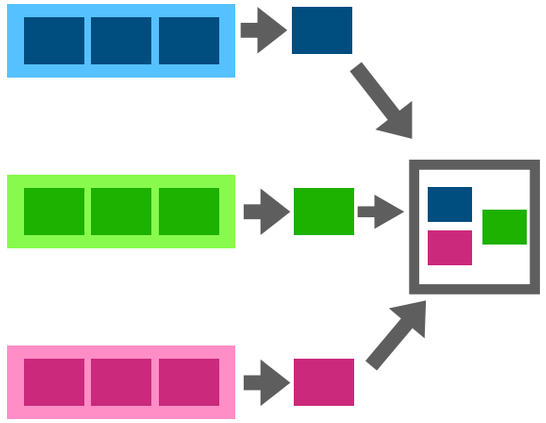
microfrontends

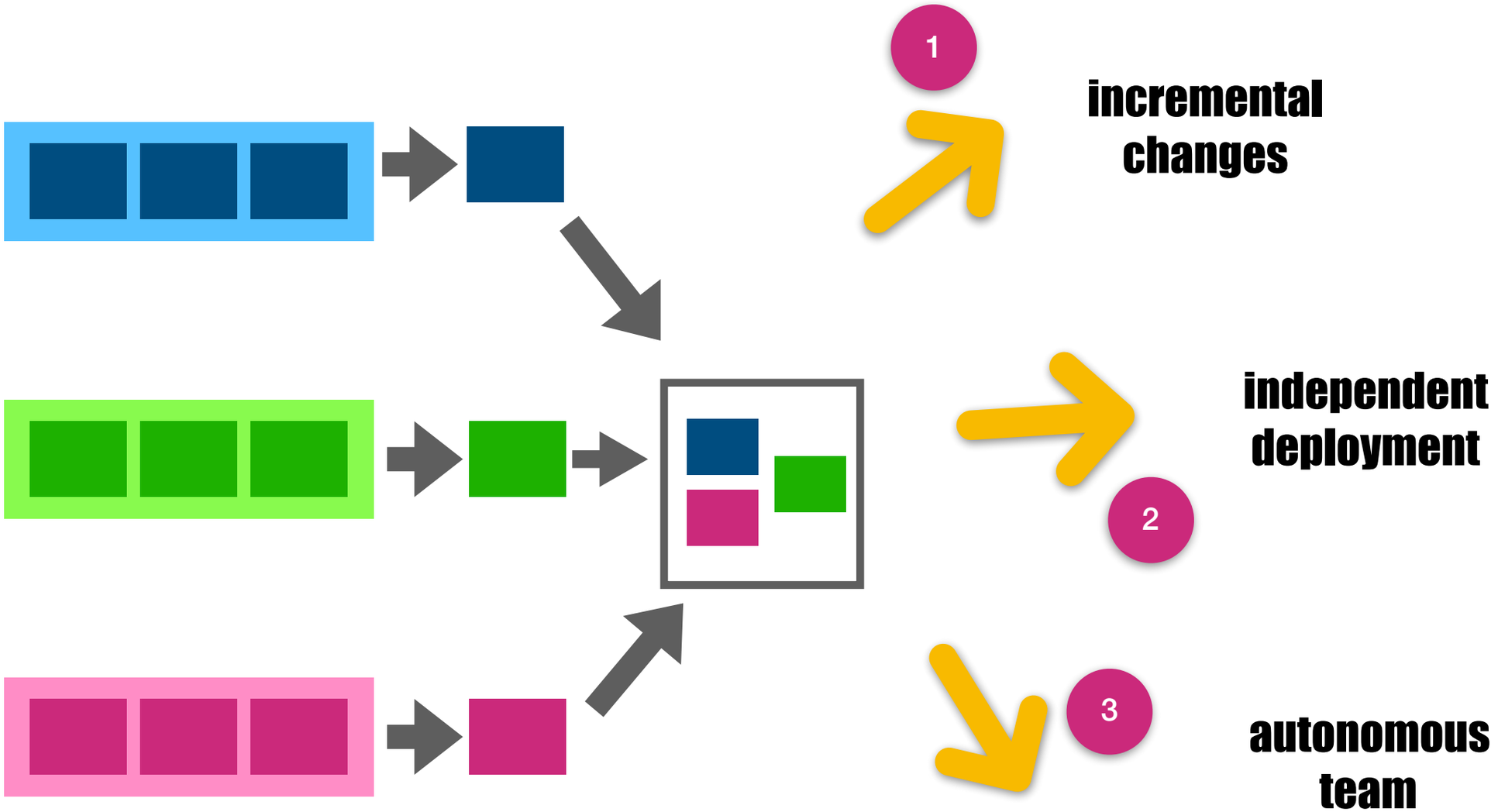


Design system

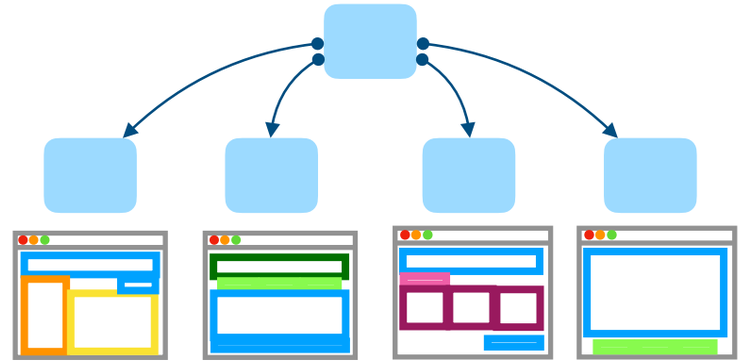


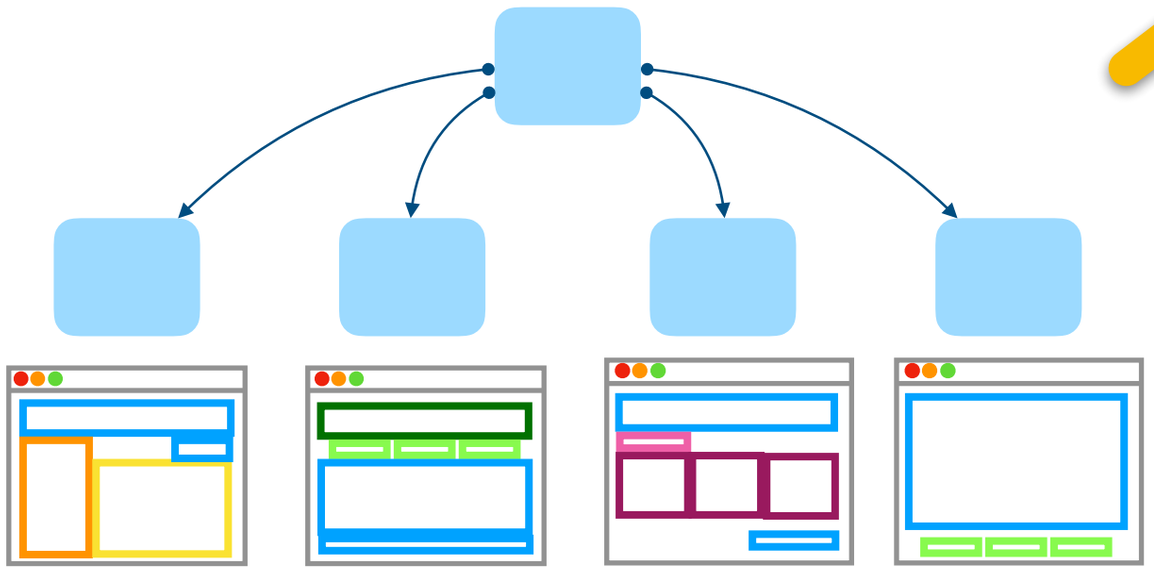
Microfrontends are independently deployable frontend applications that is modeled around business domain




1
incremental changes
independent deployment
2
3
autonomous team

A Design system is collection of reusable components, guided by clear standard, that can be assembled together to build any number of application.





1

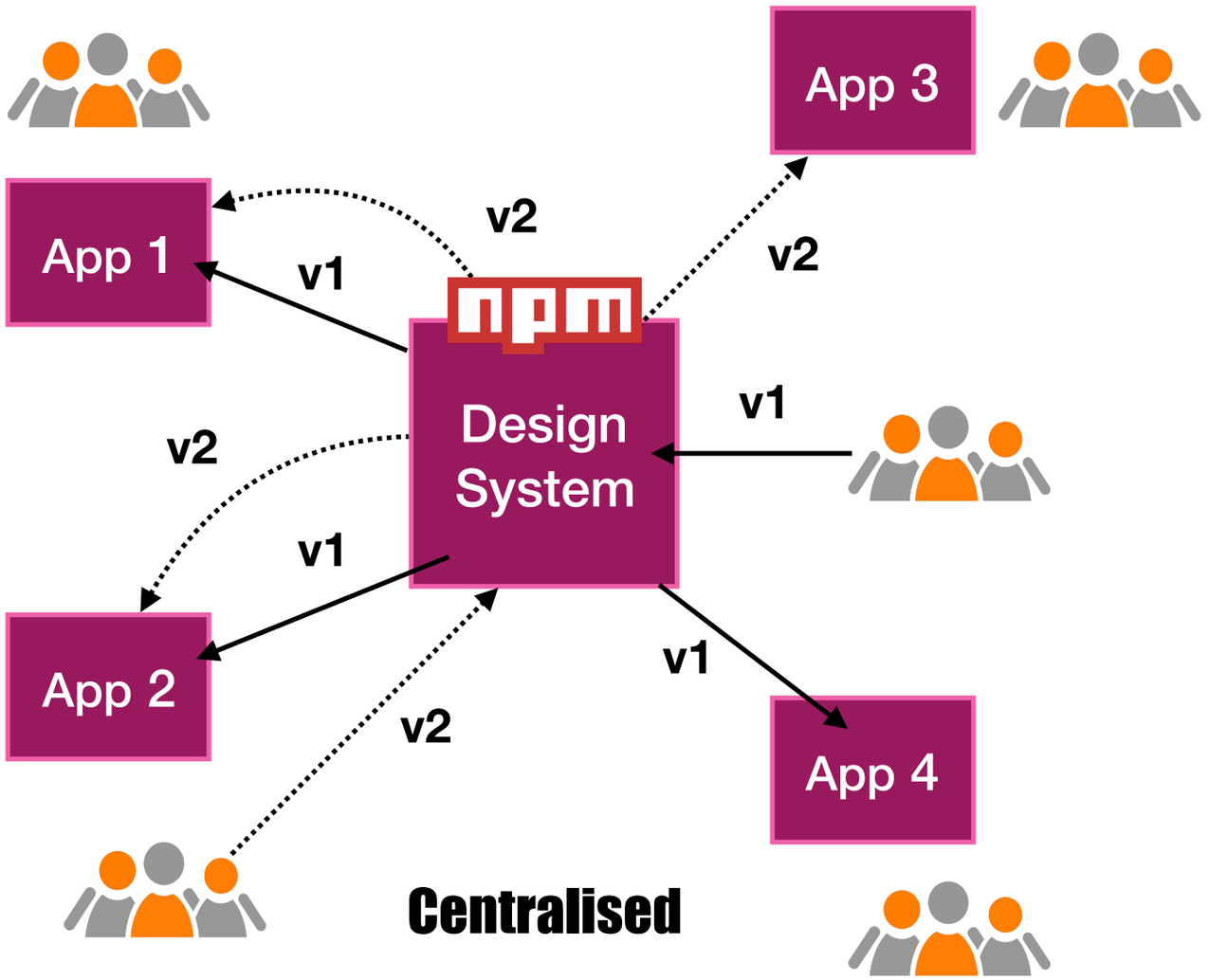
documented components

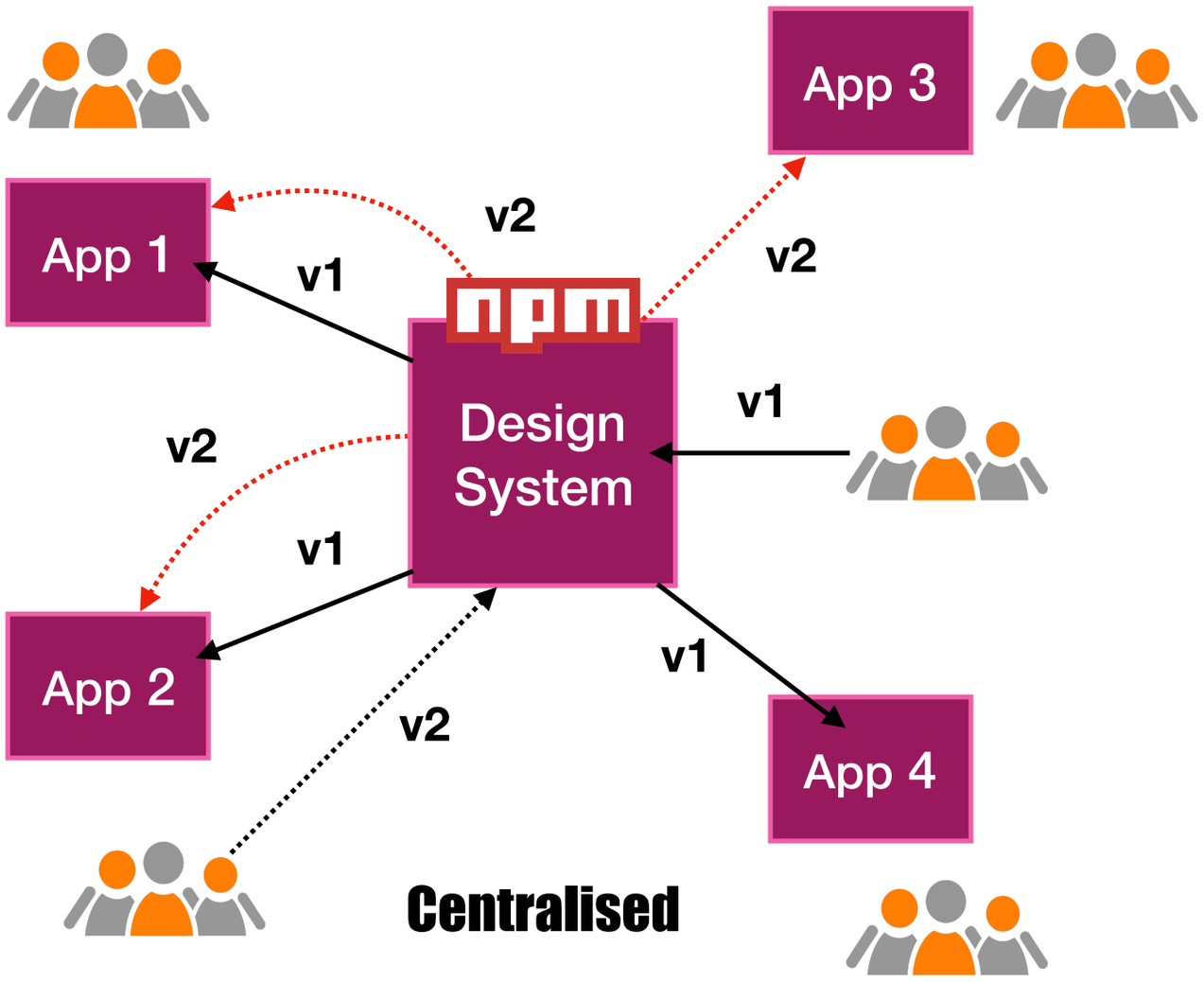
living style guide

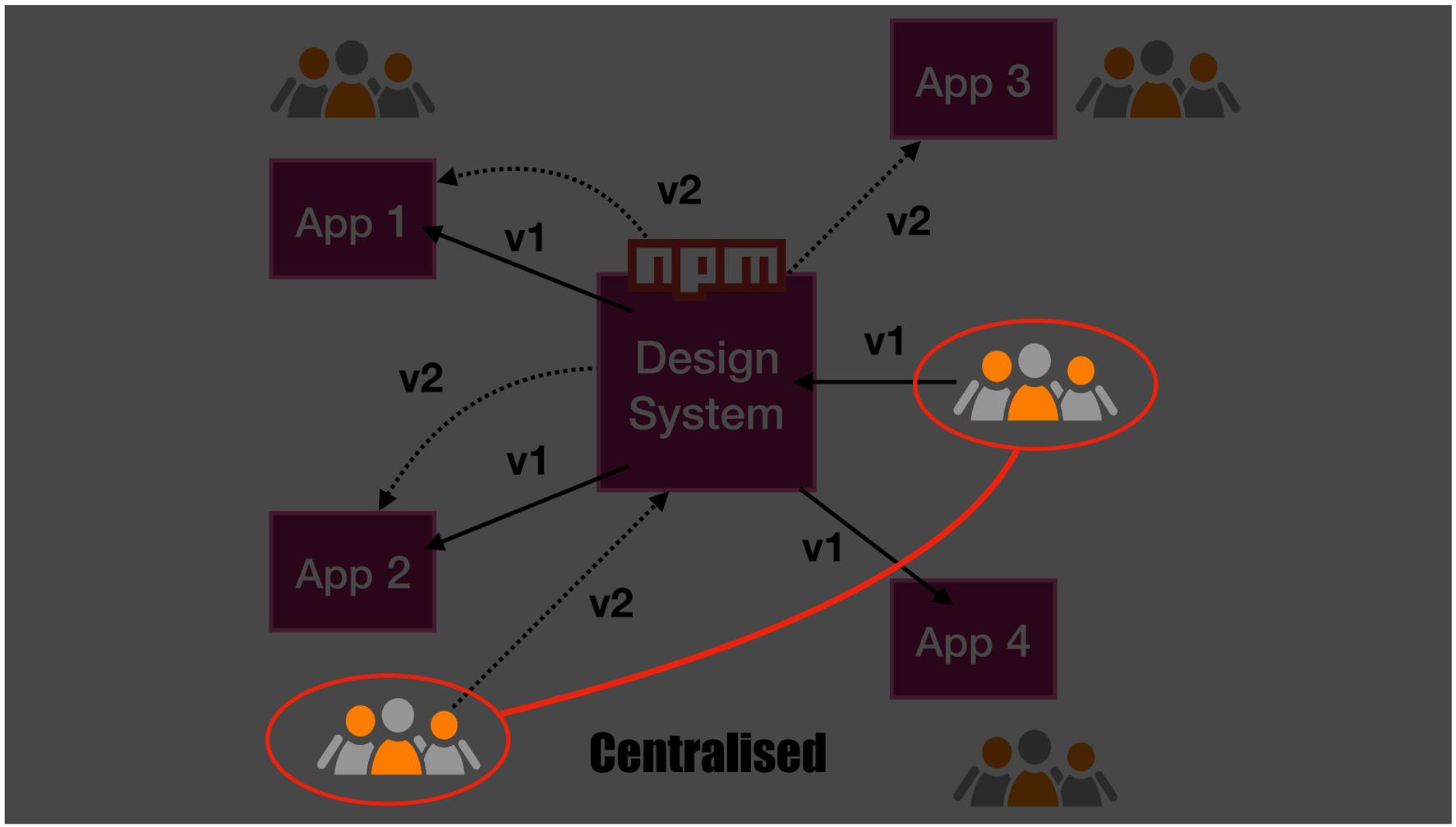
2

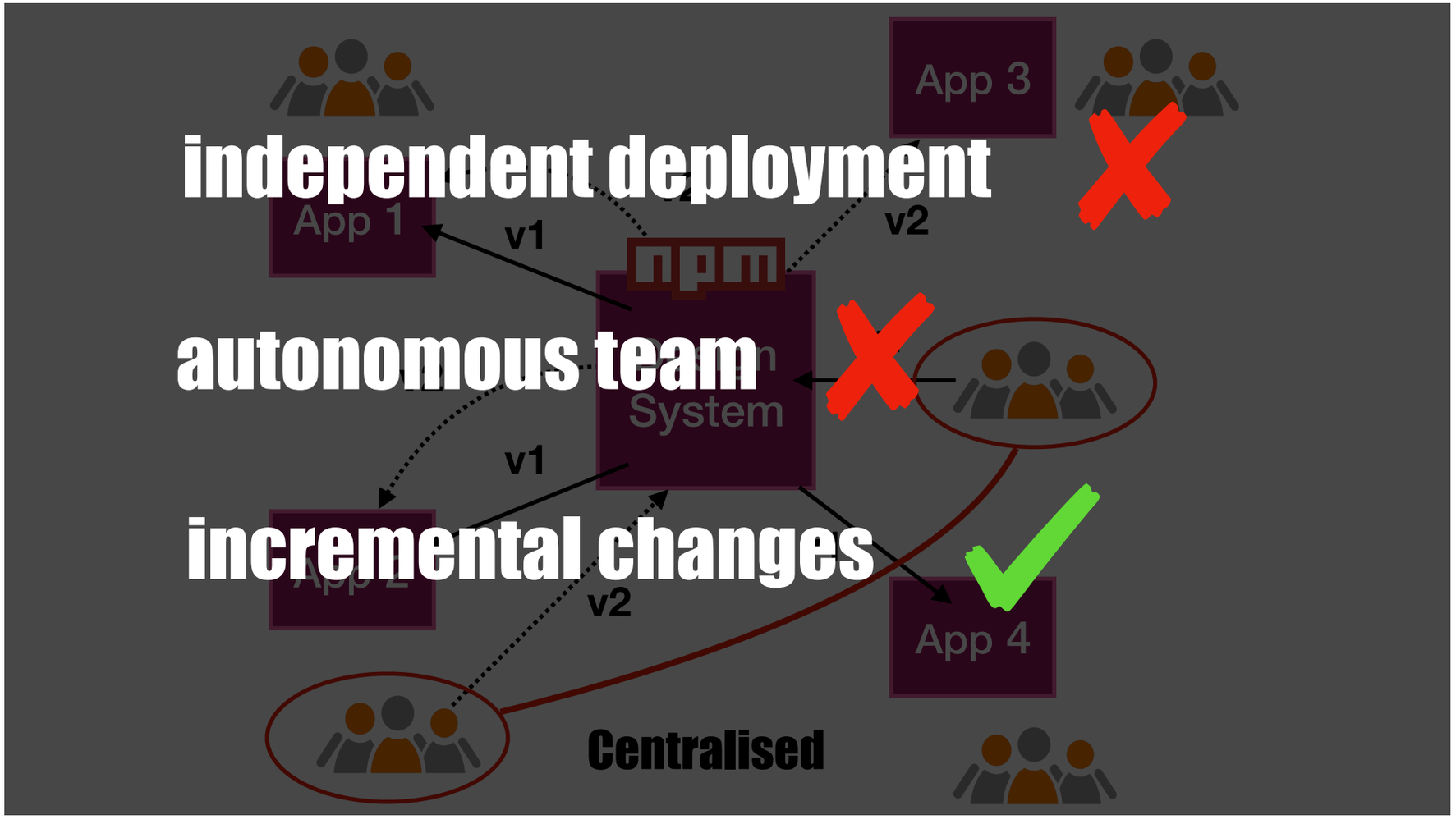
3

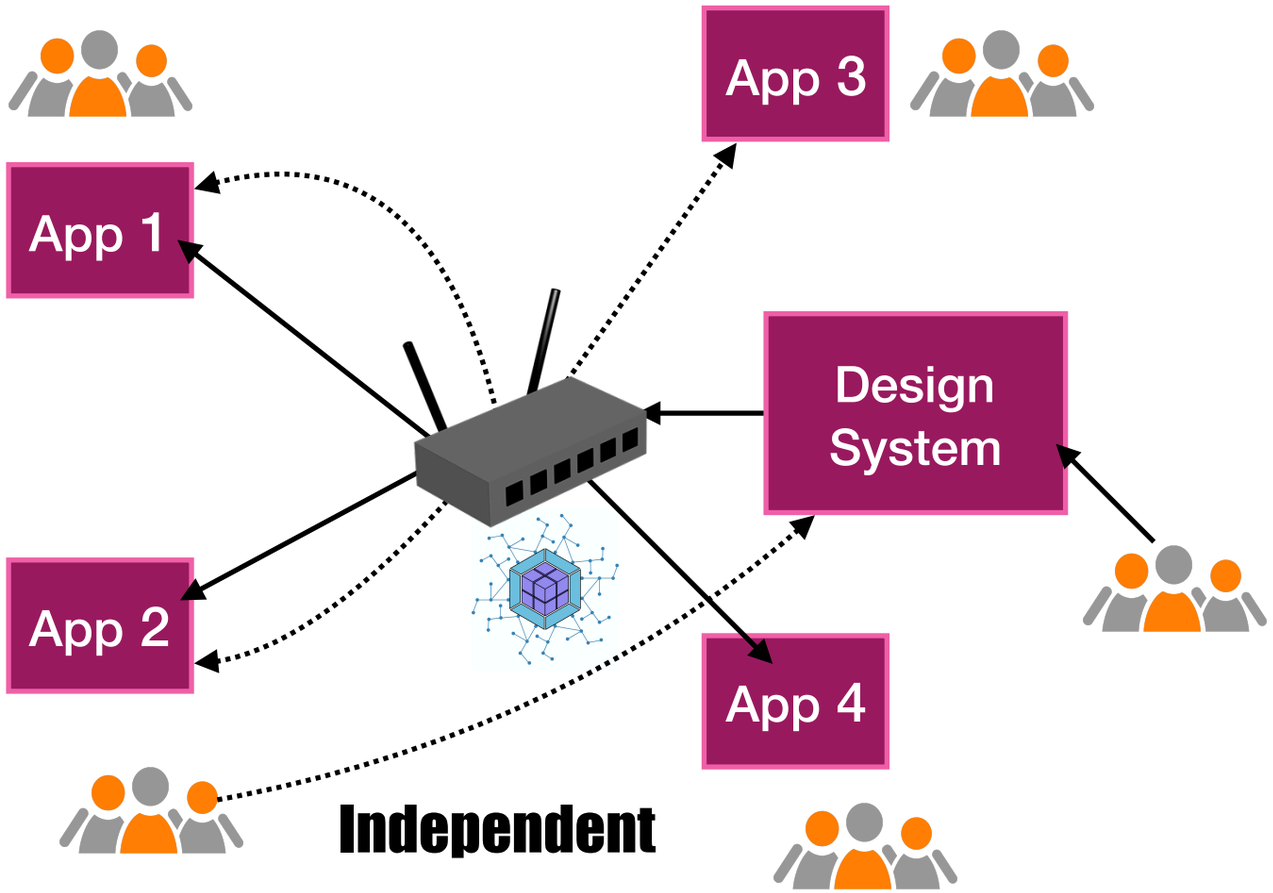
single source of truth

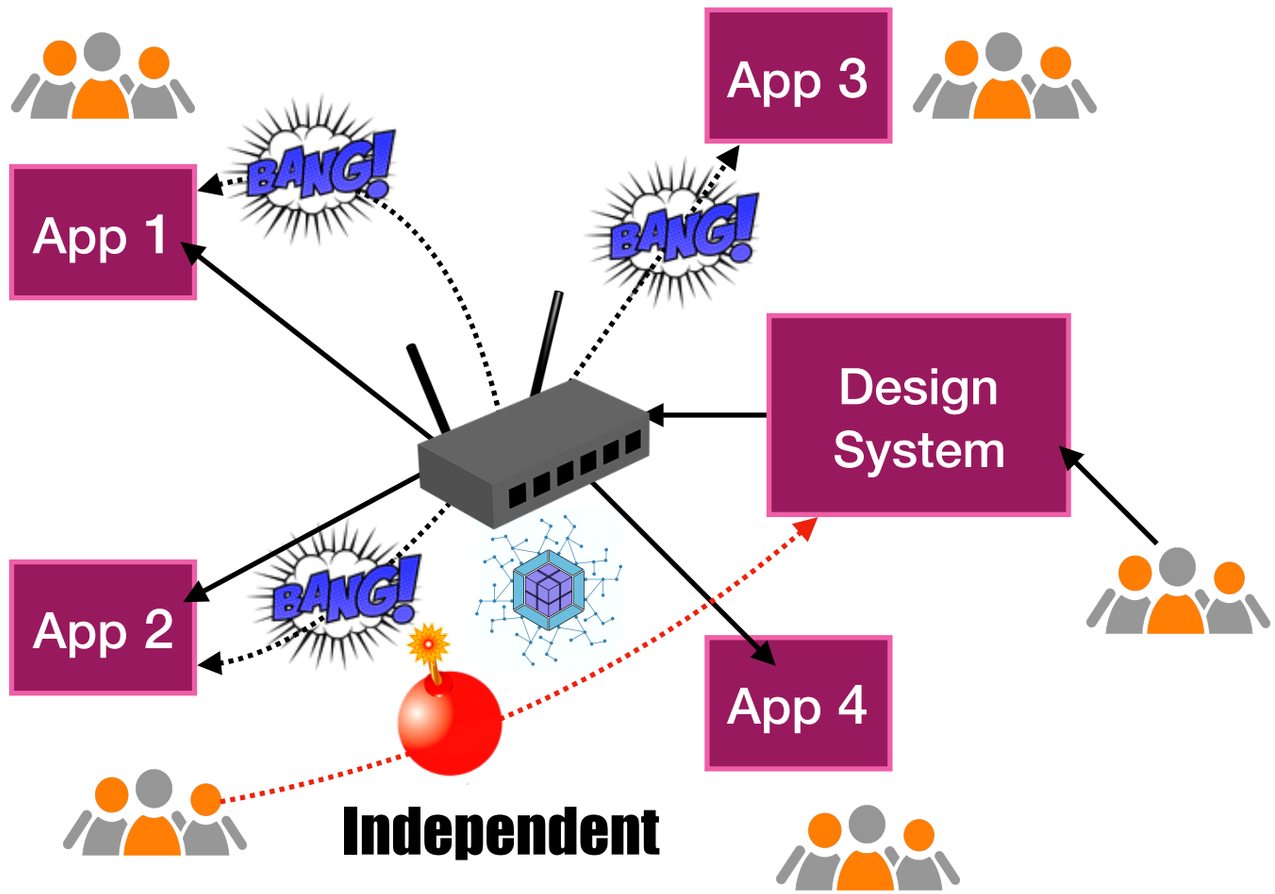

App 1
App 3
v2
v1
v2
npm
Design System
v1
v2
v1
App 2
v1
v2
App 4
Centralised

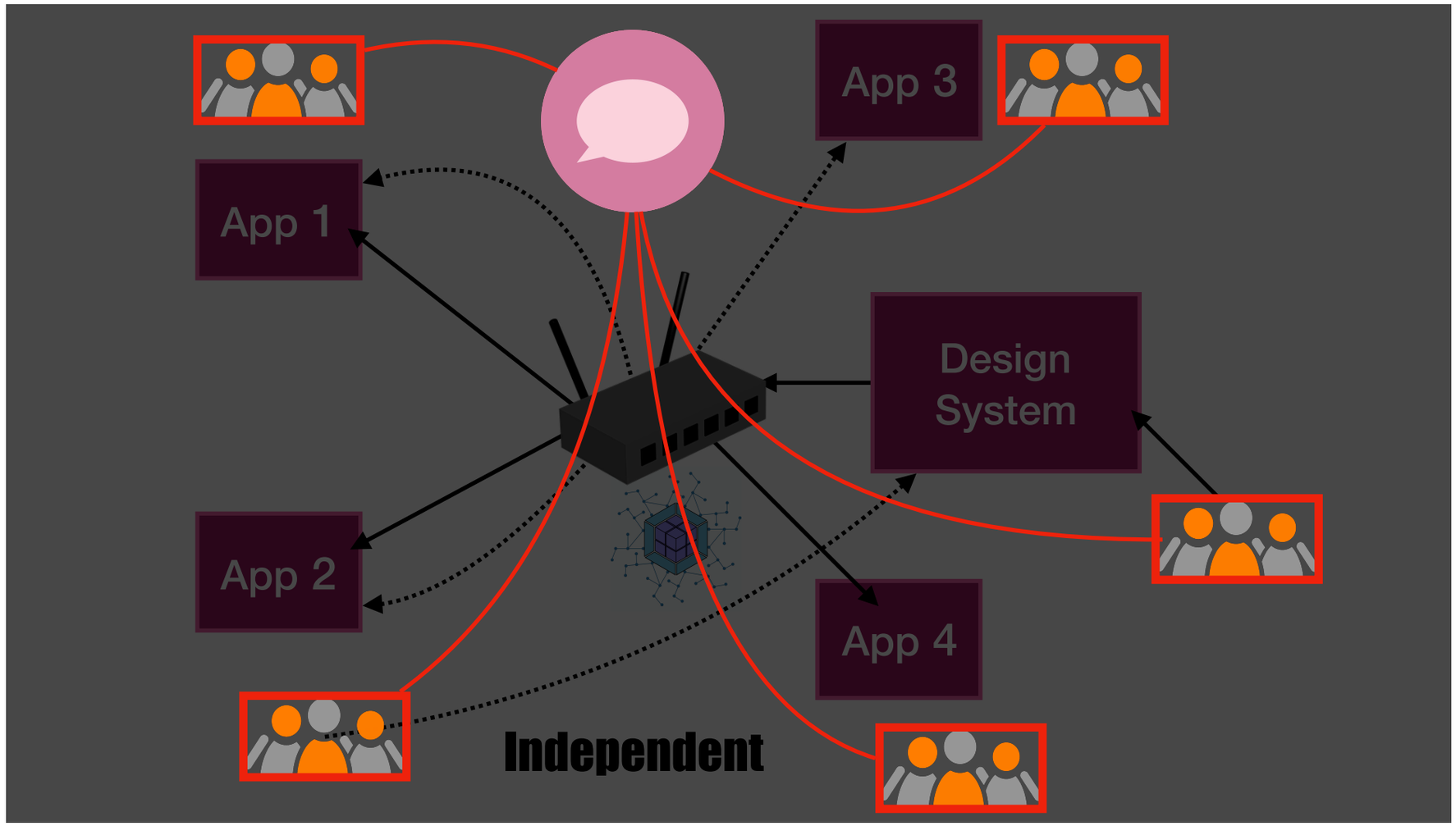

App 1
App 3
v2
v1
v2
npm
Design System
v1
v2
v1
App 2
v1
v2
App 4
Centralised

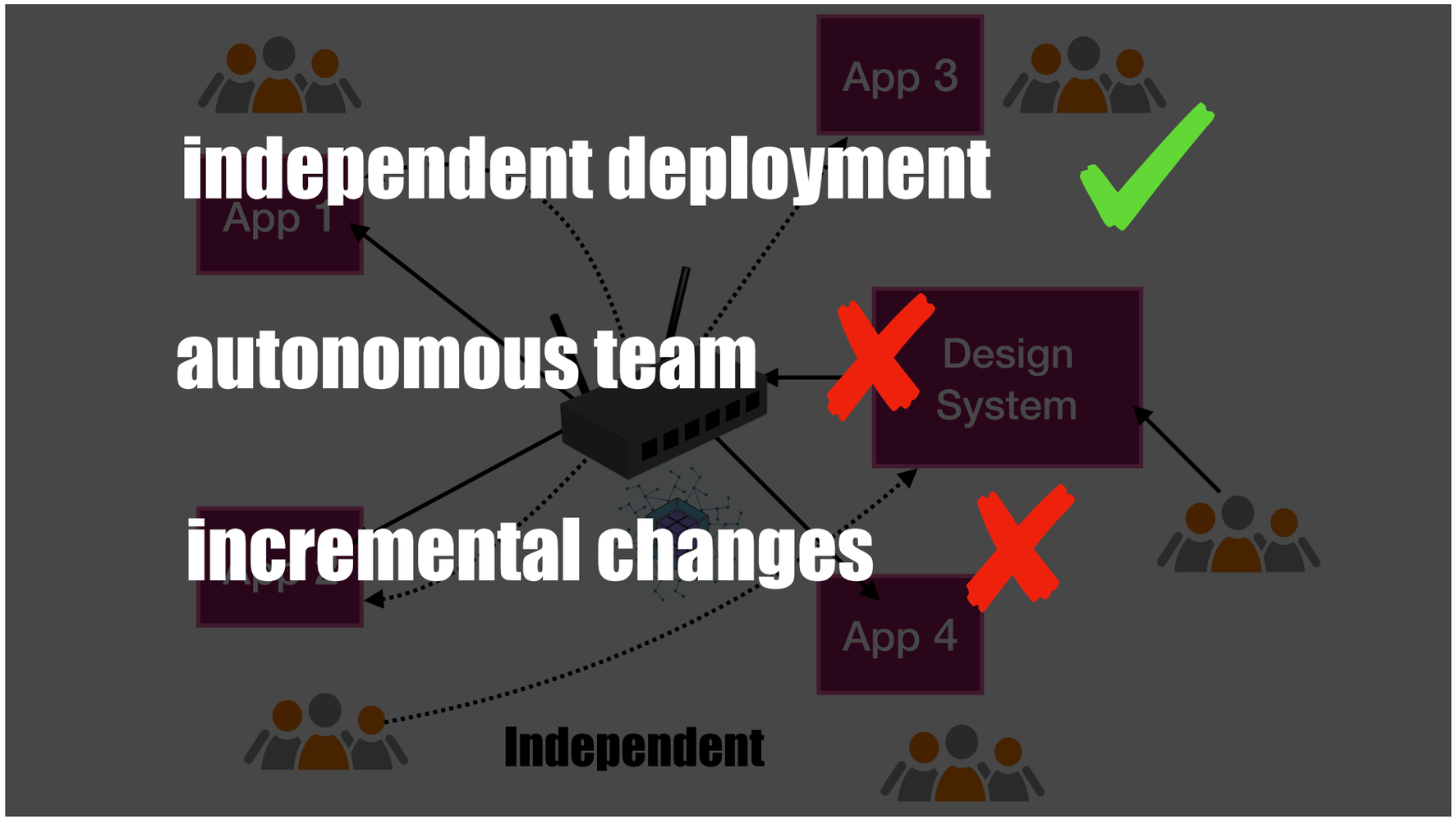

App 1
App 3
v2
v1
v2
npm
Design System
v1
v2
v1
v1
App 2
v2
App 4
Centralised


independent deployment
autonomous team
incremental changes
App 1
App 3
App 4
v1
v2
v1
v2
npm
System
Centralised


App 1
App 3
Design System
App 2
App 4
Independent


App 3
App 1
BANG!
BANG!
Design
System
App 2
BANG!
App 4
Independent

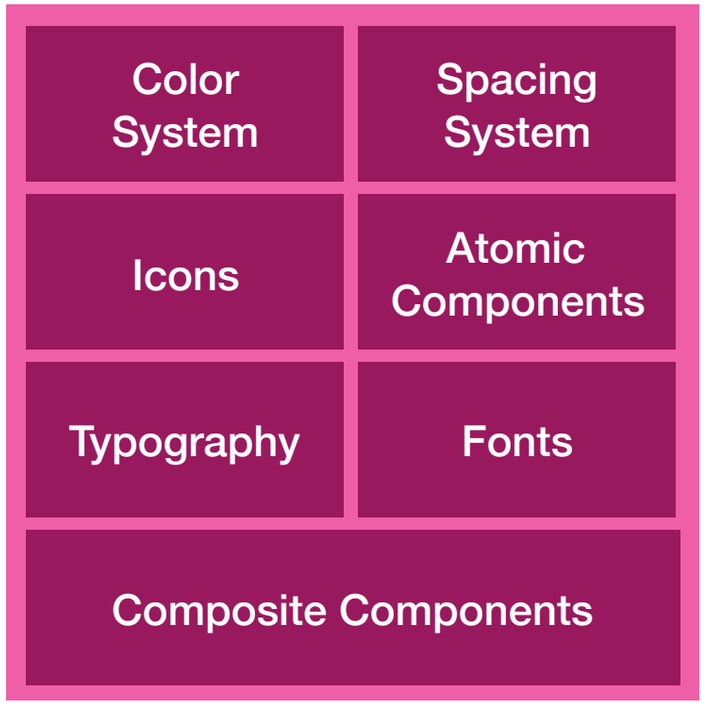

App 3
App 1
Design
System
App 2
App 4
Independent


App 3
independent deployment
App 1
autonomous team
Design System
incremental changes
App 4
Independent

Headless UI

Compound Component

Provider Design System



Consumer Design System

Abstraction

UI Logics &
Interaction handling

Provider

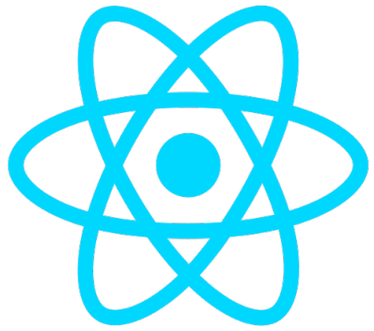
Implementation

UI look & feel

Consumer

Headless UI

```
function DropdownCombobox() {
    const [inputItems, setInputItems] = useState(items)
    const {
      isOpen,
      getToggleButtonProps,
      getLabelProps,
      getMenuProps,
      getInputProps,
      getComboboxProps,
      highlightedIndex,
      getItemProps,
    } = useCombobox({
      ...
    })
    return (...)
  }
  return <DropdownCombobox />
}
```



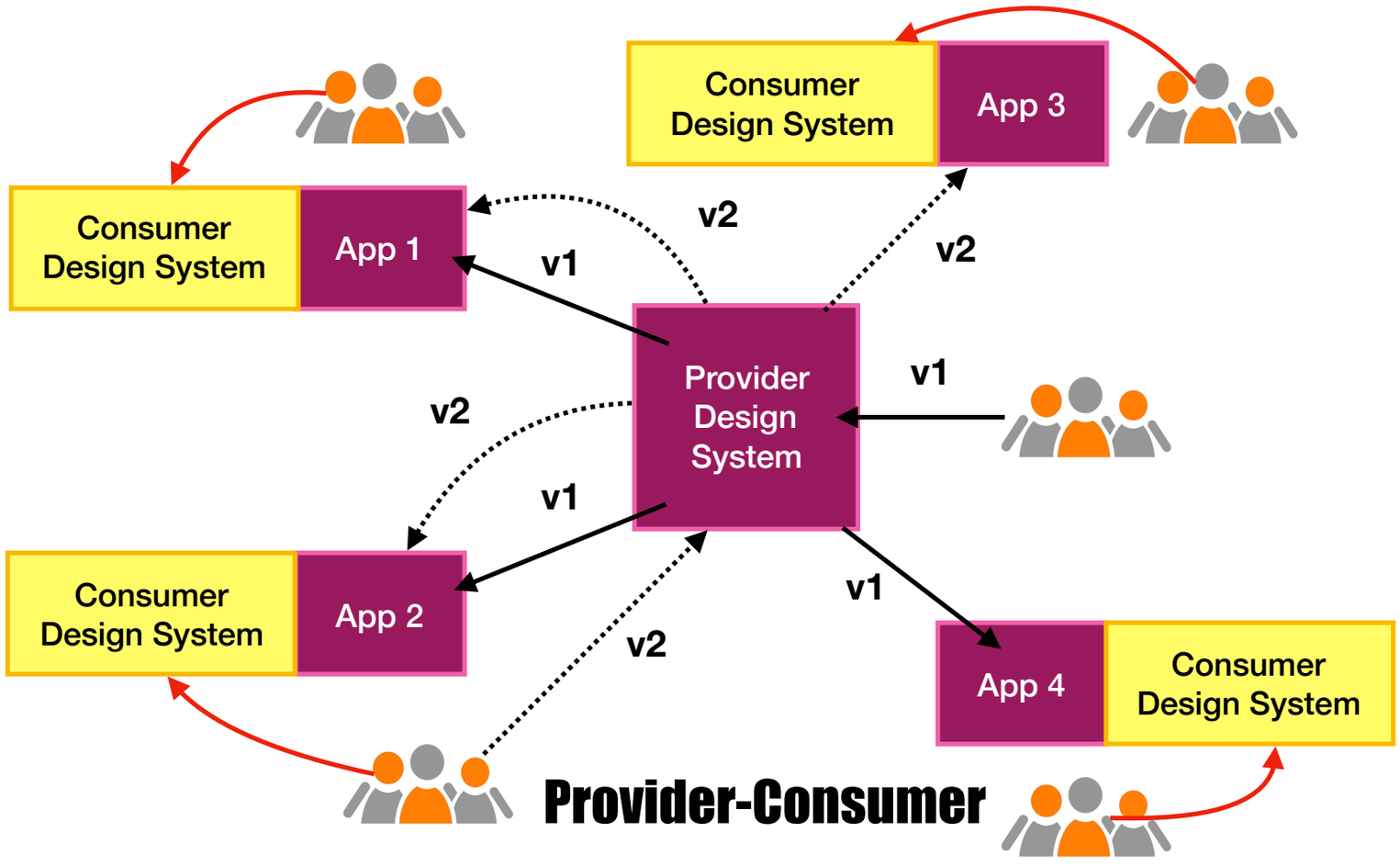
Headless UI

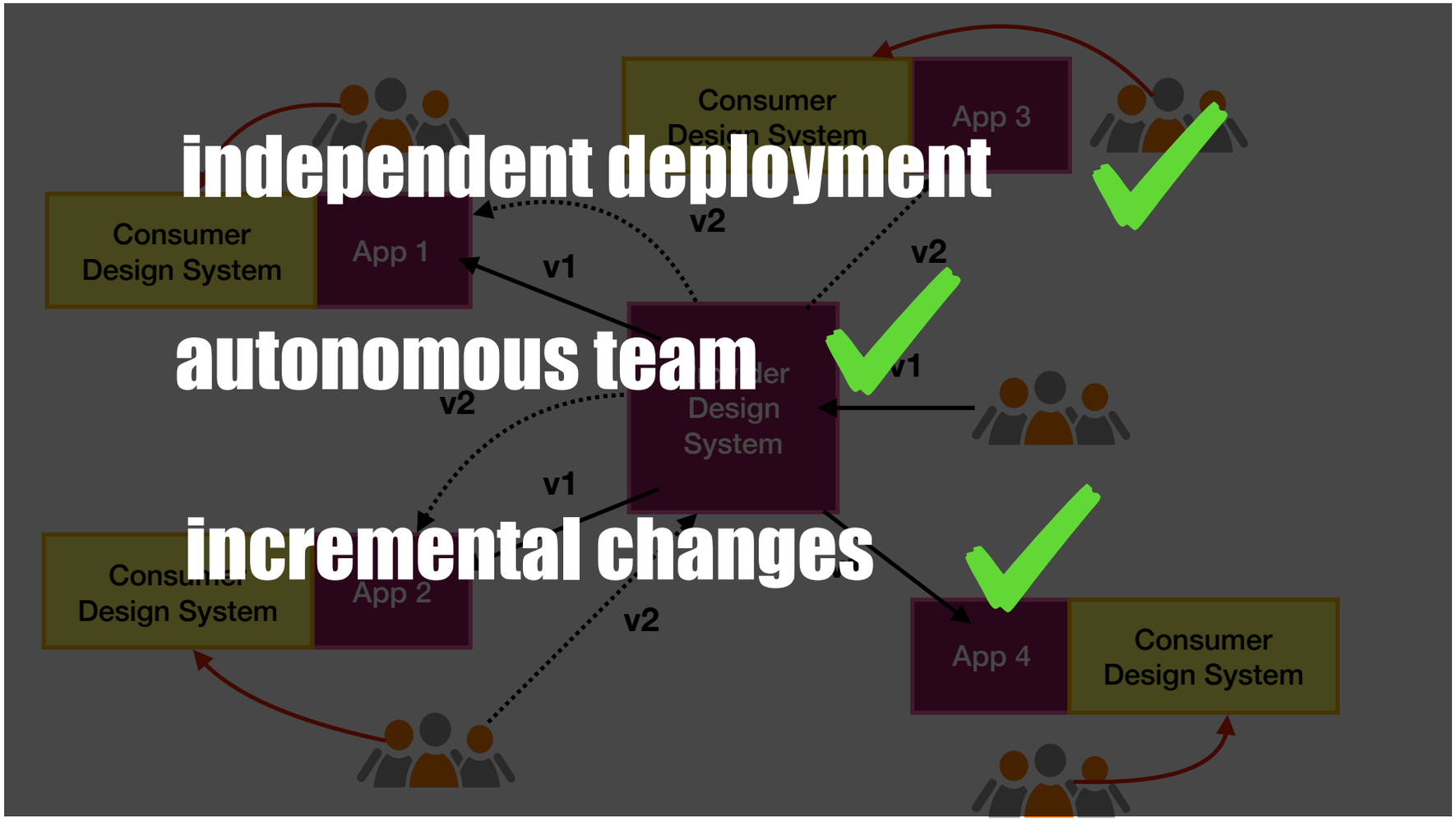
```
<frontal [itemToString]="heroToString">
  <ng-template
    let-value="inputValue"
    let-isOpen="isOpen"
    let-highlightedIndex="highlightedIndex"
    let-selectedItem="selectedItem">

    <label frontalLabel>Select your hero:</label>
    <input type="text" frontalInput/>

    <ul *ngIf="isOpen" frontalList>
      <li *ngFor="..." frontalItem
        [value]="hero"
        [class.highlight]="highlightedIndex === index">
        {{ hero.name }}
      </li>
    </ul>

  </ng-template>
</frontal>
```

Headless UI

```
<template>
  <vue-combo-blocks
    v-model="selected"
    :itemToString="itemToString"
    :items="filteredList"
    @input-value-change="updateList"
    v-slot="{
      getInputProps,
      getInputEventListeners,
      hoveredIndex,
      isOpen,
      getMenuProps,
      getMenuEventListeners,
      getItemProps,
      getItemEventListeners,
      getComboboxProps,
      reset,
    }"
  >
    ...
</vue-combo-blocks>
</template>
```

Headless UI


Consumer
Design System
App 1
Consumer
Design System
App 3
Provider
Design
System
Consumer
Design System
App 2
App 4
Consumer
Design System
v1
v2
v2
v1
v2
v1
v1
v2
Provider-Consumer


independent deployment
autonomous team
incremental changes
Consumer Design System
App 1
App 3
App 2
App 4
Consumer Design System
v1
v2