

Le *property wrapper* @Binding

Celui qui liait les vues entre elles

Le problème...

```
@State var myName = "Quentin"  
ViewB(myName)
```



View B

Le problème...

```
@State var myName = "Quentin"
```

```
var name = "Quentin"
```

~~Two sources of truth~~



Le *property wrapper* @Binding

La solution à de multiples sources de vérité



The diagram consists of two dashed-line rectangular boxes. The left box contains the text `@State var myName = "Quentin"`. The right box contains the text `@Binding var name: String`. A yellow double-headed arrow connects the end of the string "Quentin" in the left box to the start of the variable "name" in the right box, illustrating the bidirectional binding between the two.

```
@State var myName = "Quentin" @Binding var name: String
```

Quand utiliser @State ?



Quand la propriété `@State` n'a **PAS BESOIN**
d'être accessible en dehors de la vue

Quand utiliser @Binding ?

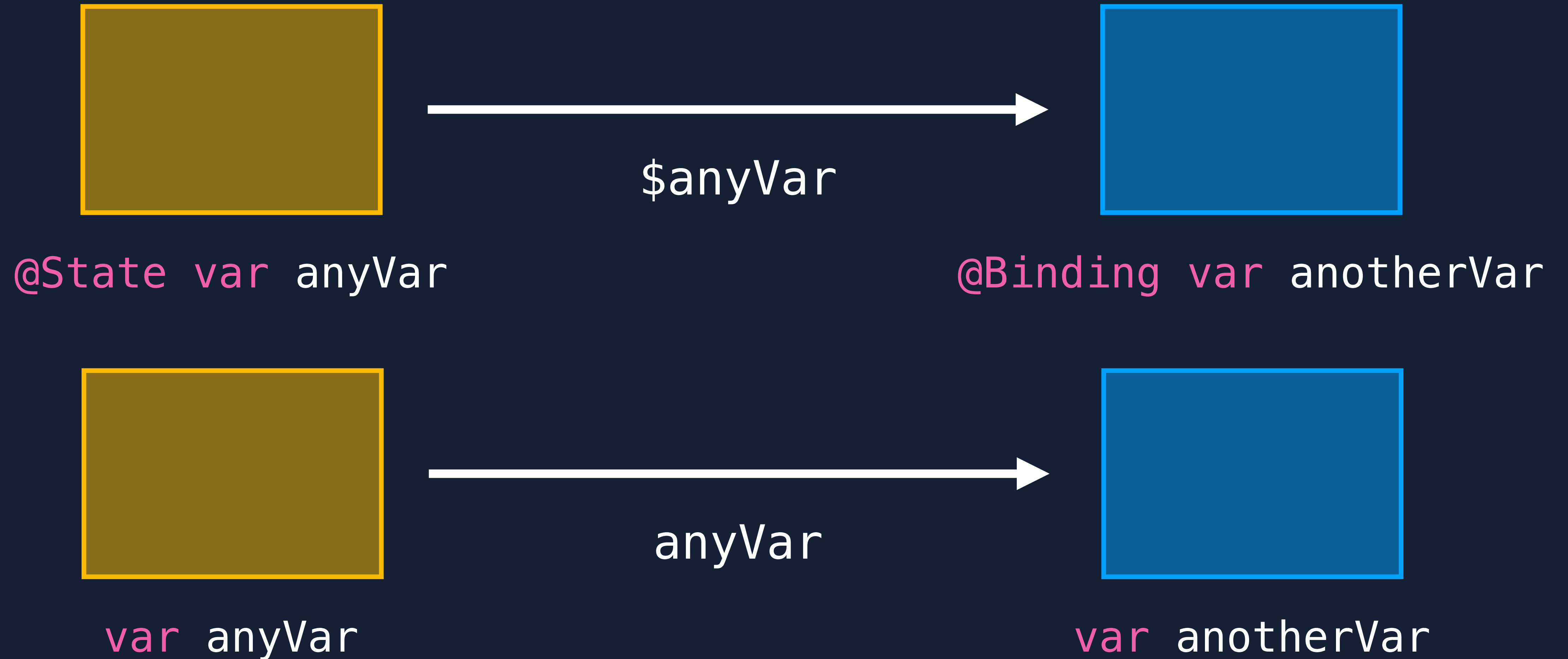


Quand la propriété `@Binding` a **BESOIN** d'une source de donnée externe

Quand utiliser le \$?



Quand utiliser le \$?



Qu'est ce qui change entre les différentes entités d'une vue ?

Cet élément change au cours de la vie du composant ?

Non

var (ou plutôt un let)



```
MonComposant(laProp: "Quentin")  
MonComposant(laProp: "Nico")
```

Oui

@State ou @Binding
@StateObject ou @ObservedObject

Est ce que la donnée appartient
au composant (= donnée interne)

Non

@Binding
@ObservedObject

Oui

@State
@StateObject