

Les conteneurs

- Un conteneur est un objet qui contient des références d'objets de même type dits composants.
- Il est plus ou moins dynamique
- Sa spécification ne dépend pas du type de ses composants
- On définit un type abstrait conteneur avec
 - Ses opérations de remplissage et vidage
 - Ses opérations pour accéder aux composants
 - Ses opérations entre conteneurs



Le conteneur Vecteur

- Un objet de type **Vecteur** est un conteneur formé de cases indicées. Chaque case contient une référence d'objet
- Le nombre de cases est fixé à la création du Vecteur, en précisant les indices de la première et de la dernière case.
- A la création, les cases contiennent des références qui ne réfèrent sur aucun objet.
- On peut mettre à jour chacune des cases et récupérer leur contenu.



Opérations de base du conteneur Vecteur

Constructeur `creerVecteur(Entier bi, Entier bs) : Vecteur`

Transformateur Vecteur : `affVal(Entier i,Element e) : Vecteur`

Observateur Vecteur : `recVal(Entier i) : Element`

Observateur Vecteur : `estInitialise(Entier i) : Booleen`

Observateur Vecteur : `borneInf() : Entier`

Observateur Vecteur : `borneSup() : Entier`



Les pré-conditions du conteneur Vecteur

- $\text{definie}(\text{creerVecteur}(bi,bs)) \Leftrightarrow bi \leq bs$
- $\text{definie}(v.\text{recVal}(ind)) \Leftrightarrow (v.\text{borneInf}() \leq ind) \text{ et } (ind \leq v.\text{borneSup}())$
- $\text{definie}(v.\text{affVal}(ind)) \Leftrightarrow (v.\text{borneInf}() \leq ind) \text{ et } (ind \leq v.\text{borneSup}())$
- $\text{definie}(\text{estInitialise}(v,ind)) \Leftrightarrow (v.\text{borneInf}() \leq ind) \text{ et } (ind \leq v.\text{borneSup}())$



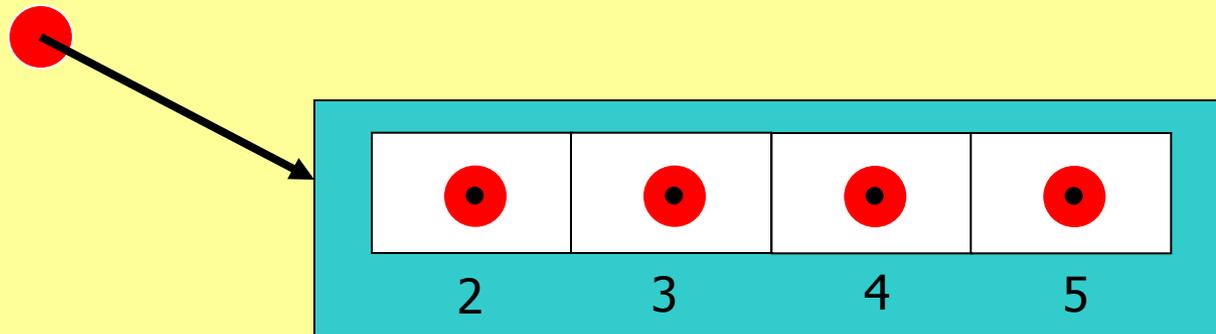
Les post-conditions du conteneur Vecteur

- `creerVecteur(bi, bs).borneInf().estEgal(bi)`
- `creerVecteur(bi, bs).borneSup().estEgal(bs)`
- `non creerVecteur(bi, bs).estInitialise(i)`
- `v.affVal(i,e).recVal(i) = e`
- `v.affVal(i,e).estInit(i)`



L'opération `creerVecteur` du conteneur `Vecteur`

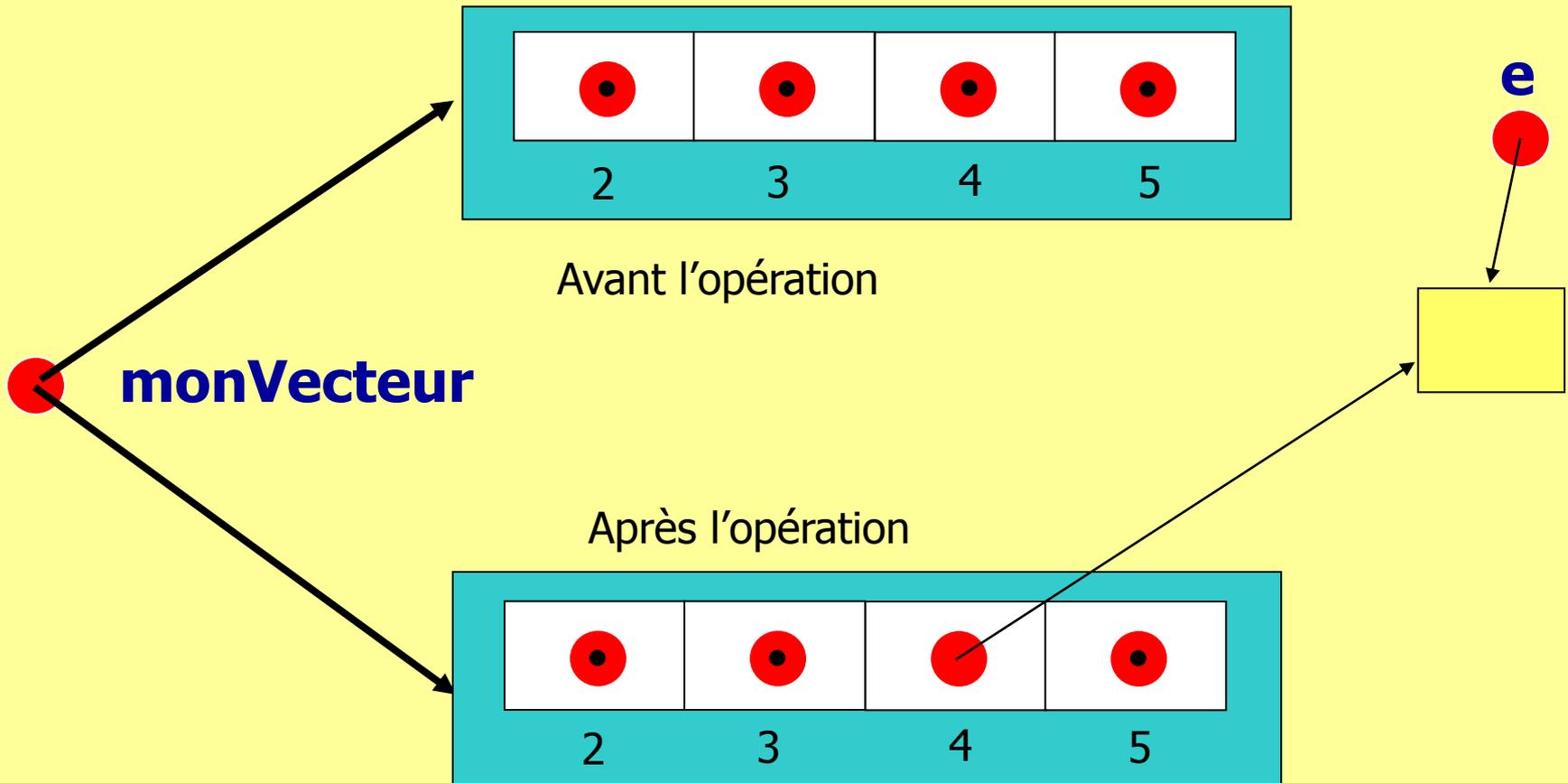
`monVecteur = creerVecteur(2,5)`



On a créé un vecteur de 4 cases indicées de 2 à 5.
Chaque case contient une référence initialisée à `refNulle`



L'opération `affVal` du conteneur `Vecteur`



`monVecteur.affVal(4,e)`



L'opération `recVal` du conteneur `Vecteur`

