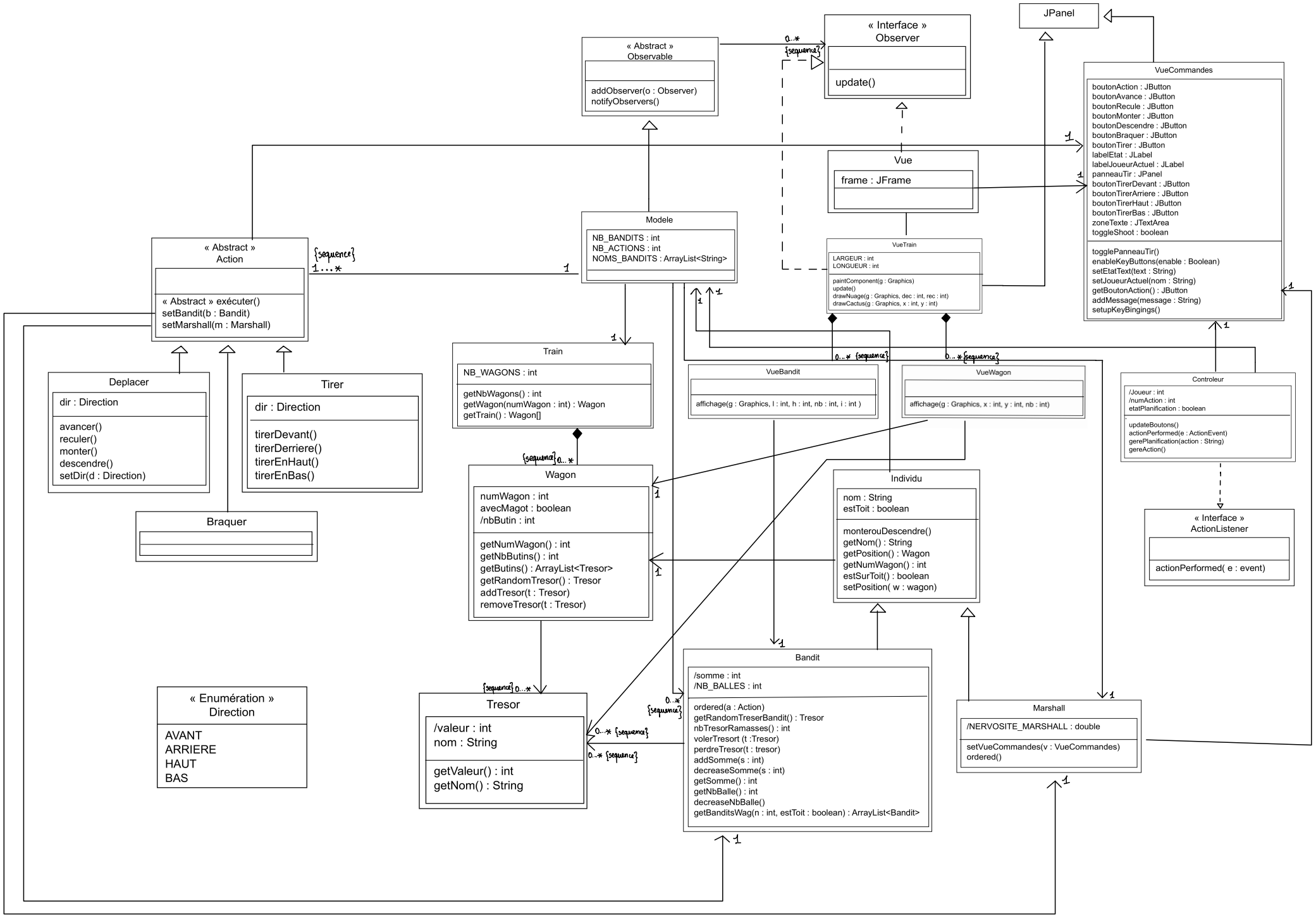
JPanel
« Interface »
Observer
update()
« Abstract »
Observable
addObserver(o : Observer)
notifyObservers()
VueCommandes
boutonAction : JButton
boutonAvance : JButton
boutonRecule : JButton
boutonMonter : JButton
boutonDescendre : JButton
boutonBraquer : JButton
boutonTirer : JButton
labelEtat : JLabel
labelJoueurActuel : JLabel
panneauTir : JPanel
boutonTirerDevant : JButton
boutonTirerArriere : JButton
boutonTirerHaut : JButton
boutonTirerBas : JButton
zoneTexte : JTextArea
toggleShoot : boolean
togglePanneauTir()
enableKeyButtons(enable : Boolean)
setEtatText(text : String)
setJoueurActuel(nom : String)
getBoutonAction() : JButton
addMessage(message : String)
setupKeyBingings()
Vue
frame : JFrame
Modele
NB_BANDITS : int
NB_ACTIONS : int
NOMS_BANDITS : ArrayList<String>
VueTrain
LARGEUR : int
LONGUEUR : int
paintComponent(g : Graphics)
update()
drawNuage(g : Graphics, dec : int, rec : int)
drawCactus(g : Graphics, x : int, y : int)
« Abstract »
Action
« Abstract » exécuter()
setBandit(b : Bandit)
setMarshall(m : Marshall)
{sequence}
1...*
Train
NB_WAGONS : int
getNbWagons() : int
getWagon(numWagon : int) : Wagon
getTrain() : Wagon[]
VueBandit
affichage(g : Graphics, l : int, h : int, nb : int, i : int )
VueWagon
affichage(g : Graphics, x : int, y : int, nb : int)
Controleur
/Joueur : int
/numAction : int
etatPlanification : boolean
updateBoutons()
actionPerformed(e : ActionEvent)
gerePlanification(action : String)
gereAction()
Deplacer
dir : Direction
avancer()
reculer()
monter()
descendre()
setDir(d : Direction)
Tirer
dir : Direction
tirerDevant()
tirerDerriere()
tirerEnHaut()
tirerEnBas()
Braquer
Wagon
numWagon : int
avecMagot : boolean
/nbButin : int
getNumWagon() : int
getNbButins() : int
getButins() : ArrayList<Tresor>
getRandomTresor() : Tresor
addTresor(t : Tresor)
removeTresor(t : Tresor)
Individu
nom : String
estToit : boolean
monterouDescendre()
getNom() : String
getPosition() : Wagon
getNumWagon() : int
estSurToit() : boolean
setPosition( w : wagon)
« Interface »
ActionListener
actionPerformed( e : event)
« Enumération »
Direction
AVANT
ARRIERE
HAUT
BAS
Tresor
/valeur : int
nom : String
getValeur() : int
getNom() : String
Bandit
/somme : int
/NB_BALLES : int
ordered(a : Action)
getRandomTreserBandit() : Tresor
nbTresorRamasses() : int
volerTresort (t :Tresor)
perdreTresor(t : tresor)
addSomme(s : int)
decreaseSomme(s : int)
getSomme() : int
getNbBalle() : int
decreaseNbBalle()
getBanditsWag(n : int, estToit : boolean) : ArrayList<Bandit>
Marshall
/NERVOSITE_MARSHALL : double
setVueCommandes(v : VueCommandes)
ordered()
0..*
1