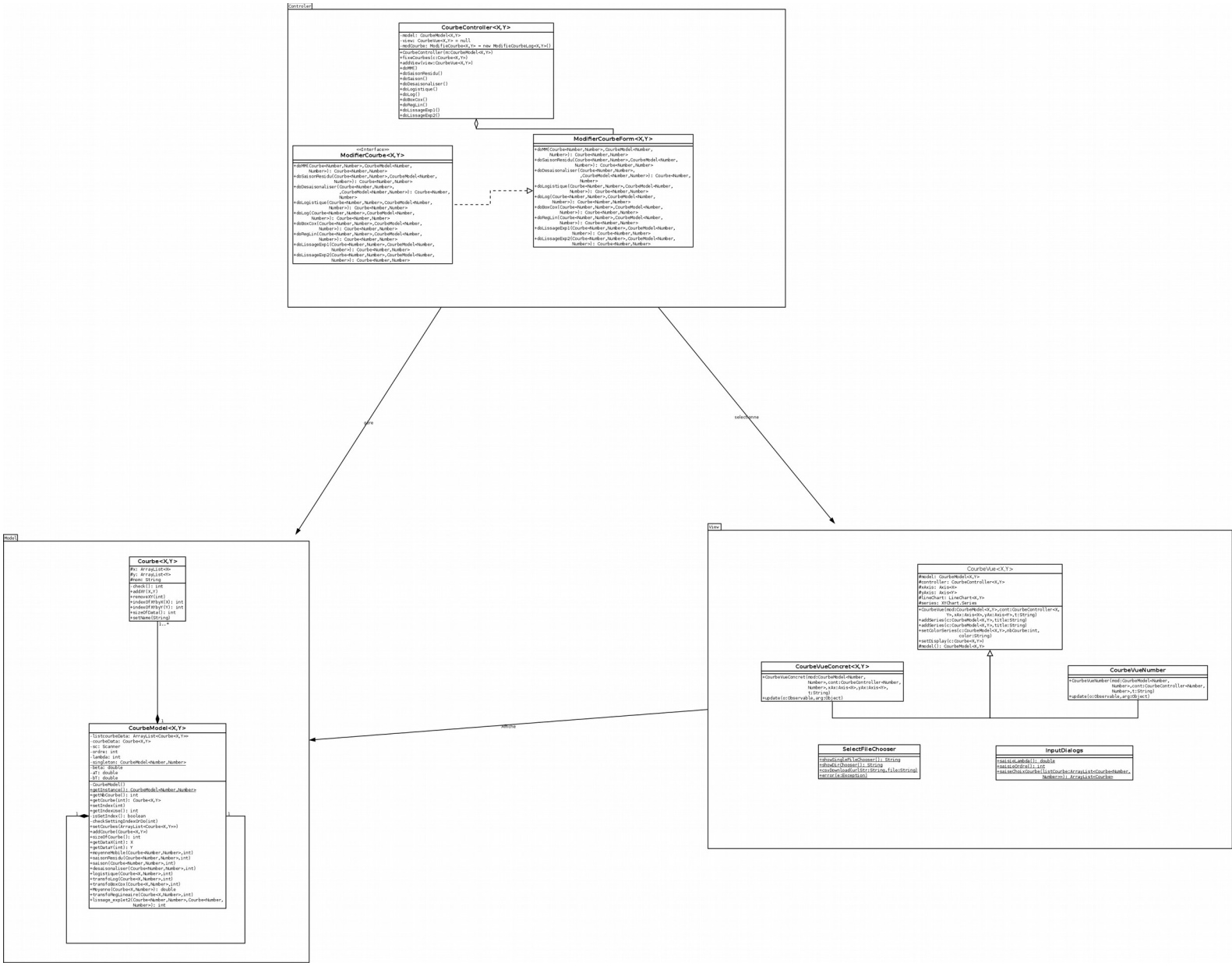
Controler
CourbeController<X,Y>
-model: CourbeModel<X,Y>
-view: CourbeVue<X,Y> = null
-modCourbe: ModifieCourbe<X,Y> = new ModifieCourbeLog<X,Y>()
+CourbeController(m:CourbeModel<X,Y>)
+fixeCourbes(c:Courbe<X,Y>)
+addView(view:CourbeVue<X,Y>)
+doMME()
+doSaisonResidu()
+doSaison()
+doDesaisonaliser()
+doLogistique()
+doLog()
+doBoxCox()
+doRegLin()
+doLissageExp1()
+doLissageExp2()
«Interface»
ModifierCourbe<X,Y>
+doMME(Courbe=Number,Number>,CourbeModel<Number,
Number>): Courbe=Number,Number>
+doSaisonResidu(Courbe=Number,Number>,CourbeModel<Number,
Number>): Courbe=Number,Number>
+doDesaisonaliser(Courbe=Number,Number>,
,CourbeModel<Number,Number>): Courbe=Number,
Number>
+doLogistique(Courbe=Number,Number>,CourbeModel<Number,
Number>): Courbe=Number,Number>
+doLog(Courbe=Number,Number>,CourbeModel<Number,
Number>): Courbe=Number,Number>
+doBoxCox(Courbe=Number,Number>,CourbeModel<Number,
Number>): Courbe=Number,Number>
+doRegLin(Courbe=Number,Number>,CourbeModel<Number,
Number>): Courbe=Number,Number>
+doLissageExp1(Courbe=Number,Number>,CourbeModel<Number,
Number>): Courbe=Number,Number>
+doLissageExp2(Courbe=Number,Number>,CourbeModel<Number,
Number>): Courbe=Number,Number>
ModifierCourbeForm<X,Y>
+doMME(Courbe=Number,Number>,CourbeModel<Number,
Number>): Courbe=Number,Number>
+doSaisonResidu(Courbe=Number,Number>,CourbeModel<Number,
Number>): Courbe=Number,Number>
+doDesaisonaliser(Courbe=Number,Number>,
,CourbeModel<Number,Number>): Courbe=Number,
Number>
+doLogistique(Courbe=Number,Number>,CourbeModel<Number,
Number>): Courbe=Number,Number>
+doLog(Courbe=Number,Number>,CourbeModel<Number,
Number>): Courbe=Number,Number>
+doBoxCox(Courbe=Number,Number>,CourbeModel<Number,
Number>): Courbe=Number,Number>
+doRegLin(Courbe=Number,Number>,CourbeModel<Number,
Number>): Courbe=Number,Number>
+doLissageExp1(Courbe=Number,Number>,CourbeModel<Number,
Number>): Courbe=Number,Number>
+doLissageExp2(Courbe=Number,Number>,CourbeModel<Number,
Number>): Courbe=Number,Number>
gere
selectionne
Model
Courbe<X,Y>
#x: ArrayList<X>
#y: ArrayList<Y>
#nom: String
-iNext(): int
+addXY(X,Y)
+removeXY(int)
+indexOfXby(X(X): int
+indexOfXby(Y(Y): int
+sizeOfData(): int
+setName(String)
1..*
1
CourbeModel<X,Y>
-listcourbeData: ArrayList<Courbe<X,Y>>
-courbeData: Courbe<X,Y>
-scc: Scanner
-ordre: int
-lambda: int
-singleton: CourbeModel<Number,Number>
-Data: double
-dT: double
-BT: double
-CourbeModel()
+getInstance():CourbeModel<Number,Number>
+getNbCourbe():int
+getCourbe(int): Courbe<X,Y>
+setIndex(int)
+getIndexuse(): int
+isSetIndex(): boolean
-checkSettingIndexOrDo(int)
+setCourbes(ArrayList<Courbe<X,Y>>)
+addCourbe(Courbe<X,Y>)
+sizeOfCourbe():int
+getDataX(int): X
+getDataY(int): Y
+moyenneMobile(Courbe=Number,Number>,int)
+saisonResidu(Courbe=Number,Number>,int)
+saison(Courbe=Number,Number>,int)
+desaisonaliser(CourbeNumber,Number>,int)
+logistique(Courbe=X,Number>,int)
+transfoLog(Courbe=X,Number>,int)
+transfoBoxCox(Courbe=X,Number>,int)
+Moyenne(Courbe=X,Number>,int)
+transfoRegLinaire(Courbe=X,Number>,int)
+lissage_regleSE(Courbe=Number,Number>,Courbe=Number,
Number>): int
1
1
1
View
CourbeVue<X,Y>
#model: CourbeModel<X,Y>
#controller: CourbeController<X,Y>
#xAxis: Axis<X>
#yAxis: Axis<Y>
#lineChart: LineChart<X,Y>
#series: XYChart.Series
+CourbeVue(mod:CourbeModel<X,Y>,cont:CourbeController<X,
Y>,xAx:Axis<X>,yAx:Axis<Y>,t:String)
+addSeries(c:CourbeModel<X,Y>,title:String)
+addSeries(c:CourbeModel<X,Y>,title:String)
+setColorSeries(c:CourbeModel<X,Y>,nbCourbe:int,
color:String)
+setDisplay(c:Courbe<X,Y>)
#model(): CourbeModel<X,Y>
CourbeVueConcret<X,Y>
+CourbeVueConcret(mod:CourbeModel<Number,
Number>,cont:CourbeController<Number,
Number>,xAx:Axis<X>,yAx:Axis<Y>,
t:String)
+update(o:Observable,arg:Object)
CourbeVueNumber
+CourbeVueNumber(mod:CourbeModel<Number,
Number>,cont:CourbeController<Number,
Number>,t:String)
+update(o:Observable,arg:Object)
Affiche
SelectFileChooser
+showDialogFileChooser():String
+showDir(Chooser):String
+saveDownloadFileStr(String,file:String)
+error(e:Exception)
InputDialogs
+saisieLambda():double
+saisieOrdre():int
+saisieChoixCourbe(listCourbe:ArrayList<Courbe<Number,
Number>>):ArrayList<Courbe>