

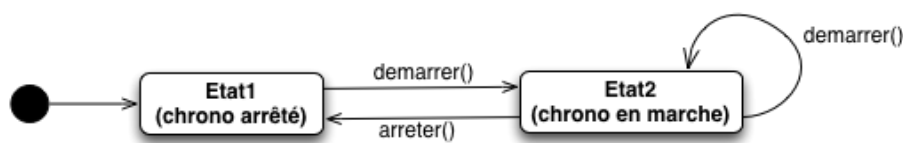
Module : Conception et Programmation Objet Avancées

TP6 Atelier de Génie Logiciel (1h30)

L'objectif de ce TP est de réutiliser la structure du patron de conception « Etat », implémentée de façon générique dans le TP5, et de l'appliquer sur un chronomètre.

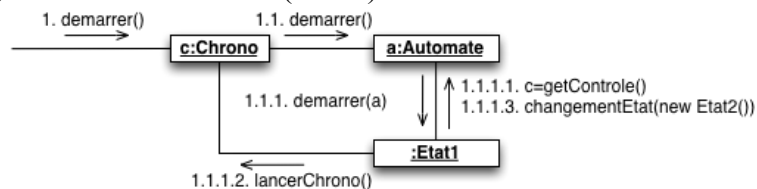
PARTIE A. « A vos marques... »

1. Reprenez le diagramme de classes final du TP5 et renommez les classes et opérations de manière à considérer un chronomètre (classe *Chrono*) proposant deux fonctions de contrôle : *lancerChrono()* et *stopperChrono()*, dont le comportement est régi par un automate à deux états et répondant à deux événements : *demarrer()* et *arreter()*.
2. Générez le code uniquement pour les classes du pattern « state » (les interfaces, *Automate*, *IEtat*, *Etat1*, *Etat2*,...) puis implémentez les opérations de manière à faire adopter à l'automate le comportement suivant :

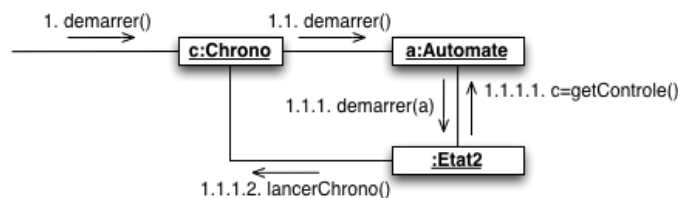


3. Contrôlez le *Chrono* (dont on ignore l'implémentation) en respectant les traitements suivants :

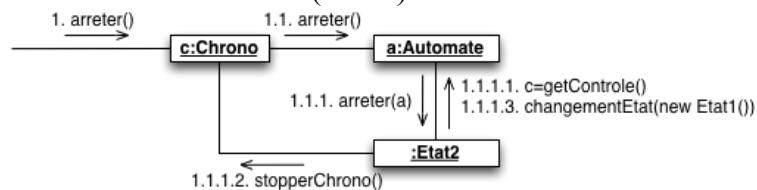
- Appel de *demarrer()* sur le *Chrono* arrêté (Etat 1) :



- Appel de *demarrer()* sur le *Chrono* en marche (Etat 2) :



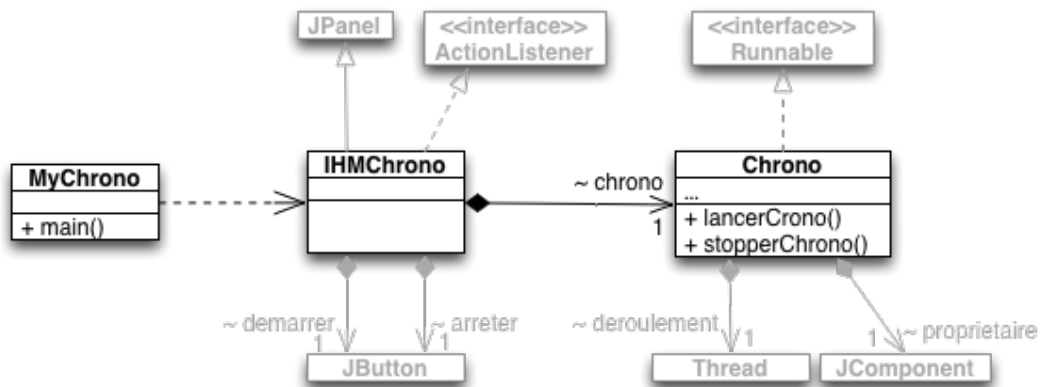
- Appel de *arreter()* sur le *Chrono* en marche (Etat 2) :



Dans les autres situations, il ne doit rien se passer.

PARTIE B. « Prêts... »

1. Récupérez sous Célène l'archive *Chrono1.zip* contenant trois classes conçues ainsi :



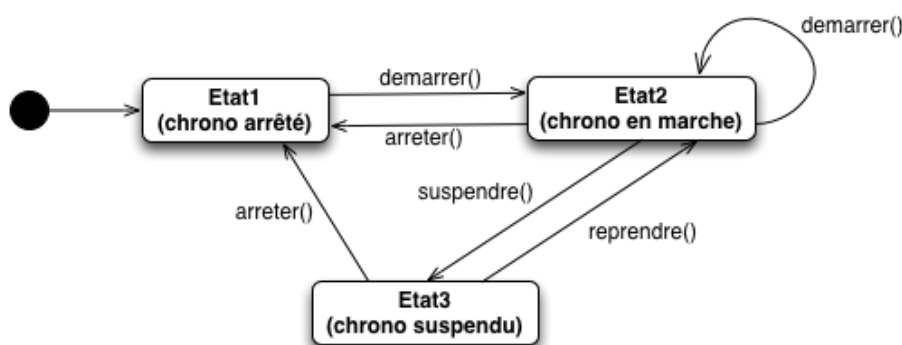
2. « Branchez » la classe *Chrono* sur l'automate :
 - *Chrono* doit réaliser l'interface *IEvenement* (implémentez les opérations)
 - *Chrono* et *Automate* doivent être mutuellement liés
3. Compilez puis testez votre *Chrono* : `$java MyChrono 50`

PARTIE C. « Partez !! »

L'intérêt du pattern « state » et de faciliter l'évolution du fonctionnement d'un objet via une explicitation (dans le code) de l'automate sous-jacent.

Nous souhaitons donc améliorer notre *Chrono* en ajoutant deux nouveaux contrôles : *suspendreChrono()* et *repandreChrono()* permettant respectivement d'arrêter temporairement le chronomètre puis de le redémarrer sans réinitialisation.

En utilisant l'archive *Chrono2.zip*, modifiez votre automate de manière à lui faire adopter le comportement formalisé dans le diagramme ci-après :



PARTIE D. Retour sur Bank

Reprenez la partie C de la feuille de TP2, dans laquelle les règles d'usage d'un compte courant sont explicitées.

1. Formaliser ces règles sous la forme d'un diagramme d'états-transitions
2. Implémentez ce comportement en mettant en œuvre le design pattern « state »

→ Rendez-vous sur Célène pour un QCM de restitution de connaissances (5').