

Satellites et Balises

Alain Plantec

Présentation du scénario

Une entreprise spécialisée dans l'étude des océans organise des campagnes de mesure de caractéristiques physiques comme la température, les courants, la luminosité et la composition du milieu marin. Pour cela, un ensemble de balises autonomes sont larguées dans les zones à étudier.

Une fois lancée, une balise effectue le cycle suivant :

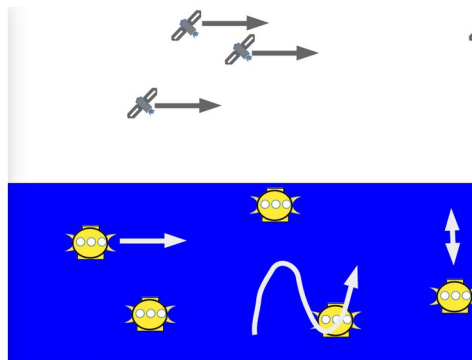
- Etape d'enregistrement : la balise plonge et se déplace dans l'eau ; elle effectue des mesures et les enregistre ;
- Etape de remontée et de synchronisation : la balise remonte en surface et se place en attente de synchronisation avec un satellite ;

Les éléments mobiles

La simulation met en œuvre des éléments mobiles : des balises et des satellites.

Les satellites se déplacent autour de la terre, toujours dans le même sens. Leur vitesse de déplacement est variable et certains peuvent être géostationnaires.

Les balises se déplacent dans le milieu. Le déplacement est variable. Dans un premier temps, sa nature est décidée lors de la programmation de la balise à terre. Par exemple, comme le montre la figure ci-dessous, un déplacement peut être horizontal, vertical ou bien sinusoïdal. La balise peut aussi rester immobile à une certaine profondeur.



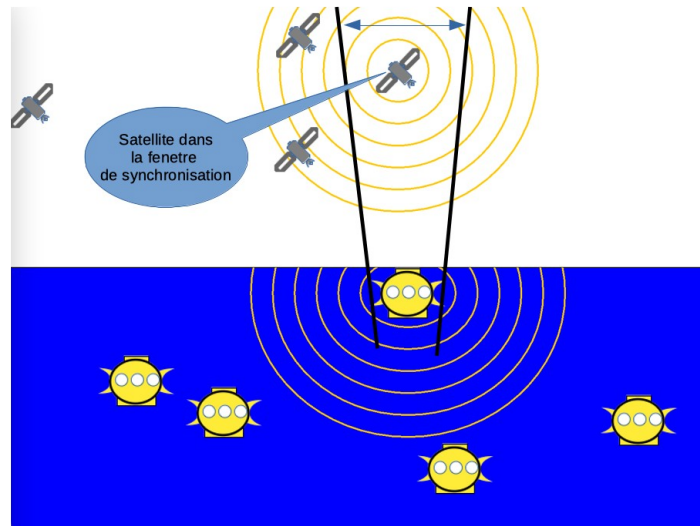
Etape de collecte

Le rôle d'une balise est de collecter des données dans le milieu marin. L'étape de collecte consiste pour la balise à enregistrer des données et les stocker dans sa mémoire. Lorsque sa mémoire est pleine, elle entame une procédure d'émission de ses données vers un satellite pour pouvoir reprendre la collecte.

Etape de synchronisation

Pour vider sa mémoire, une balise remonte à la surface. Elle attend le passage d'un satellite disponible avec lequel elle est en mesure de se synchroniser. Etre en mesure de se synchroniser signifie que le satellite est disponible et qu'il se situe au dessus de la balise comme le montre la figure ci-dessous.

Lorsque la balise et un satellite sont synchronisés, la balise envoie les données au satellite. Quand toutes les données sont transférées, la balise repasse à l'étape de collecte.



Plan de travail

Il est demandé de programmer une simulation avec une visualisation graphique. Au cours d'une simulation, un ensemble de satellites et de balises sont visibles et se caractérisent par des stratégies de déplacement différentes.

La simulation doit mettre en évidence la synchronisation entre une balise et un satellite. Par exemple, par une ligne reliant le satellite à la balise ou par un changement de couleur des vues. Pour cette simulation, les patrons Stratégie et Observable/Observateur doivent être utilisés respectivement :

- la mise en œuvre du déplacement des balises
- la synchronisation entre une balise et un satellite.

Interpréteur de commandes

Objectif

Jusqu'à présent, les éléments de la simulation sont introduits par programme (en dur dans votre code). On voudrait pouvoir interagir avec la simulation à l'aide d'un Shell simple dédié pour :

- ajouter ou supprimer des éléments (satellites, balises)
- changer dynamiquement les propriétés des éléments (vitesse, stratégie de déplacement, ...)
- changer les caractéristiques visuelles (couleur, image utilisée)

Ainsi, au démarrage de la simulation, aucun élément mobile n'est présent. Ils sont ajoutés via la saisie de commandes et leur interprétation.

Voici un exemple de script possible (brouillon) :

```
// Création d'un satellite à la hauteur de 2000 mètres et affectation de sa
// référence dans la variable sat1
sat1 := new Satellite (hauteur = 2000) ;
// sat1 démarre
sat1.start() ;
// Création d'une balise avec un déplacement horizontal
b1 := new Balise(deplacement = #horizontal, profondeur = 200) ;
// b1 démarre
b1.start()
```

Plan de travail

Pour la mise en œuvre de ce langage spécifique, il faut tout d'abord définir une syntaxe abstraite (le

méta-modèle) et mettre à l'épreuve ce méta-modèle en mettant en œuvre des scripts (des modèles). Il est par exemple possible d'utiliser le pattern Commande pour associer un comportement aux entités de ce langage. La mise à l'épreuve de votre méta-modèle s'effectue à l'aide d'exemples et de tests.

Après cette étape, il s'agit d'intégrer un composant permettant la saisie de script par l'utilisateur. Il faut donc définir une syntaxe concrète pour vos script avec Antlr et l'analyseur associé à l'aide d'un visiteur de Antlr.

Documentation

Il est demandé de documenter avec soins le modèle, la vue et comment les patrons Stratégie et Observable/Observateur sont mis en œuvre dans la simulation. UML doit être bien exploité dans votre documentation. Vous ne vous contenterez pas de modèles de classes. Vous vous appuierez aussi sur des diagrammes d'état et de séquence, éventuellement des use-case. Ces différents schémas doivent être expliqués. Aidez-vous de votre propre code pour expliquer : le code des classes et des fonctions java est inclus et commenté avec soins.