

## Travail Pratique 2

## Simulateur pour l'O.T.A.I.

Ce travail pratique vous permettra de mettre en application toutes les notions vues jusqu'à ce jour dans le cadre de votre cours : conception et programmation orientée objet.

### Mise en situation :

L'O.T.A.I (Organisme du Transport Aérien International) est une jeune entreprise qui désire mettre sur pied une compagnie de transport aérien sans réservations.

Cet organisme devra acquérir une flotte de véhicules aériens qu'elle utilisera afin de desservir certains aéroports du monde. Ses clients seraient principalement des passagers et des marchandises qui auraient la possibilité de se présenter à un aéroport desservi par l'organisme, et ce, sans réservation, afin de se diriger, le plus rapidement possible, vers une destination aussi desservie par l'organisme. Des services annexes seront aussi proposés pour des urgences, des incendies et des tours d'observation.

Mais avant de mettre sur pieds une telle entreprise, l'organisme désire faire des simulations afin de faire un choix éclairé sur les aéroports à desservir et le nombre d'avions nécessaires.

### Scénarios :

Aucun scénario ne vous sera fourni durant la phase de développement. Vous devrez les créer vous-même. Il faudra donc développer une application permettant de générer des scénarios.

Générateur de scénarios

Pierre-Elliott-Trudeau (67° 5' N, 95° 5' O) MinPassagers : 1, MaxPassagers : 3, MinMarchandises : 2, MaxMarchandises : 6

Nom : Position : MinPassagers : MaxPassagers : MinMarchandises : MaxMarchandises :

Ajouter

AirBus 350 (Passagers), Capacité : 760, Vitesse : 8, Temps embarquement : 5, débarquement : 1, entretien : 3

Nom : Type : Passagers Vitesse : Embarquement : Débarquement : Entretien :

Ajouter

Image carte : carte.png Image aéroport : aéroport.png Image avion : avion.png

Générer le scénario

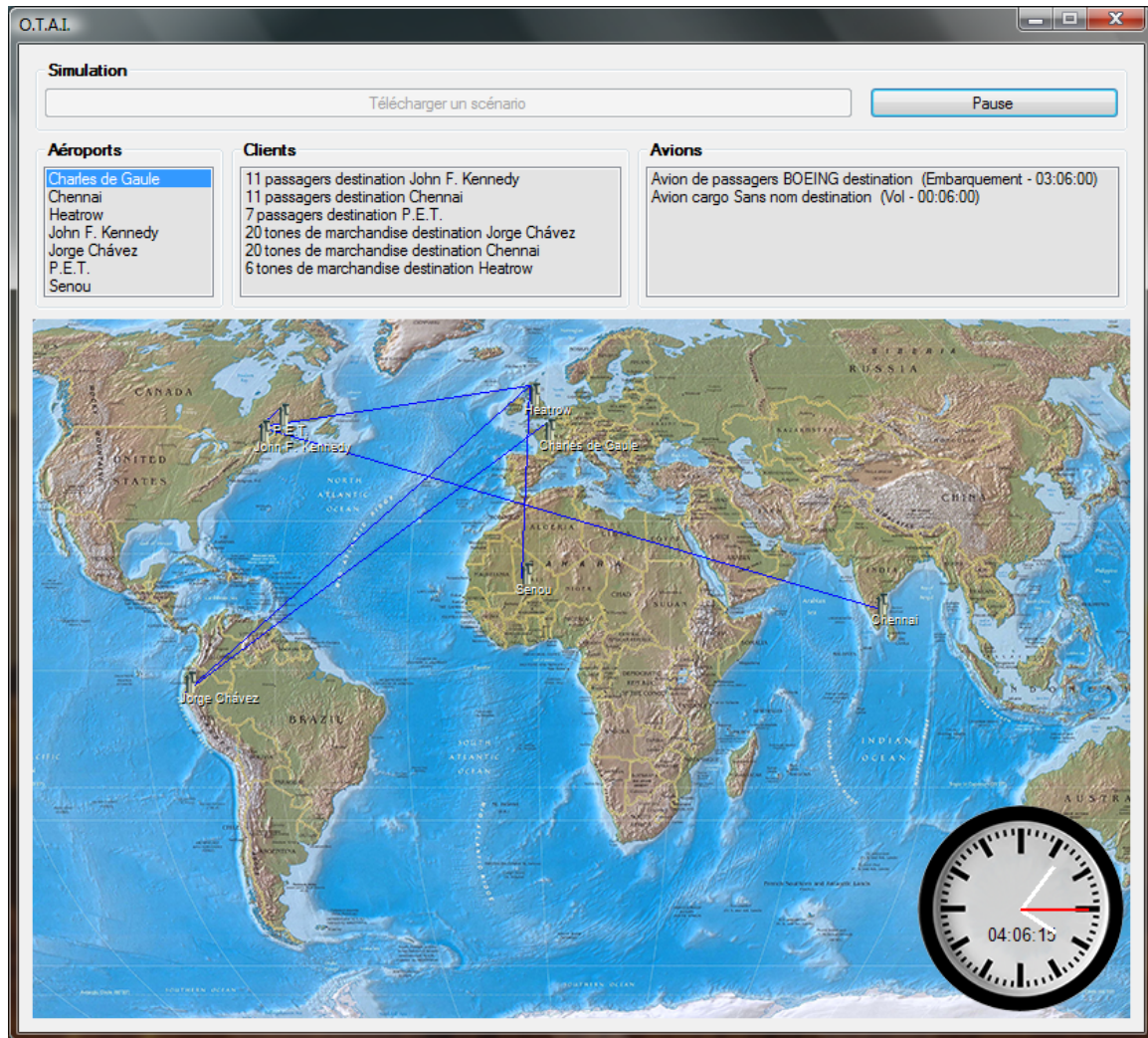
L'application de simulation devra donc se baser sur ses scénarios. Pour se faire, ils devront donc être sauvegardé dans un fichier XML donc la structure ressemblera à l'exemple ci-dessous :

```
<?xml version="1.0" encoding="utf-16"?>
  <ArrayOfCAeroport xmlns:xsi="http://www..." xmlns:xsd="http://www...">
    <CAeroport>
      <Nom>Pierre-Eliott-Trudeau</Nom>
      <Position>67° 5' N, 95° 5' O</Position>
      <Avions>
        <CAvion xsi:type="CAvionPassagers">
          <Nom>AirBus 350</Nom>
          <Capacite>740</Capacite>
          <Vitesse>8</Vitesse>
          <TempsEmbarquement>5</TempsEmbarquement>
          <TempsDebarquement>1</TempsDebarquement>
          <TempsEntretien>4</TempsEntretien>
        </CAvion>
        <CAvion xsi:type="CAvionCiterne">
          <Nom>CL-245</Nom>
          <Vitesse>4</Vitesse>
          <TempsChargement>12</TempsChargement>
          <TempsLargage>5</TempsLargage>
          <TempsEntretien>2</TempsEntretien>
        </CAvion>
        ...
      </Avions>
      <MinPassagersHeure>1</MinPassagersHeure>
      <MaxPassagersHeure>3</MaxPassagersHeure>
      <MinMarchandisesHeure>3</MinMarchandisesHeure>
      <MaxMarchandisesHeure>10</MaxMarchandisesHeure>
    </CAeroport>
    ...
  </ArrayOfCAeroport>
```

### Simulation :

Au départ, le scénario doit être chargé afin d'initialiser, désérialiser les objets relatifs au scénario et démarrer la simulation. L'horloge se mettra alors en marche et toutes les informations doivent être affichées à chaque « tick » d'horloge. Celle-ci sera régie par un « timer » qui augmentera de 15 secondes à chaque « tick ». Pendant le développement, vous pouvez insérer une barre de glissement afin d'ajuster le temps de réaction du « timer ». Il est à noter que l'horloge d'un scénario débute toujours à l'heure 00 : 00 : 00.

Lors de la simulation, la liste des aéroports, triés en ordre alphabétique de nom, devra toujours être présente dans la boîte de liste appropriée. Et lors d'un clic sur un aéroport de cette liste, les listes des clients en attentes à cet aéroport ainsi que les avions reliés à cet aéroport devront être affichées dans les boîtes de liste appropriées selon l'affichage approprié.



Lors de la construction du scénario, le taux d'achalandage des clients pour chacun des aéroports est déterminé. Ainsi, chaque heure, le simulateur devra injecter, de façon aléatoire, un nombre de passagers et de marchandises dans chacun des aéroports selon les paramètres contenus dans le scénario. Sans oublier les appels de secours, entre 1 et 3 par heure, les incendies, entre 1 et 2 par heure, et les observations, 1 par heure.

La destination des passagers et des marchandises sera fixée aléatoirement parmi tous les aéroports sauf celui d'origine. Dans le cas où des passagers ou marchandises sont déjà en attente pour cette destination, leur nombre se verra ajusté pour accepter les nouveaux clients. Le lieu des appels de secours, des incendies et des observations seront aussi fixés, aléatoirement mais peu importe où sur la carte.

**Clientèle :**

L'organisme desservira cinq types de clients : passagers, marchandises, observateurs, secours et incendies. Les informations de ces clients devront être affichées dans la liste de clients dans l'ordre suivant : Secours, Feux, Passagers, Marchandises et Observateurs. Dans le cas où il y a plusieurs destinations de passagers ou de marchandise, ceux-ci seront affichés en ordre décroissant selon le nombre de passagers ou le poids des marchandises.

**Secours :**

Chaque heure, l'organisme reçoit entre un et deux appels de secours. Le lieu peut être n'importe où sur la carte. L'affichage de ce type de clientèle devra être sous la forme suivante : « Secours (69° 15' N, 99° 55' O) ». L'aéroport, possédant un hélicoptère de secours, le plus proche du lieu devra prendre en charge cet appel.

**Feux :**

Chaque heure, l'organisme reçoit entre une et trois déclarations d'incendies nécessitant un avion-citerne. Le lieu de l'incendie peut être n'importe où sur la carte et son envergure entre 1 et 5. L'affichage de ce type de clientèle devra être sous la forme suivante : « Incendie d'envergure 3 (37° 27' N, 49° 12' O) ». L'aéroport, possédant un avion-citerne, le plus proche du lieu devra prendre en charge l'incendie.

**Passagers :**

Chaque heure, des passagers se présentent. L'affichage de ce type de clientèle devra être sous la forme suivante : « 35 passagers à destination de Heatrow ». L'avion possédant une capacité de passagers capables de suffire à la demande devra prendre en charge les passagers. Sinon, il faudra séparer la clientèle en plusieurs groupes.

**Marchandises :**

Chaque heure, des marchandises sont acheminées dans les aéroports. L'affichage de ce type de clientèle devra être sous la forme suivante : « 76 tonnes à destination de P.E.T. ». L'avion possédant une capacité capable de suffire à la demande devra prendre en charge les marchandises. Sinon, il faudra séparer la marchandise en plusieurs groupes.

**Observateurs :**

Chaque heure, l'organisme reçoit un lieu nécessitant observation. L'affichage de ce type de clientèle devra être sous la forme suivante : « Observation (12° 13' N, 55° 54' O) ». L'aéroport, possédant un avion d'observation, le plus proche du lieu devra prendre en charge l'observation.

**Véhicules :**

La flotte de l'organisme comprendra cinq types de véhicules : avion de passager, cargo de marchandises, avion d'observation, hélicoptère de secours et avion-citerne.

**Hélicoptère de secours :**

Les hélicoptères de secours possèdent une vitesse et un lieu de destination.

L'hélicoptère de secours n'a qu'à effectuer un aller-retour entre son aéroport d'origine et le lieu de secours pour compléter le sauvetage. La couleur des trajets des hélicoptères de secours devra être rouge.

**Avion-citerne :**

Les avions-citernes possèdent une vitesse, un temps de chargement qui consiste au temps que prend l'avion pour se remplir d'eau, un temps de largage qui consiste au temps que prend l'avion-citerne pour larguer son eau, un temps d'entretien qui consiste au temps que prendra l'entretien une fois la mission terminée et un lieu de destination où se trouve l'incendie.

L'avion-citerne devra faire l'aller-retour entre un lieu de ravitaillement et le lieu de l'incendie. Une fois le nombre de largages nécessaires effectué, selon l'envergure de l'incendie, l'avion devra atterrir à son aéroport d'origine pour compléter l'extinction. La couleur des trajets des avions-citernes devra être jaune.

**Avion de passagers et de marchandises :**

Ces avions possèdent une capacité, c'est-à-dire le nombre de passagers ou de marchandises maximum qu'elles peuvent transporter, une vitesse, un temps d'embarquement qui consiste au temps que prendront les passagers ou marchandises afin d'être à bord, un temps de débarquement qui consiste au temps que prendront les passagers ou les marchandises pour quitter l'avion, un temps d'entretien qui consiste au temps que prend l'entretien une fois le transport terminé et une destination qui consiste à l'aéroport de destination.

L'avion n'a qu'à se rendre à destination pour compléter le transport. La couleur des trajets des avions de passager devra être verte et celle des avions de marchandises bleue.

**Avion observateur :**

Les avions observateurs possèdent une vitesse et un lieu de destination.

L'avion devra se rendre au lieu d'observation, effectuer un cercle autour du lieu et revenir à l'aéroport d'origine pour compléter l'observation. La couleur des trajets des avions d'observation devra être grise.

**Points techniques :**

Certains points nécessitent quelques précisions. Il est, par contre, à noter que certaines notions ne seront pas expliquées dans le cadre de ce travail pratique. Il vous faudra donc, parfois, fouiller de la documentation afin de trouver une solution.

**Position cartographique :**

Les positions sont représentées selon un système de cartographie utilisé par l'organisme. Chaque position possédera une latitude et une longitude. Celles-ci contiendront un nombre de degré et de minute ainsi qu'une direction. Ex. : 33° 8' N, 46° 56' O

Ainsi, les cartes sont divisées en quatre cadrans : Nord-ouest, Nord-est, Sud-ouest et Sud-est.

Les latitudes possibles sont entre 0 et 90 degrés nord, et 0 et 90 degrés sud. Les longitudes possibles sont entre 0 et 180 degrés ouest, et 0 et 180 degrés est. Le point : « 0° 0' N, 0° 0' O » est donc situé au centre de la carte. Latitude et longitude seront séparées par une virgule. Si une mauvaise position est donnée, celle-ci sera initialisée à : « 0° 0' N, 0° 0' O ».

Votre classe « CPositionCartographique » devra fournir une méthode « Transposer » qui retournera un « PointF » et recevra comme paramètre un « SizeF ». Le paramètre consistera à la taille de votre carte puisque celle-ci peut différer d'un scénario à l'autre. La valeur de retour, quant à elle, consistera au point X et Y sur votre carte.

**Développement orienté objet :**

Le développement devra être rigoureux et respecter le paradigme orienté objet. Il vous faudra donc bien analyser la situation afin de déterminer la hiérarchie de classes nécessaires, les données membres et les associations afin de répondre aux besoins, et ce, en gardant toujours à l'esprit que vos applications doivent être robuste, facilement modifiable et facilement utilisable.

Les collections de données devront hériter de la classe « CollectionBase ». C'est donc dire que vous aurez les classes : CTabAéroports, CTabAvions, et CTabClients.

Les fouilles de ces collections devront redéfinir les méthodes « IndexOf » et « Equals », les tris devront redéfinir la méthode « CompareTo » et utiliser la méthode « Sort ».

La classe « COTAI » sera la classe de base de l'application. C'est elle qui aura la tâche de désérialiser le scénario et fournira une méthode permettant d'obtenir des nombres aléatoires.