

Systemes et Algorithmiques Répartis

Systeme de gestion de bus

Table des matières

SERVEUR DE GESTION DE TRAFIC	2
FONCTIONNEMENT	2
DIAGRAMME DE CLASSE	3
SERVEUR DE BUS	4
COMMUNICATIONS AVEC LE SERVEUR DE GESTION DE TRAFIC	4
INITIALISATION DE LA LIGNE DE BUS	4
RECEPTION DES ORDRES	4
ENVOI DES DONNEES	4
DIAGRAMME DE CLASSE	5
CONTROLEUR	6
CHOIX TECHNIQUES	7
PROTOCOLES DISPONIBLES	7
1. TCP	7
2. UDP	7
3. RMI	7
SYNTHESE CHOIX TECHNIQUES	7
CONFIGURATION PRE-EXECUTION	8
SYSTEME DE GESTION	8
SERVEUR DE LIGNE	8
CONTROLEUR	8
GUIDE UTILISATEUR	9

Serveur de gestion de trafic

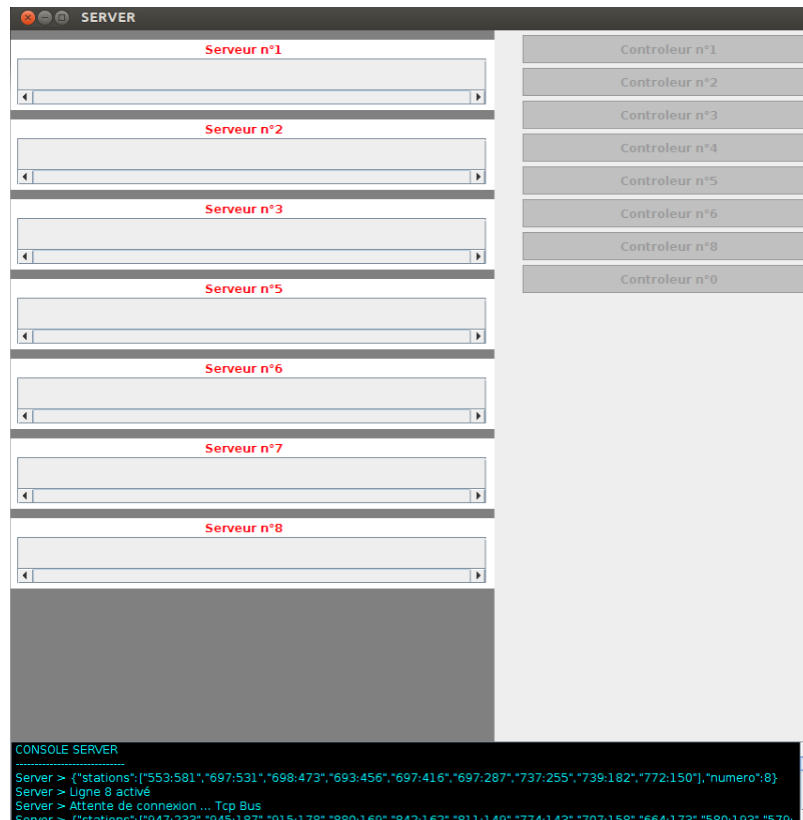


Fig. 1 : interface graphique du SGT

Fonctionnement

Le serveur est la partie centrale du programme. Il doit faire transiter les données entre le contrôleur et le serveur de bus.

Pour ce faire nous avons utilisé un canal TCP pour communiquer avec le serveur. Sur ce canal nous communiquons pour envoyer les ordres du contrôleur au serveur de bus. D'autre part pour faire parvenir les données des Bus nous utilisons un canal UDP.

Le serveur lance deux services pour communiquer en TCP, un pour les serveurs de bus et un pour les contrôleurs pour les échanges d'initialisations et envoi d'ordres. Mais aussi un service pour l'envoi de données en UDP.

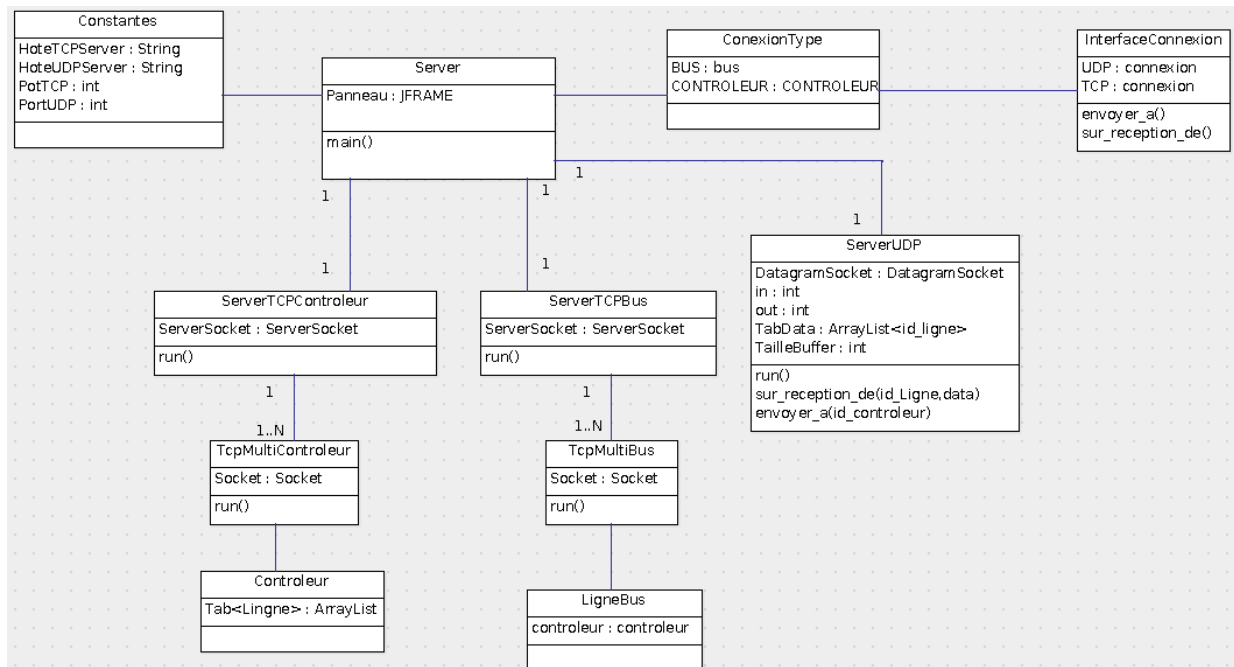
Lors de la connexion d'un serveur de Bus, une socket ligne Bus est créée. Ensuite le serveur lui attribut son numéro de ligne et la position des stations. Le serveur de bus commence à émettre la position de ses bus traités par le service de données une fois cette phase d'initialisation terminée. Quand le serveur reçoit les données d'une ligne de bus, nous utilisons un algorithme, producteur-consommateur avec un Tampon de Taille N décidé à l'avance. Lorsqu'une ligne est attribuée à un contrôleur, la méthode consommer est appelé.

A la connexion du Contrôleur une socket Contrôleur est créée, il reçoit les lignes disponibles. Le contrôleur peut choisir les lignes qu'il veut administrer, si la ligne est déjà administrée par un autre contrôleur il est amené a en choisir une autre.

L'interface graphique du serveur comporte une console qui permet d'afficher en temps réel les différentes requêtes des agents, un panneau qui affiche les lignes qui sont

connectées et les données des bus envoyées et aussi un panneau pour montrer les contrôleurs connectés.

Diagramme de classe



La classe « Constante » : contient les différents attributs de configuration.

La classe « Server » : contient le main du server

La classe « ConnexionType » implémente InterfaceConnexion qui fournit les méthodes « envoyer_a » et « sur_reception_de ».

Les classes « ServeurTCPControleur » et « ServerTCPBus » sont chargées de lancer les services bus et contrôleurs.

Les classes « TcpMultiControleur » et « TcpMultiBus » sont des Threads qui permettent de gérer la connexion Tcp entre les agents.

La classe « ServerUDP » est la classe qui gère la connexion UDP.

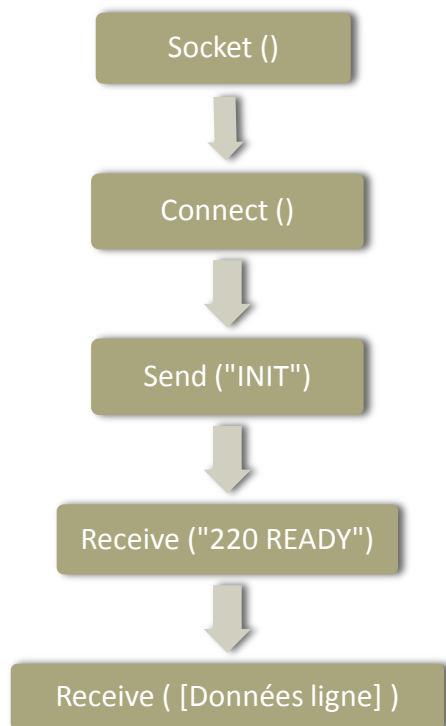
Les classes « Controleur » et « LigneBus » représentent les données métiers de l'application.

Serveur de bus

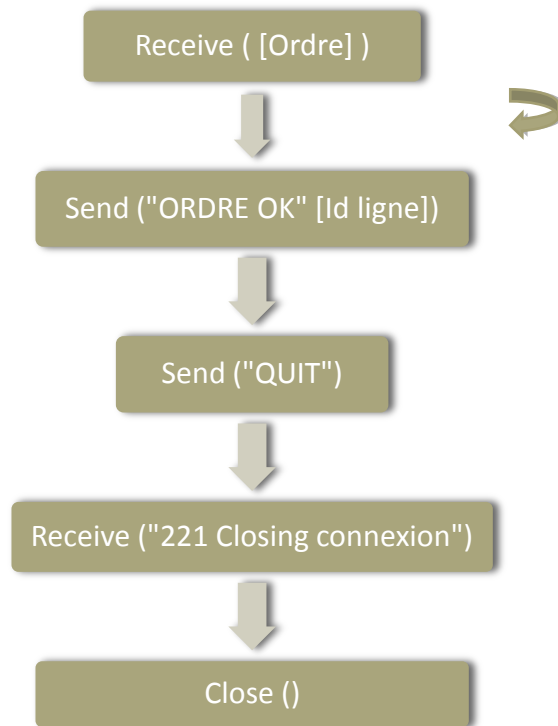
Le serveur de bus (ou serveur de ligne) doit tout d'abord initialisé sa ligne en paramétrant ses stations. Puis il doit créer des bus et les faire démarrer sur la ligne. Ensuite il pourra optionnellement modifier la destination et ou la vitesse d'un ou plusieurs bus en fonction des ordres reçus via le serveur de gestion de trafic.

Communications avec le Serveur de gestion de trafic

Initialisation de la ligne de bus



Réception des ordres



Envoi des données

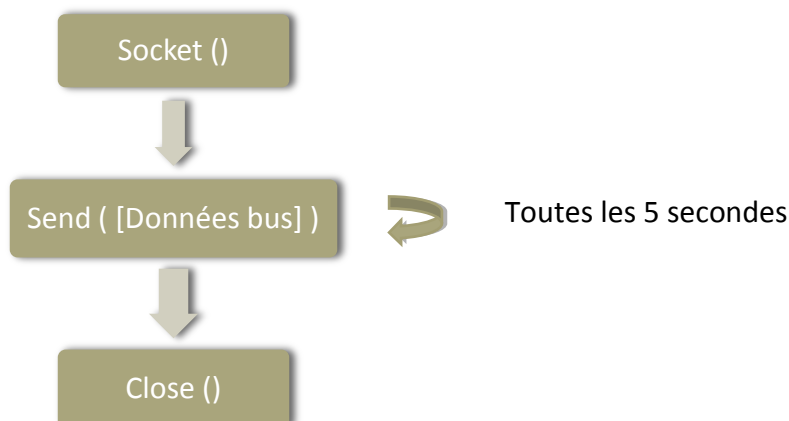
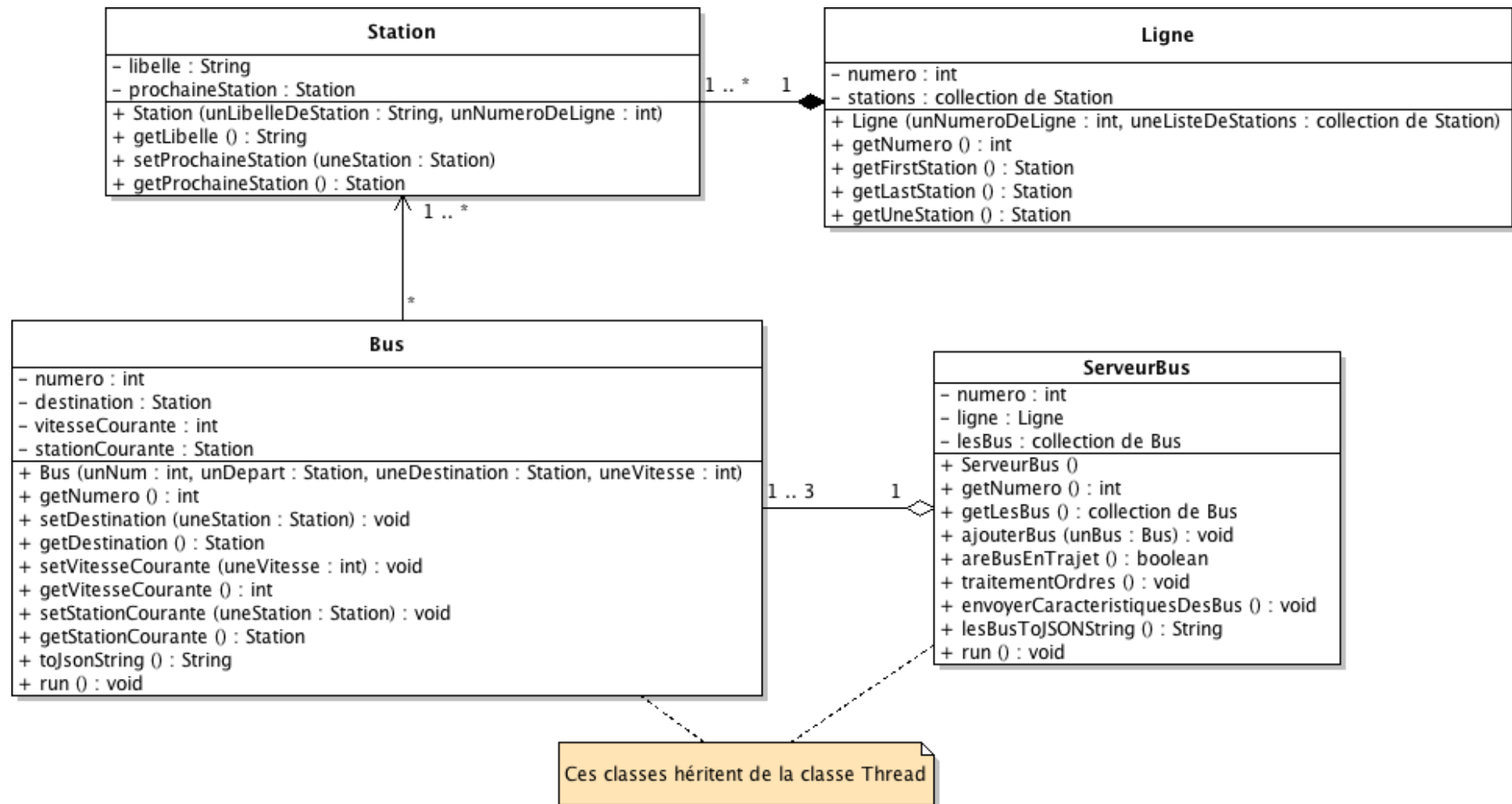


Diagramme de classe



Contrôleur

A la connexion du contrôleur sur le système de gestion, une socket TCP pour l'envoi d'ordre et l'initialisation de la connexion est créée ainsi qu'une socket UDP pour la réception des données des bus. Une fois connectée, le contrôleur a la possibilité de choisir une ou plusieurs lignes parmi celles connectées. Si la ligne est déjà administrée par un autre contrôleur, il peut alors en choisir une autre.

Le contrôleur possède le même diagramme de classe que le serveur de bus.

L'interface du contrôleur affiche en temps réel la position des bus sur une carte ainsi que leurs caractéristiques. Les caractéristiques d'un bus peuvent être modifiées en cliquant sur le bus concerné puis en modifiant dans le menu de droite les options désirées. Pour mettre à jour les informations du bus il est nécessaire de cliquer sur l'icône du bus sur la carte. Cependant les informations arrivent toutes les cinq secondes, une petite latence est à prévoir.

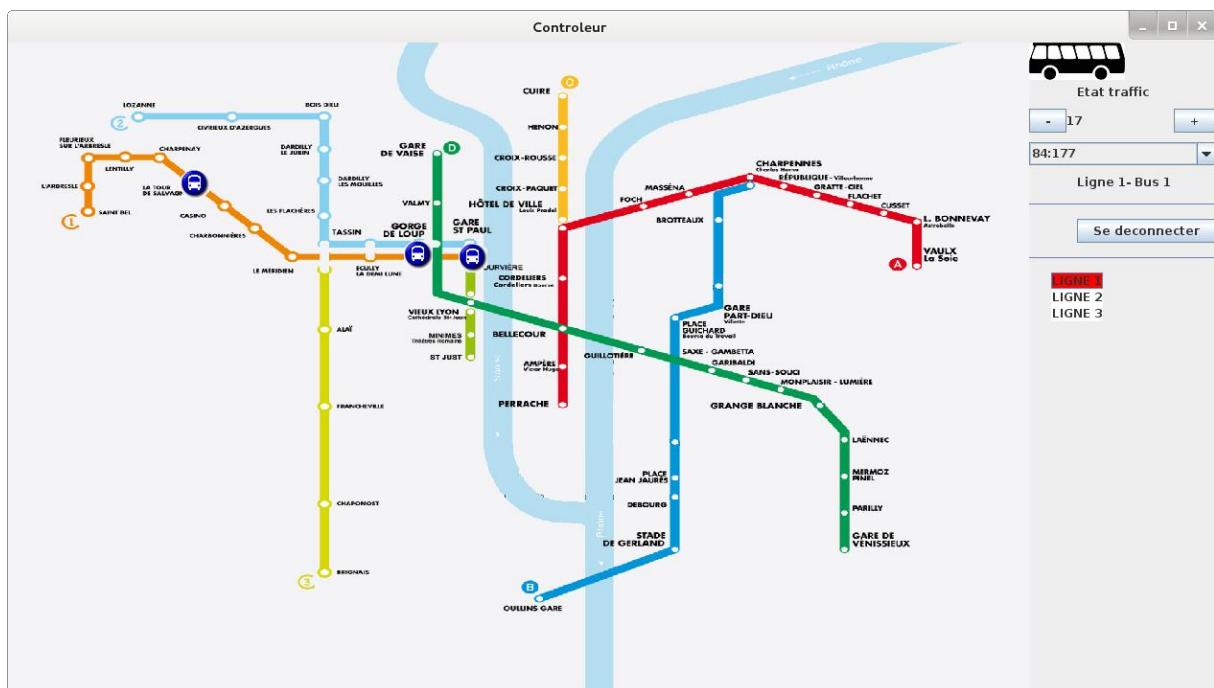


Fig. 2 : interface graphique du contrôleur

Choix techniques

Protocoles disponibles

1. TCP

Nous avons choisi d'offrir un service TCP aux bus et aux contrôleurs pour les envois d'ordres, l'initialisation et les déconnexions. Toutes les phases d'administration se passe en TCP.

Nous avons utilisé TCP parce que l'action est bloquante, TCP nous permet de pouvoir gérer les accusés réceptions et nous garantir un échange synchrone entre les agents. Il existe un petit temps de latence entre dans la réception des ordres que nous avons observé aux cours des tests réalisés.

2. UDP

Pour l'envoi de données, nous avons choisi d'implémenté ce service en UDP pour que l'action ne sois pas bloquante. Et pour pouvoir gérer toutes les données nous avons implémenté un contrôle de flux afin de rendre la connexion synchrone. UDP a répondu parfaitement aux besoins surtout pour afficher les données des bus en temps réel.

3. RMI

Nous n'avons pas retenu cette solution. RMI est très facile à implémenter mais on n'a pas le contrôle sur le nombre de messages échangés. La connexion se voit beaucoup plus lente en RMI.

Synthèse choix techniques

La solution que nous retenons mélange du TCP et de l'UDP pour obtenir la meilleure optimisation. Pour le dialogue entre le contrôleur et les serveurs de bus, le nombre de messages échangés est aléatoire selon le contrôleur et nous voulons avoir la garantit que le message sois reçu. C'est pour cela que TCP nous paraît le plus adéquat pour cette liaison. Pour l'envoi de données des bus, un canal UDP dans notre cas était ce qu'il y a de mieux, car il n'y a pas besoin de la notion d'acquittement.

Configuration pré-exécution

Voici les différents paramètres à modifier afin de lancer les applications.

Système de gestion

Dans le package utilitaire, la classe Constantes.java :

- IP : adresse IP de la machine où se trouvent tous les contrôleurs
- TCPSystem : port de communication TCP avec le contrôleur
- PortUDPSystem : port UDP utilisé avec le contrôleur et le bus
- portTcpBus : port de communication TCP avec le bus

Serveur de ligne

Dans le package dauphine.mi.projetsar, la classe ServeurBus.java :

- IP_DISTANTE : adresse IP du système de gestion
- PORT_ORDRES_DISTANT : port TCP du système de gestion dédié au bus
- PORT_CARACTERISTIQUES_DISTANT : port UDP du système de gestion

Contrôleur

Dans le package fr.dauphine.sar.reseau, la classe Constantes.java :

- IPDistant : adresse ip du système de gestion
- portDistantTCP : port TCP du système de gestion
- portDistantUDP : port UDP du système de gestion

Guide utilisateur

Le serveur de gestion du trafic doit être lancé en premier. Une fois lancée la console de Debug intégrée à l'interface retrace les événements réseaux de l'application.

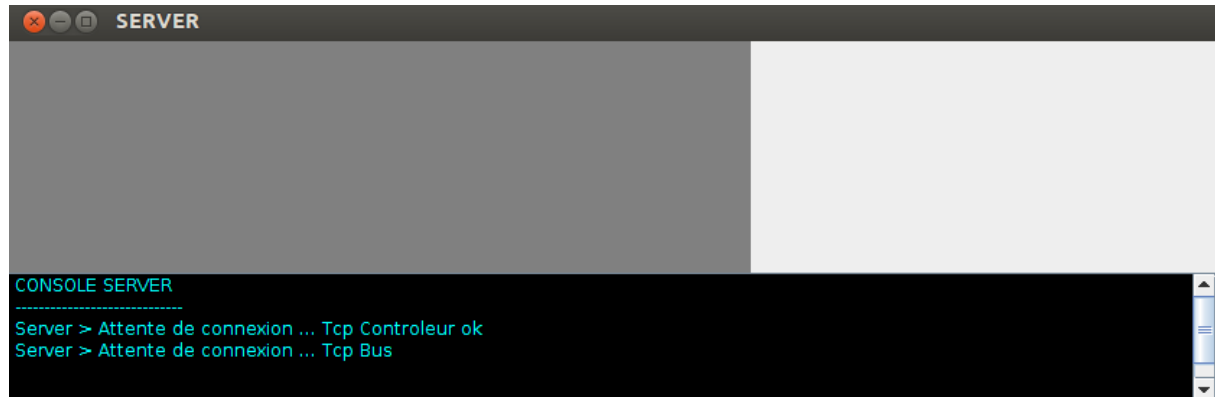
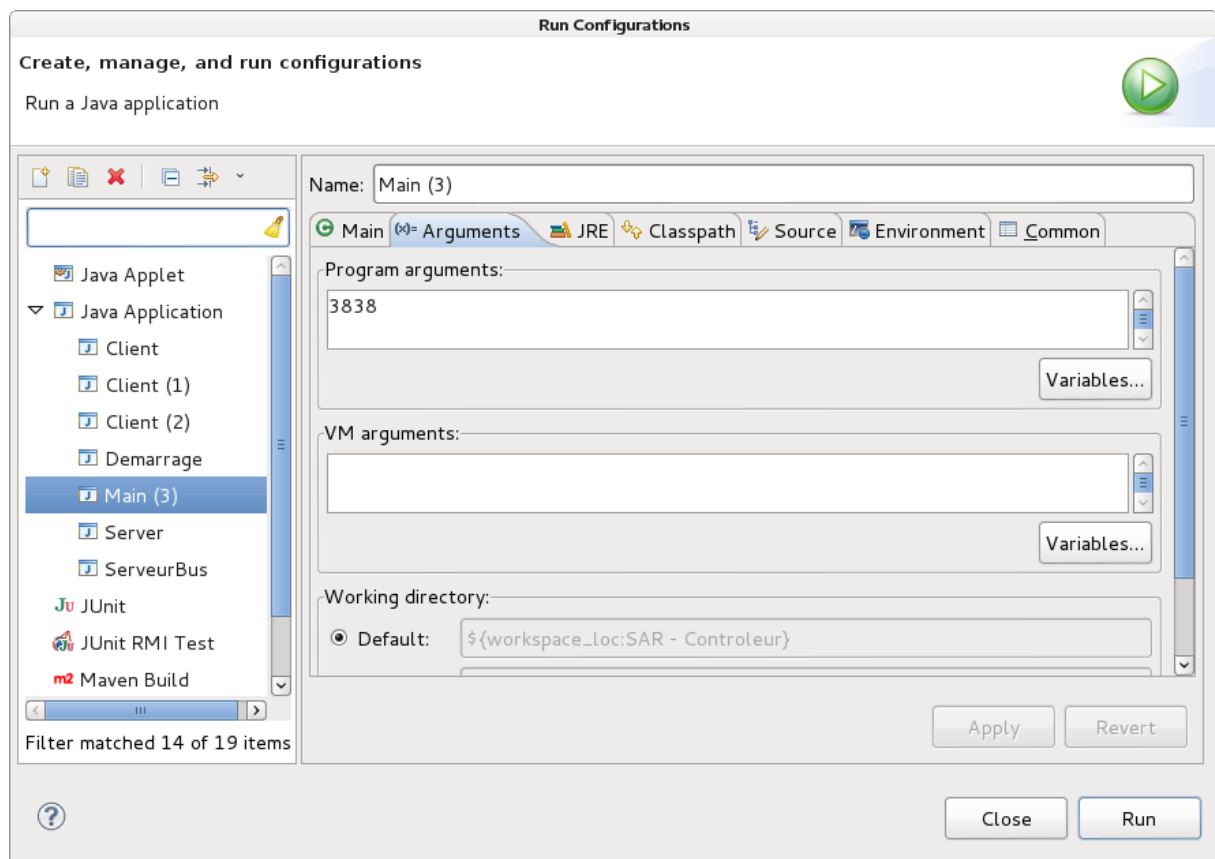


Fig. 3 : interface de Debug du SGT

Ensuite, il est nécessaire d'exécuter le(s) serveur(s) de ligne(s). Le nombre maximum de lignes acceptées est de huit.

Le contrôleur doit être lancé avec un argument : le numéro de port. Sans cette information le message « Il manque des arguments » sera affiché dans la console.



Une fois les différents agents lancés, le contrôleur peut se connecter au système de gestion. Le contrôleur peut être lancé et ne pas être connecté au serveur. Afin de récupérer la liste des lignes disponibles il faut appuyer sur le bouton « Se Connecter ».

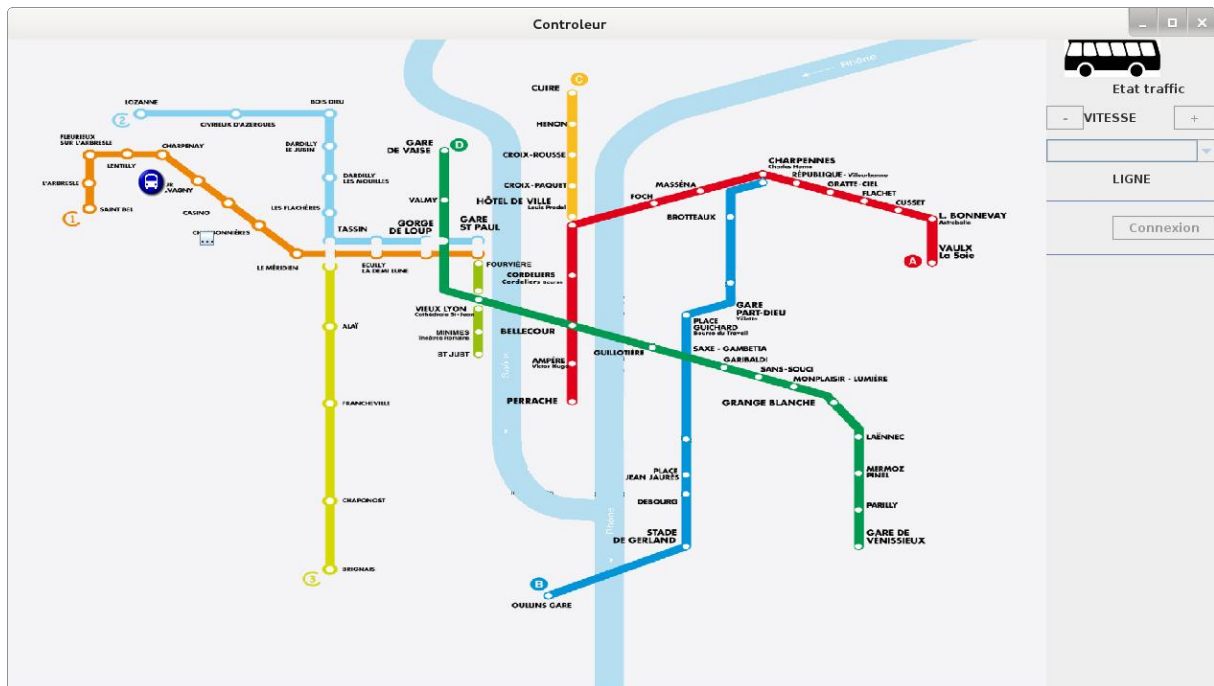


Fig. 4 : contrôleur à la connexion

Une fois la liste récupérée, double cliquer sur une ligne pour qu'elle vous soit attribuée. S'il est indisponible un message le précisant sera afficher dans la console système.

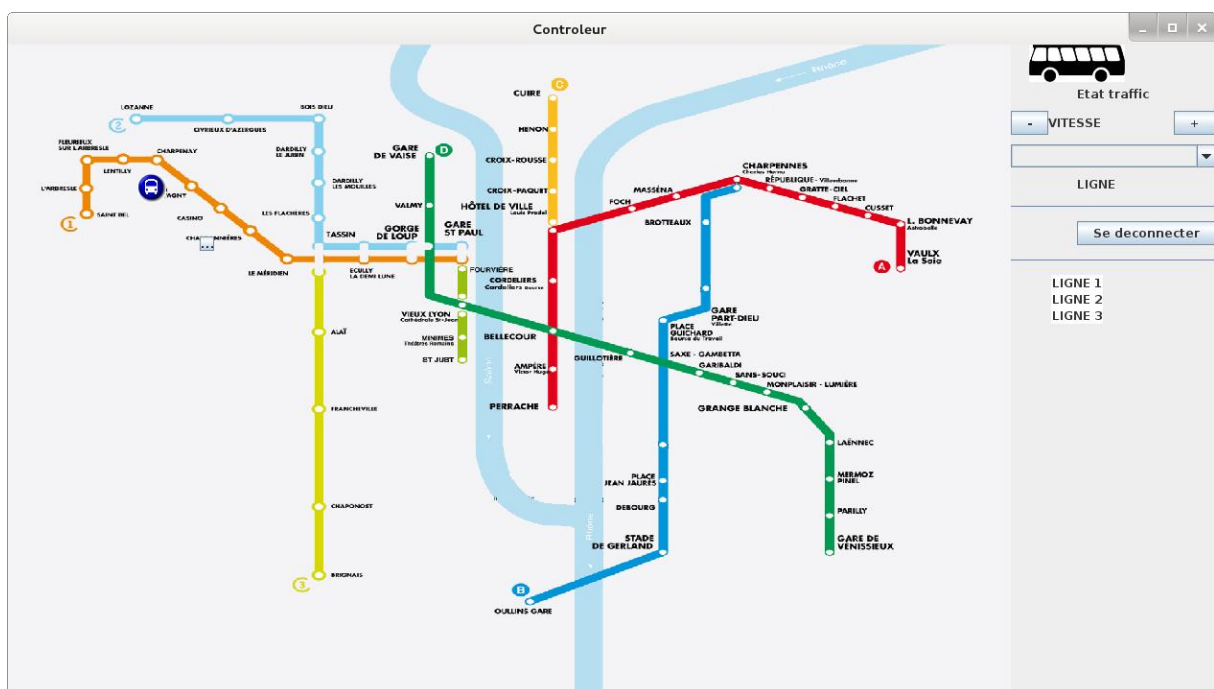


Fig. 5 : contrôleur avec lignes mais panneau de droite vide

Une fois les bus apparurent sur la carte, il suffit de cliquer sur l'un d'entre eux pour mettre à jour le menu de droite contenant les informations.

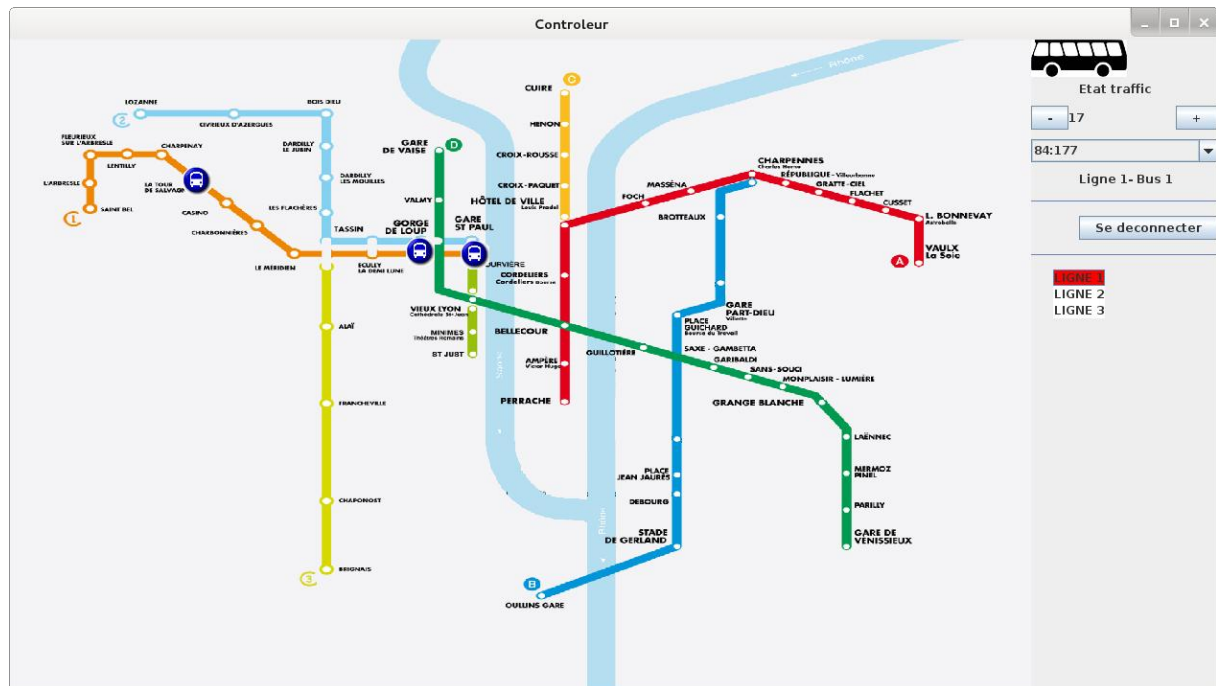


Fig. 6 : contrôleur avec lignes mais panneau de droite plein

Les deux boutons bordant l'affichage de la vitesse permettent d'augmenter et réduire la vitesse d'un bus. La liste déroulante permet de modifier la destination finale d'un bus. Un simple clique est nécessaire pour l'envoi de l'ordre au serveur de ligne. Une fois la commande envoyée il suffit de recliquer sur le bus concerné pour que ses informations soient mise à jour.

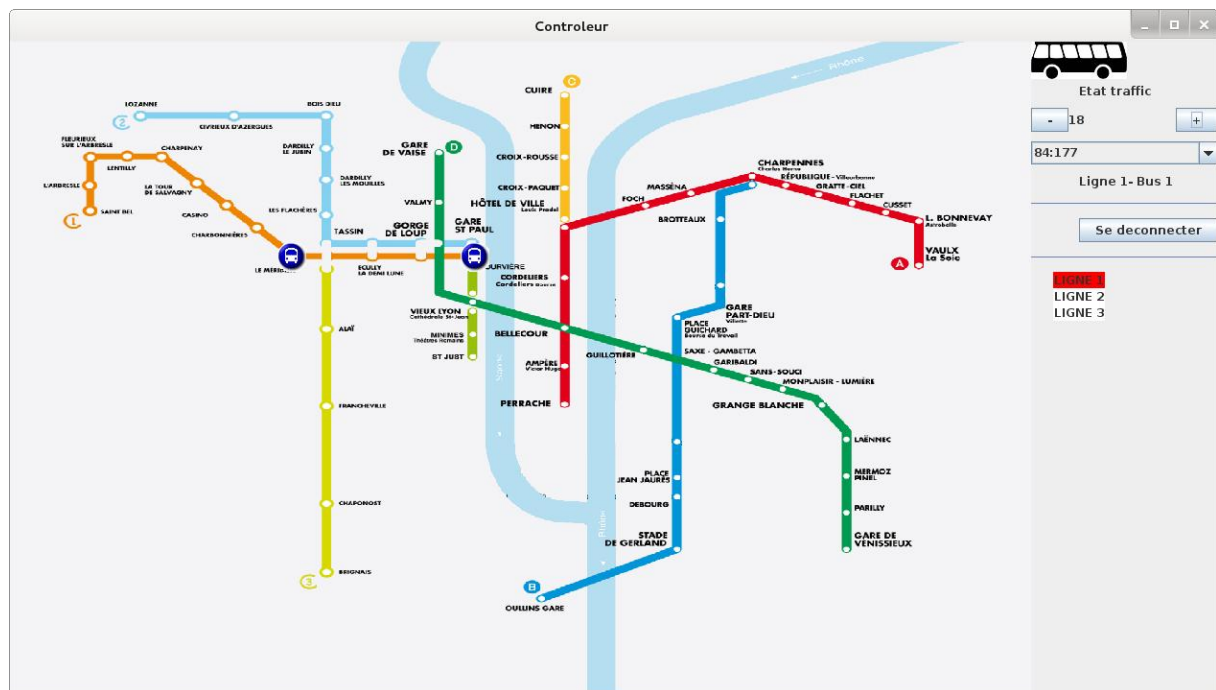


Fig. 6 : contrôleur avec lignes mais panneau de droite plein et information modifiée