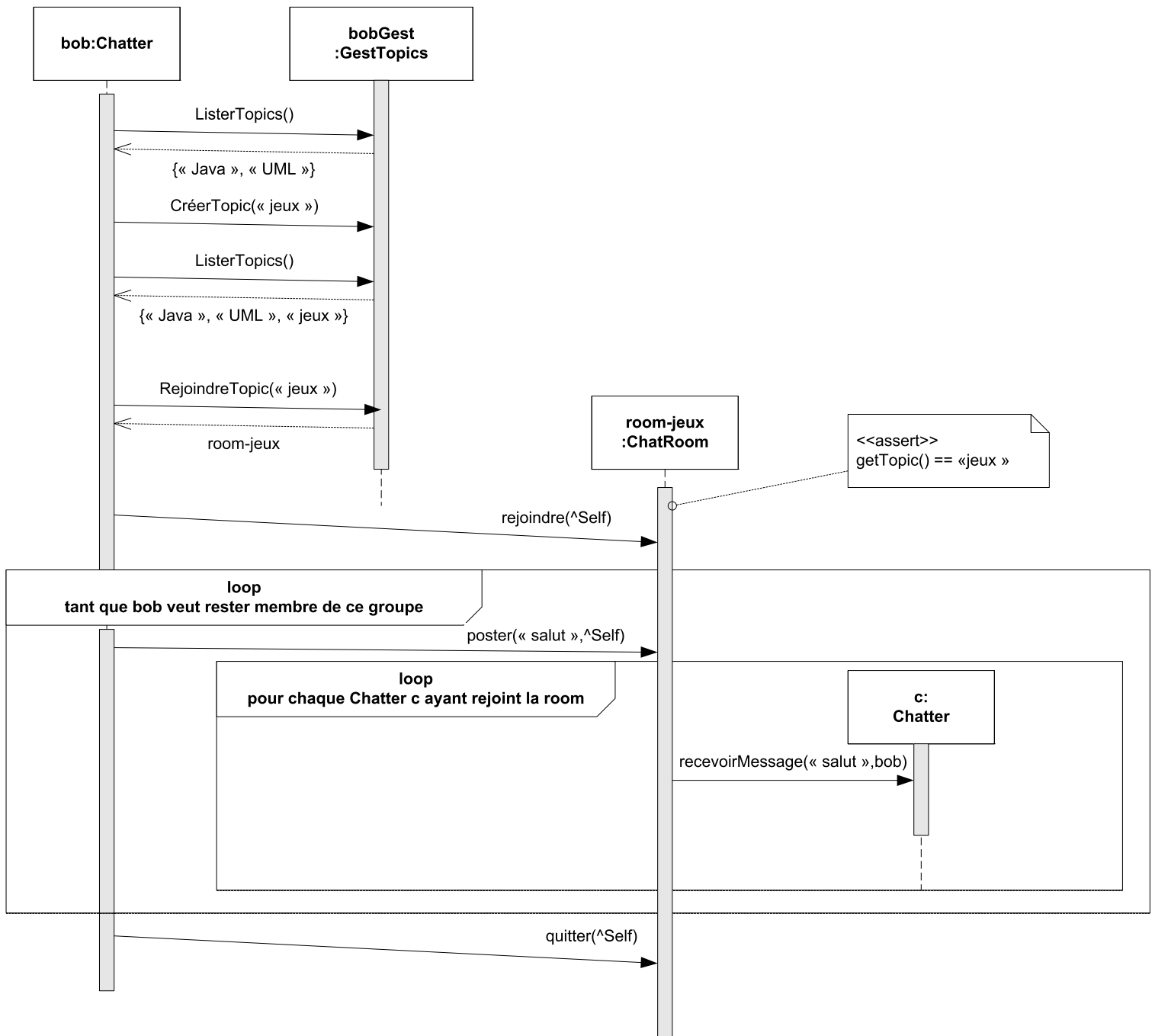
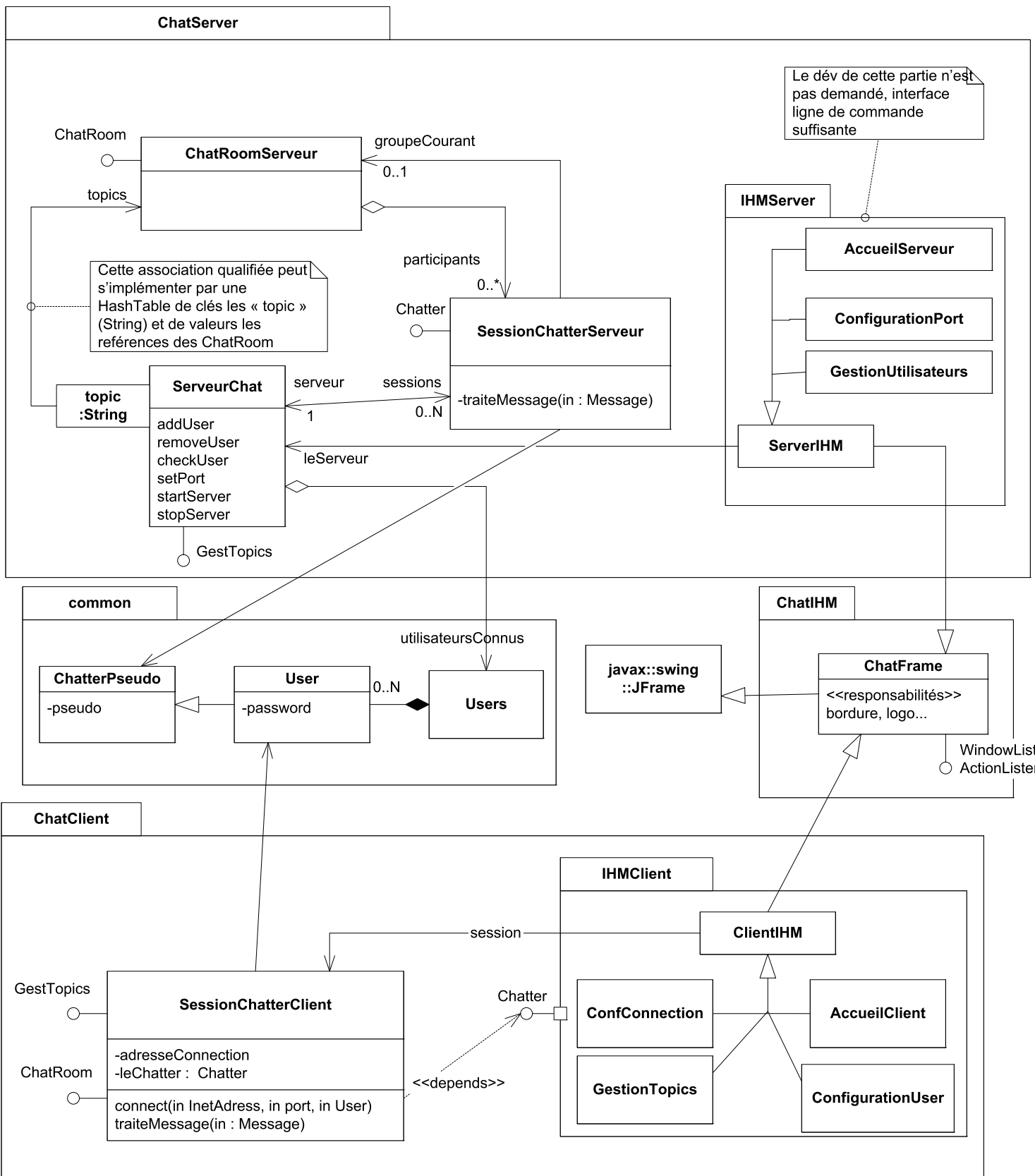


Interfaces du package Chat



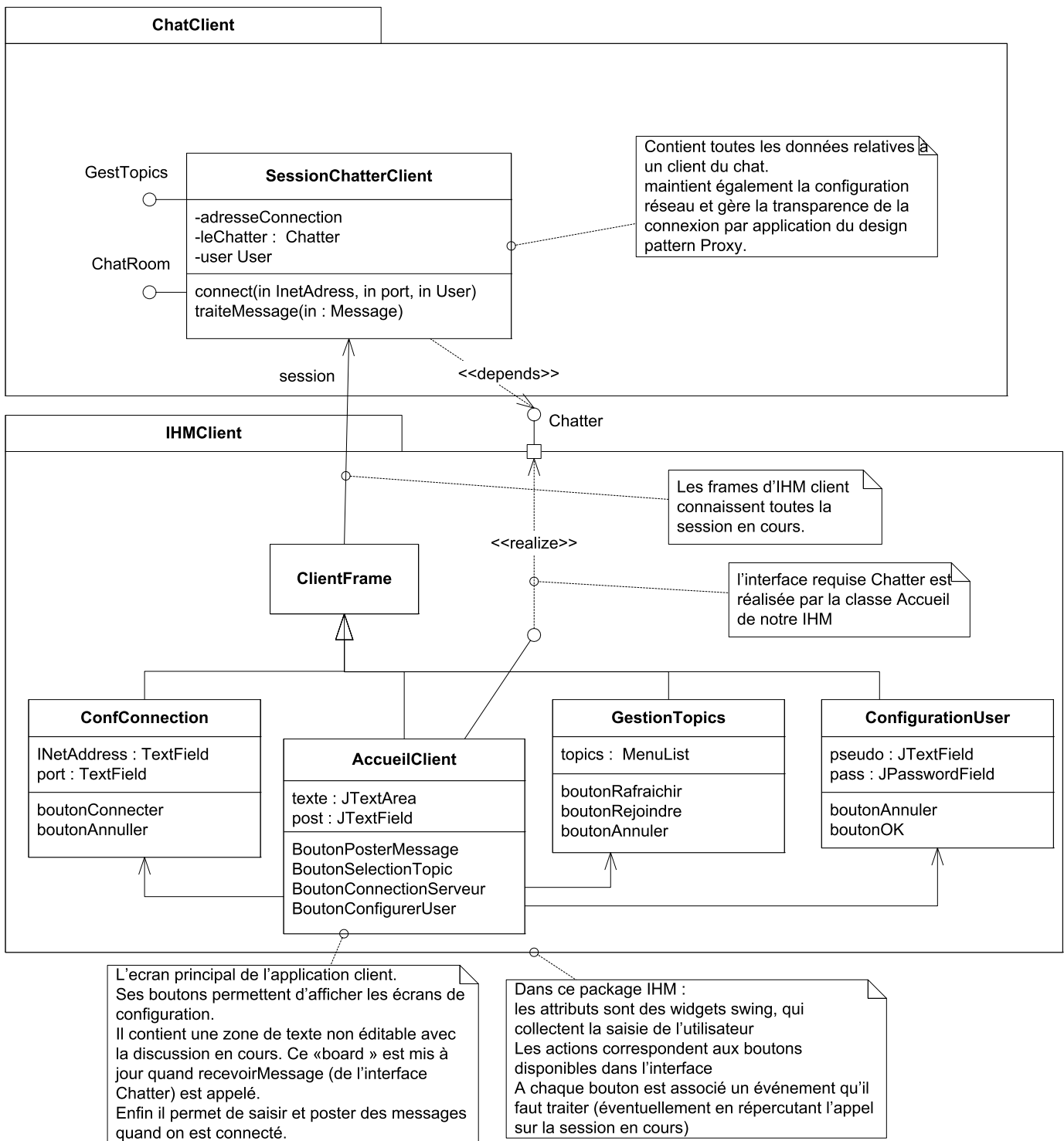
Interaction nominales des interfaces du package Chat



Une première ébauche de l'infrastructure finale attendue

Ce diagramme a servi en phase de préparation : il donne une vue globale des applications client et serveur

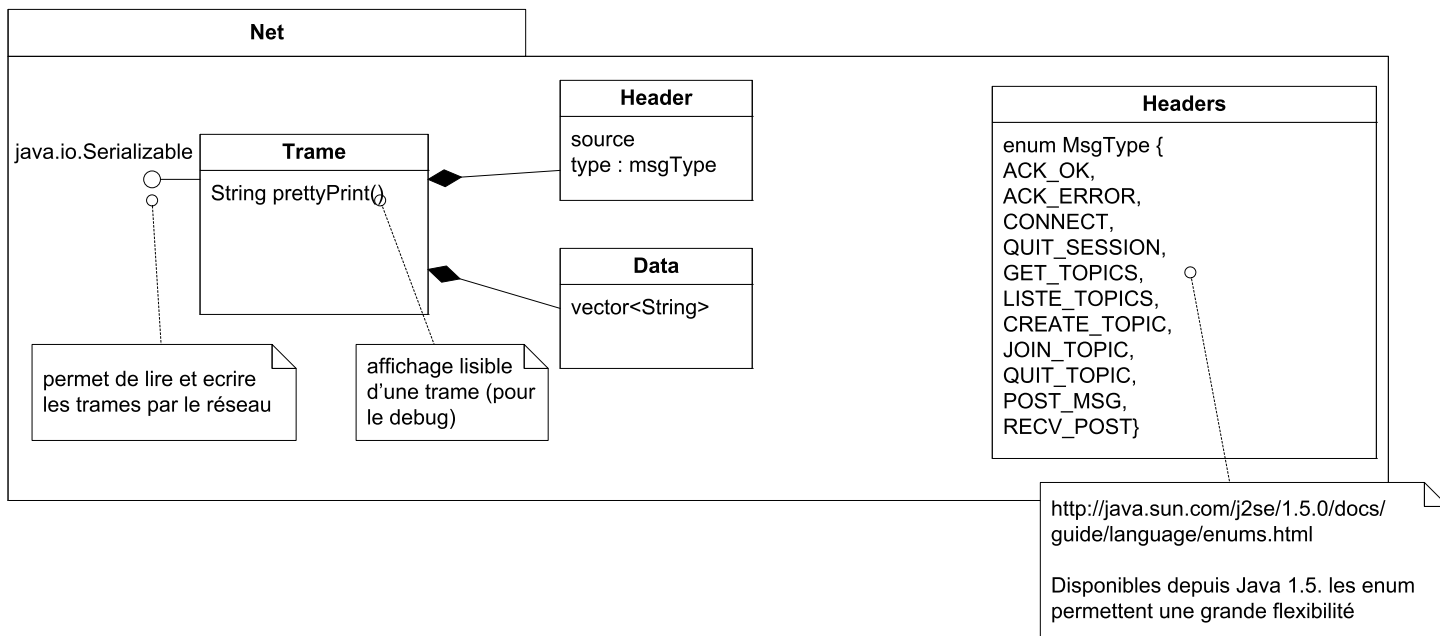
Certains éléments de ce diagramme ont été modifiés dans la conception finalement retenue



Une première ébauche de l'infrastructure détaillée du client

Ce diagramme a servi en phase de préparation : il donne une vue détaillée du client

La modification majeure de ce diagramme en version finale est la séparation de la session chatter client en deux entités (une chatroom et un gestionnaire de topics)



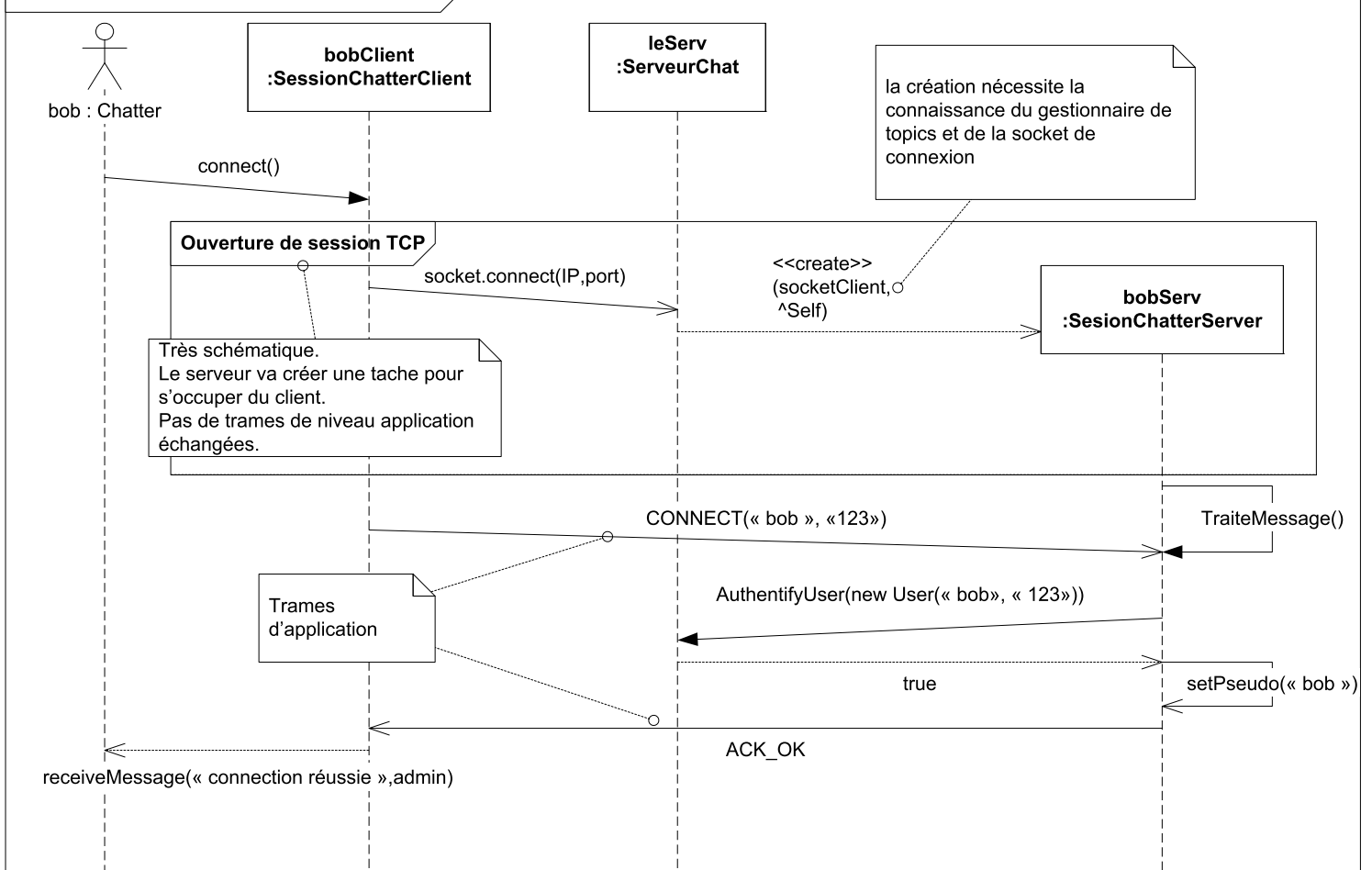
Type du message	Data associé	Utilisation	Direction
ACK_OK,	aucune	Acquittement nominal, sans données. Tout message doit être acquitté	Cli->Serv Serv->Cli
ACK_ERROR,	String: raison de l'erreur	Acquittement erreur : la demande n'a pas pu être satisfaite.	Cli->Serv Serv->Cli
CONNECT,	String pseudo, String Pass	Demande d'authentification de l'utilisateur client.	Cli->Serv
QUIT_SESSION,	aucune	pour initier une deconnexion du serveur.	Cli->Serv Serv->Cli
GET_TOPICS,	aucune	demande de la liste de topics ouverts	Cli->Serv
LISTE_TOPICS,	vector<string> : les topics	réponse à la trame précédente	Serv->Cli
CREATE_TOPIC,	String : titre du topic	demande à créer un nouveau topic	Cli->Serv
JOIN_TOPIC,	String : titre du topic	demande à rejoindre un groupe de discussion	Cli->Serv
QUIT_TOPIC,	aucune	demande à quitter le groupe de discussion auquel le chatter est connecté.	Cli->Serv

Ebauche de la partie liées au protocole applicatif mis en place

Une étude des trames à utiliser dans l'application

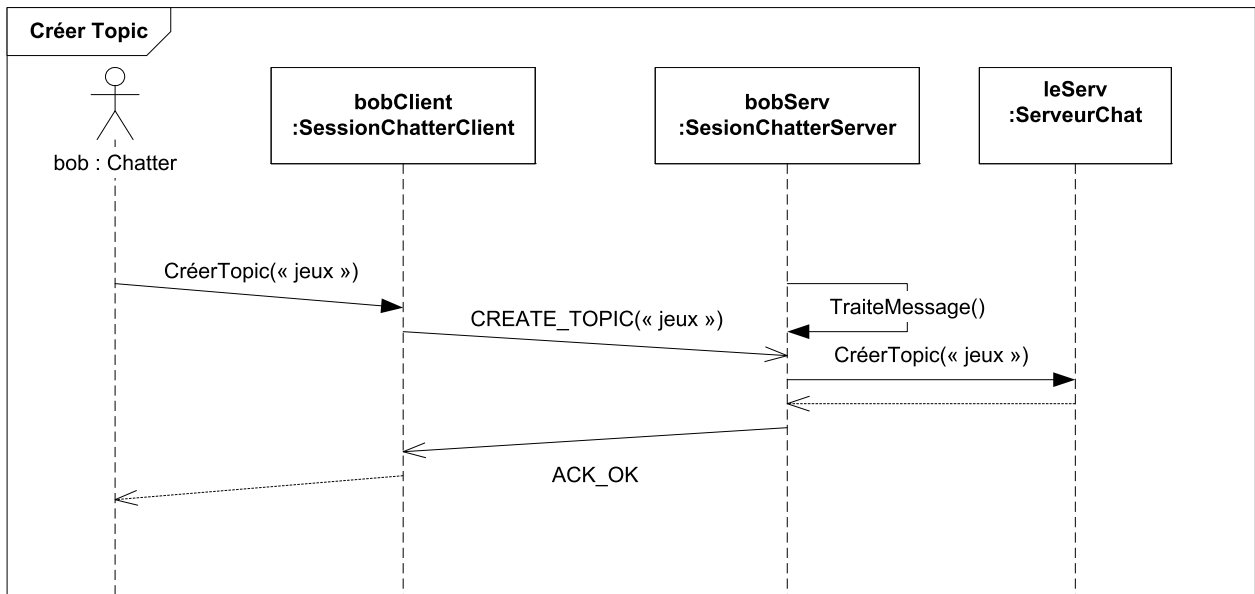
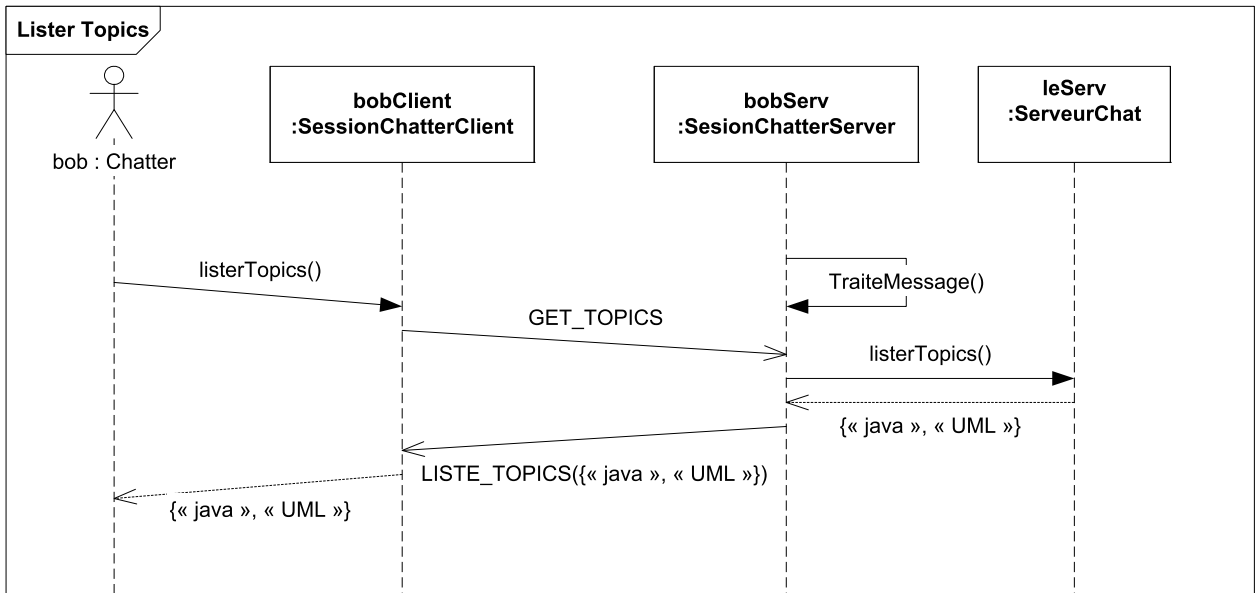
Ces trames pourraient etres complétées : elles sont découvertes à l'aide des diagrammes de séquence qui vont suivre

Connexion à l'ouverture d'une session de chat

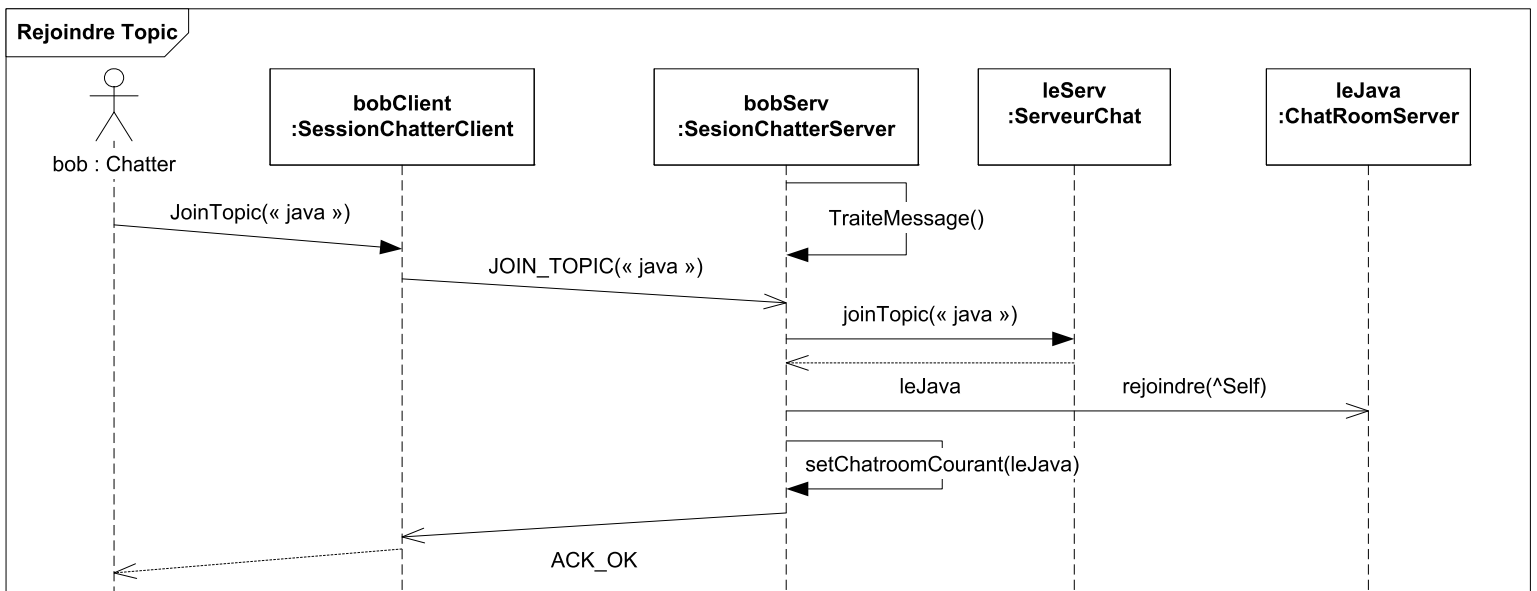


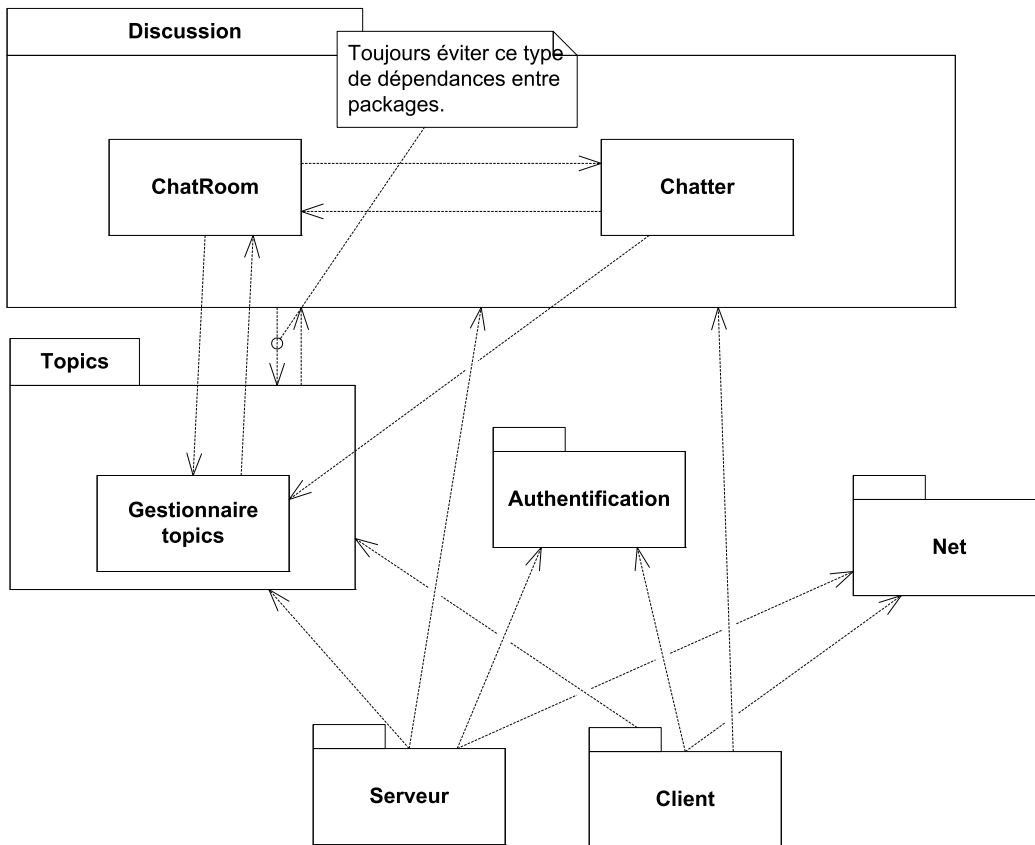
La partie connexion et authentification de l'utilisateur

L'authentification n'a pas été implémentée dans le proto fourni en corrigé.

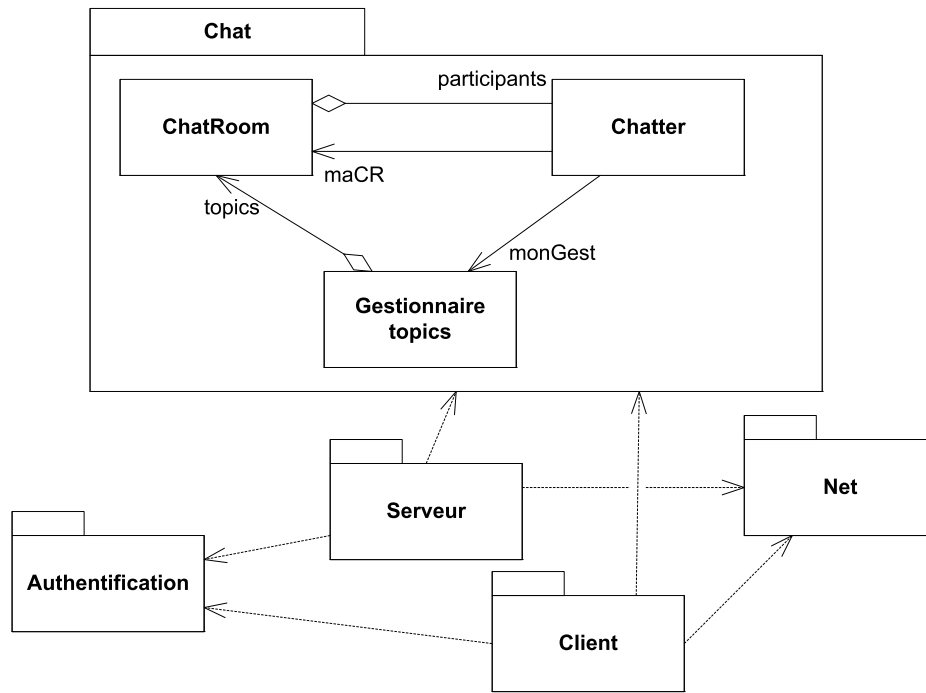


Trames de protocole liées à la gestion des topics.
Il manque détruire topic ?



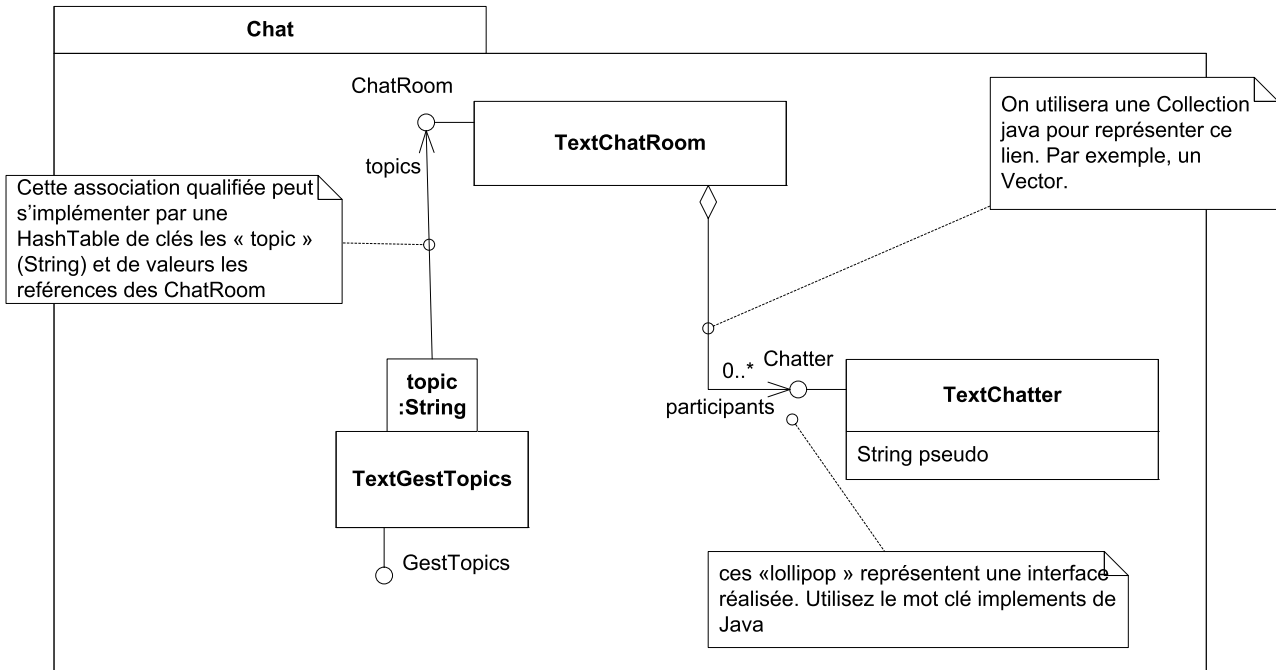


On a a présent une bonne idée de la solution qui va être mise en place
 Détaillons et affinons le découpage en package
 un premier découpage grossier

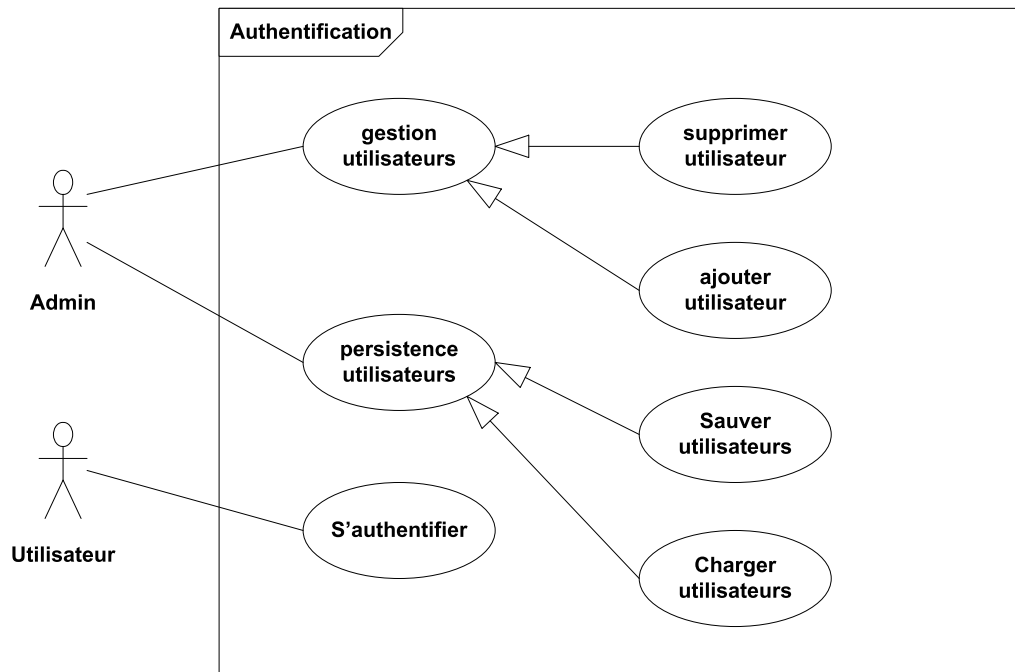


Le découpage en package retenu

L'IHM n'apparait pas : elle ne dépend que du client et du chat (elle implémente l'interface Chatter)



Une implém basique du package Chat



Les cas d'utilisation du package Authentication

Authentification

<<interface>> GestAuthentification

```
void addUser(String login, String pass) throws UserExists;  
void removeUser(String login) throws UserUnknown;  
  
void authentify (String login, String Pass) throws UserUnknown,WrongPassword;  
  
static GestAuthentification charger (String path) throws IOException;  
void sauver (String path) throws IOException;
```

Ajout d'un nouvel utilisateur.
Lève une exception si un utilisateur portant ce login existe déjà.

Supprime un utilisateur.
Lève une exception si l'utilisateur portant ce login n'existe pas.

Demande à authentifier un utilisateur.
Lève une exception si l'utilisateur est inconnu ou si le password fourni est erroné.

Crée un gestionnaire d'authentification en chargeant les données depuis un fichier dont le chemin est passé en argument.
Notons que cette définition est static : c'est une opération de classe et non d'instance .
Peut lever une exception d'entrée sortie s'il y a un problème à la lecture du fichier.

Permet de sauver les données des utilisateurs dans un fichier.
Cette opération d'instance peut lever des exceptions d'entrée-sortie si l'écriture dans le fichier échoue.

java.lang.Exception

AuthenticationException

+String login

Le nom d'utilisateur qui est en faute.

UserUnknown

WrongPassword

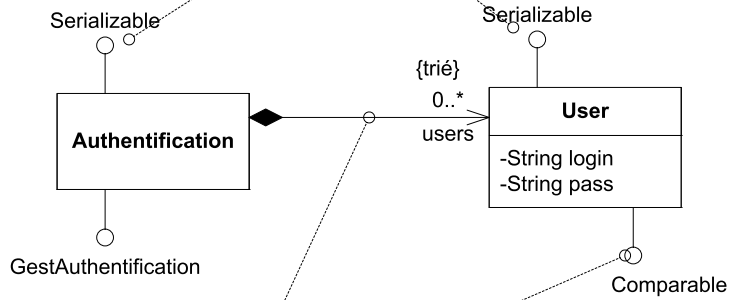
UserExists

Un package gérant l'authentification

Notons que les password devraient en principe être gérés par un char[] : la String java est persistante, c'est un trou de sécurité de passer les password dans des string

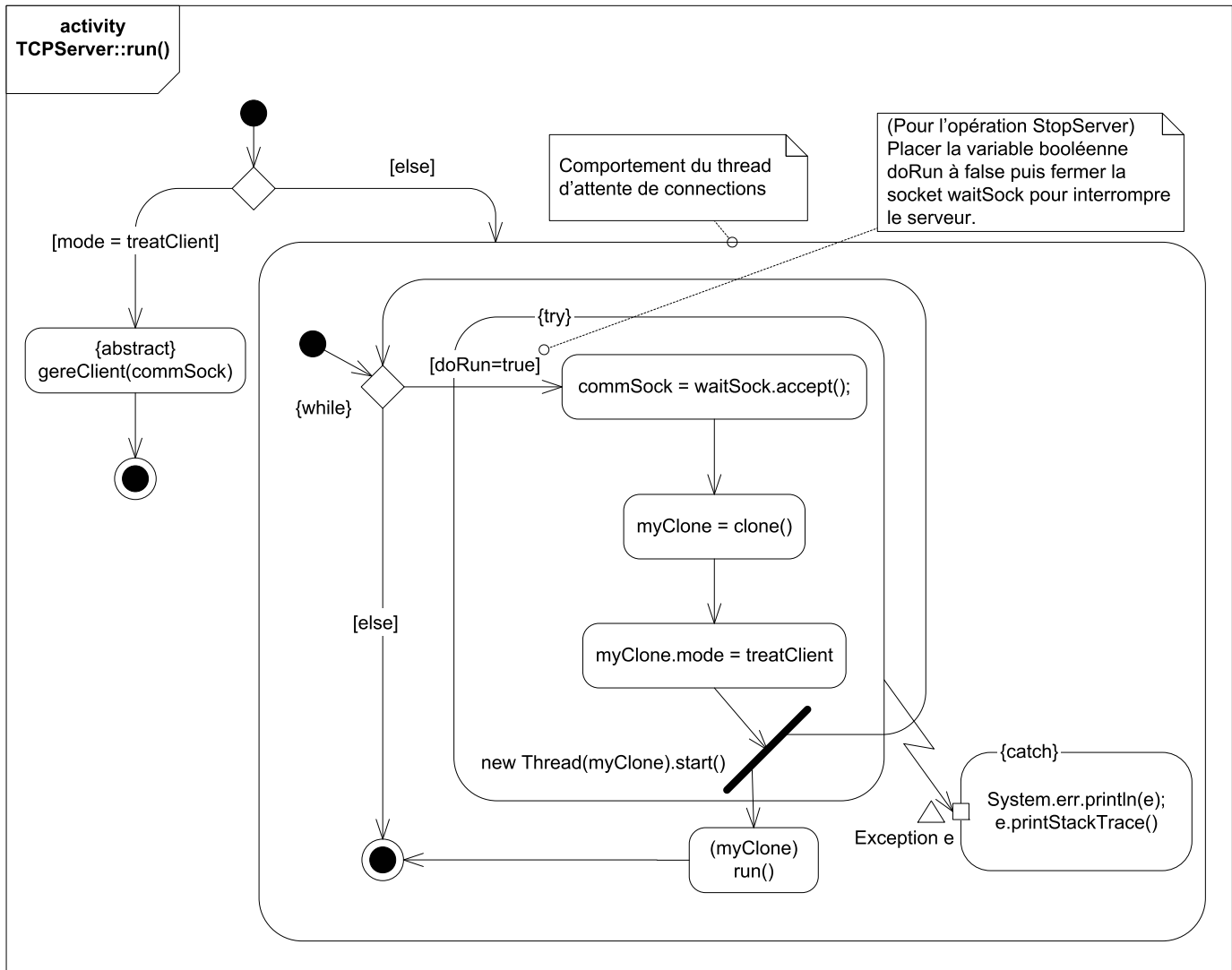
Le package java.net standard propose déjà une implémentation de ces concepts (java.net.Authenticator...), à utiliser de préférence.

On va s'appuyer sur la sérialisation de Java pour implémenter le service de persistance par fichier.

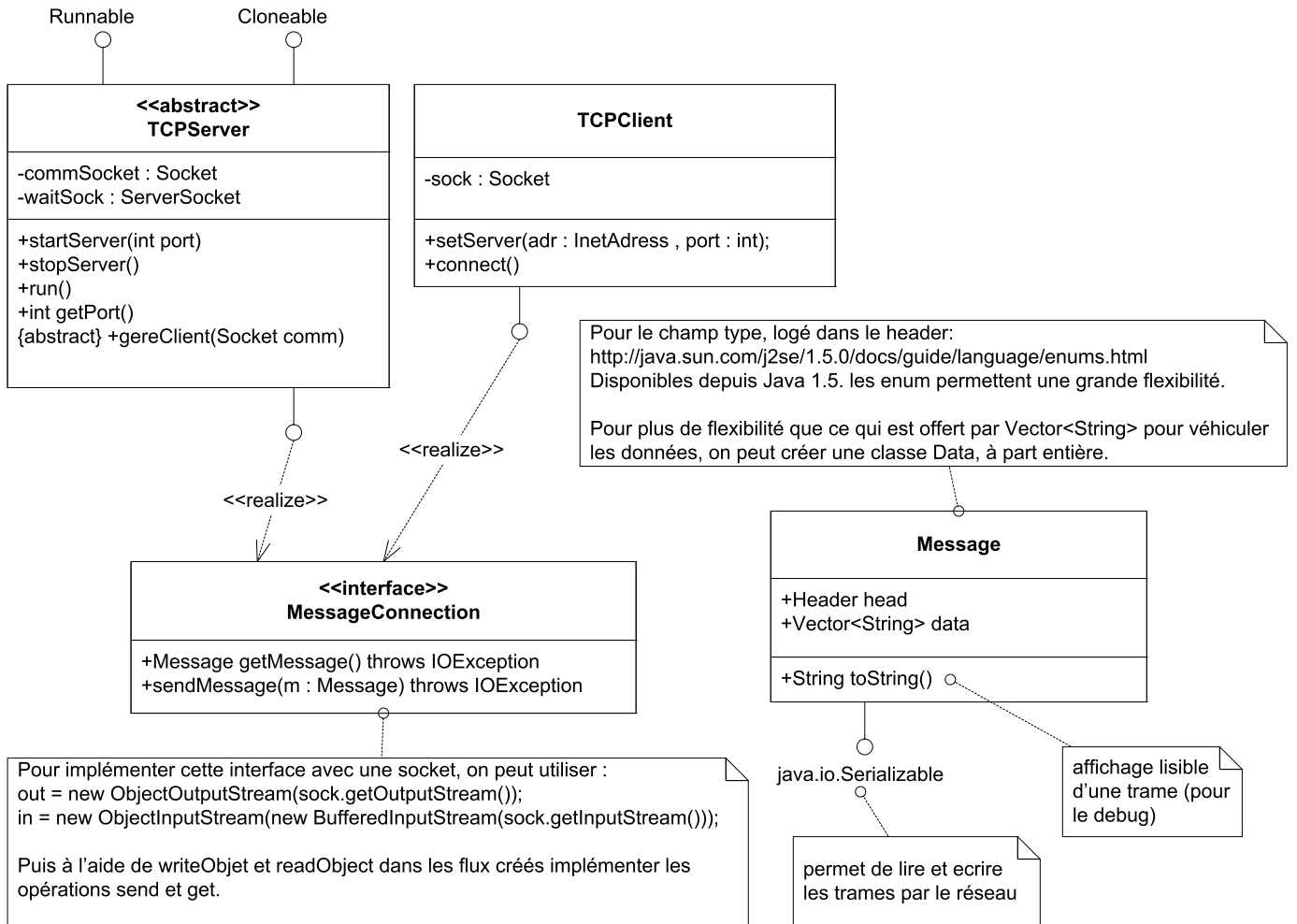


On utilisera un SortedSet pour implémenter cette association. Ceci impose que les objets contenus implémentent Comparable et qu'ils surchargent correctement la méthode equals de Object. Ces deux implémentations doivent se contenter de comparer les login, leur role est de permettre rapidement de trouver l'entrée qui correspond à un utilisateur.

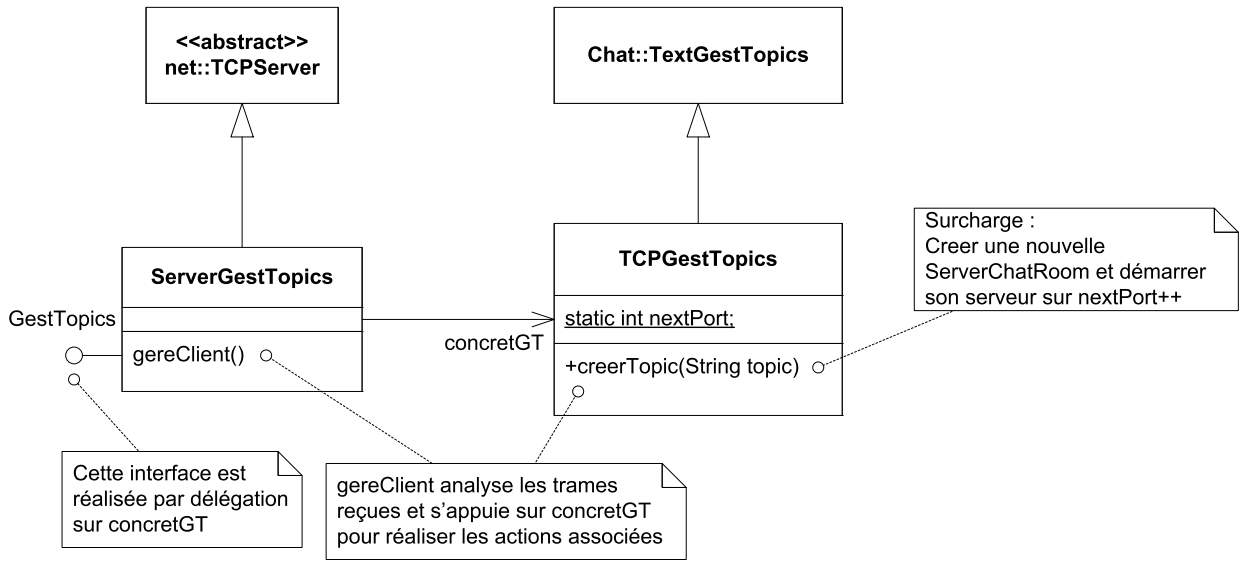
Une implémentation possible de l'interface liée au service d'authentification



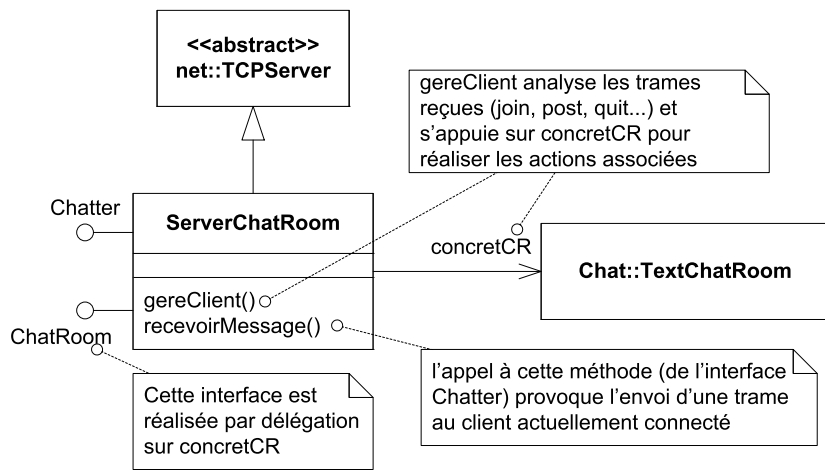
Activité décrivant le comportement d'un serveur TCP multi threadé



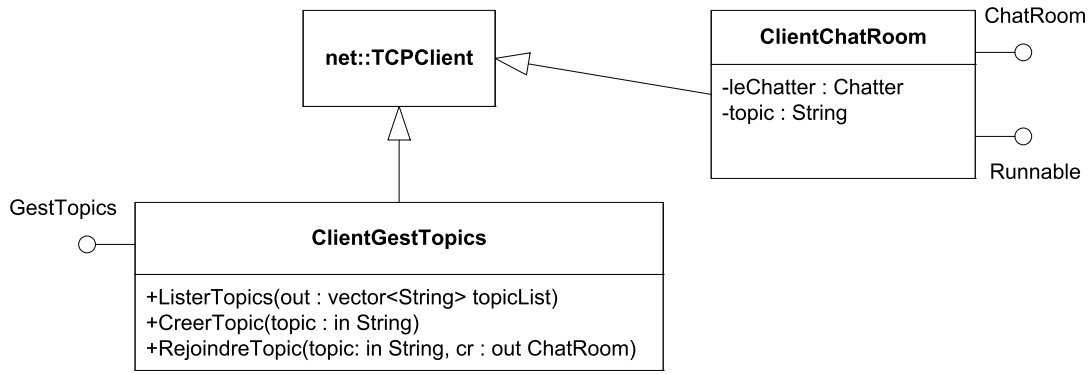
Détail des classes implémentant les connexions TCP



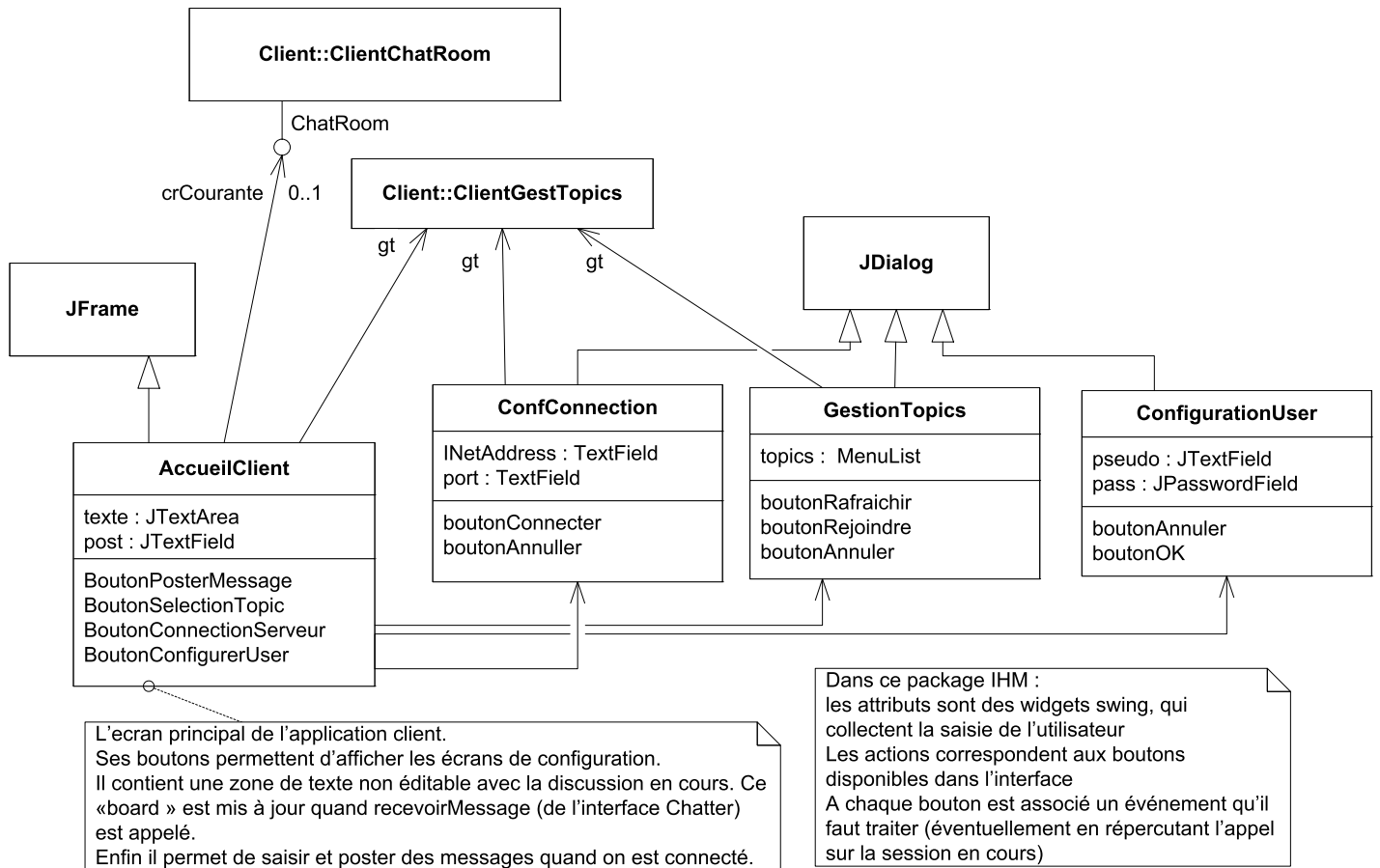
Structure finale du package Server : partie gestion des topics



Structure finale du package Server : partie gestion des chatroom



Structure finale du package Client



implém retenue pour l'IHM du client