Document de spécification

Majeure RES 302

Groupe 1

CUI Yanni

yanni.cui@telecom-bretagne.eu

CHEN Li

li.chen@telecom-bretagne.eu

24/02/2017

Table des matières

I.Introduction

Afin de réaliser un système de messagerie instantanée dans lequel nous pouvons nous connectons et échangeons des messages entre nous , nous défions un en-tête de 16 bit incluant plusieurs champs de différents types. Chaque ou plusieurs champs ensembles vont indiquer un certain rôle d'un message, et ils peuvent réaliser une fonction du système.

Les fonctions suivantes sont disponibles dans ce protocole :

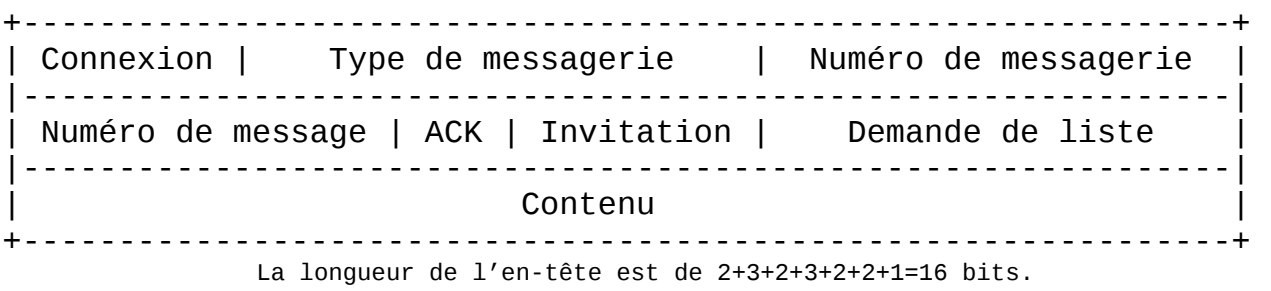
- Connexion avec un nom d'utilisateur.
- Message bidirectionnel ayant au minimum une chaîne de caractère.
- Demande de la liste des utilisateurs actuellement connectés au serveur ainsi qui leur statut.
- Invitation à une messagerie privée.
- Offrir deux modes de messagerie privées. (mode centralisé ou décentralisé)
- Accepter ou refuser une invitation.
- Déconnexion.

Ce protocole fonctionne au-dessus d'UDP, alors pour qu'assurer la fiabilité de la liaison entre client et serveur, il faux fournir des champs de « numéro de message » et de « ACK ». Plus précisément, le champ « numéro de message » donne l'ordre de chaque message pour que le récepteur ne confond pas l'ordre, et le champ d'« ACK » est pour une rétroaction efficace. En plus, le système a un mécanisme de retransmission si le temps d'attente pour un ACK est dépassé ou le NACK arrive.

II.Description de message

Description globale :

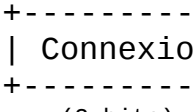
Dans notre protocole, nous avons des champs de connexion(2 bits), du type de messagerie(3 bits), du numéro de messagerie(2 bits), du numéro de message(3 bits), de l'ACK(2 bits), de l'invitation(2 bits) et de demande de liste(1 bit).



La longueur de l'en-tête est de 2+3+2+3+2+2+1=16 bits.

Chaque champ va être expliqué précisément au-dessous. Par ailleurs, la longueur du champ « contenu » n'est pas défini, c'est-à-dire que le message dans notre protocole a une longueur variable.

1. Connexion/Déconnexion



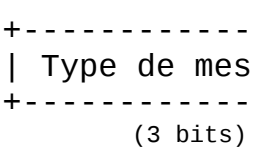
Le champ de connexion est pour indiquer l'état de connexion d'un client.

Ses valeurs et les définitions correspondes sont présentés au dessous :

Valeur	Définition
00	Demande une connexion d'un client vers le serveur.
01	Le serveur accepte la demande de connexion d'un client.
10	Le serveur refuse la demande de connexion d'un client.
11	Demande une déconnexion d'un client vers le serveur.

Dès qu’un utilisateur réussit à connecter le serveur, la valeur de ce champ va rester 01.

2. Type de messagerie

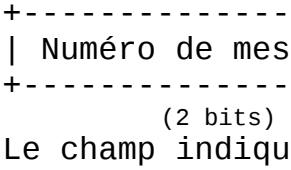


Le champ indique le type de messagerie, mais il aussi contient les demande d’initialiser ou quitter une messagerie privée :

Valeur	Définition
000	Messagerie publique.
001	Messagerie privée et centralisée.
010	Messagerie privée et décentralisée.
101	Demande d’initialiser une messagerie privée et centralisée.
110	Demande d’initialiser une messagerie privée et décentralisée.
111	Demande de quitter la messagerie en cours.

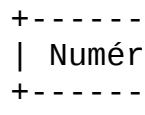
Nous supposons que tous les nouveaux utilisateurs seront mis dans la messagerie publique par défaut, alors la demande de connexion est transmise dans la messagerie publique.

3. Numéro de messagerie



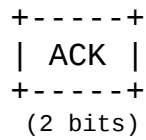
Le champ indique que de quelle messagerie ce message vient. Il a une longueur de 2 bits donc dans notre protocole nous pouvons avoir 1 messagerie publique (unique) et 4 messageries de chaque mode privé maximum(9 messageries en total).

4. Numéro de message



Le champ indique l’ordre de message et il peut être utilisé dans le cas de renvoyer un message perdu ou erroné. Afin de numéroter tous les messages, il applique la méthode du module, c’est-à-dire que le champ ne peut que numéroter des messages de 0 à 7, quand le huitième message arrive, il sera numéroté comme 0.

5. ACK/NACK

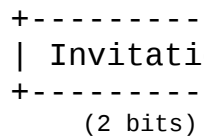


Le champ nous permet de la fiabilité de la liaison avec le champ du numéro de message.

Chaque fois l’émetteur envoie un message, le récepteur doit renvoyer un feed-back :

Valeur	Définition
00	Le cas l’émetteur envoie un messager.
01	Le message est bien reçu par un récepteur.(ACK)
10	Le message est erroné et refusé par récepteur.(NACK)
11	Le message a été envoyé par l’émetteur et l’émetteur a envie d’une réponse du récepteur, mais l’émetteur ne reçoit pas la répondu quand le temps d’attente est dépassé, alors l’émetteur va envoyer un ‘WAIT’ pour dire au récepteur qu’il n’a pas reçu la réponse du message, donc soit le récepteur enverra une réponse ayant le même numéro de message que le dernier s’il a déjà répondu, soit il enverra une nouvelle réponse s’il n’a pas répondu au message.

6. Accepter/Refuser une invitation

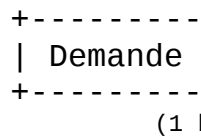


Le champ de l’invitation doit fonctionner avec le champ « contenu » où le serveur trouve la liste du nom invité.

Ses valeurs et les définitions correspondes sont présentés au dessous :

Valeur	Définition
00	Aucune invitation n’est envoyée.
01	Un utilisateur accepte l’invitation d’autre utilisateur.
10	Un utilisateur refuse l’invitation d’autre utilisateur.
11	Un utilisateur envoi une invitation au serveur portant une liste des noms d’utilisateur dans le champ « contenu » pour que le serveur peut transférer l’invitation aux certains utilisateurs.

7. Demande de liste



Le champ de demande de liste est désignée pour qu’un client demande au serveur une liste des utilisateurs actuellement connectés, le serveur doit renvoyer la liste des noms d’utilisateur ainsi que leur statut.

Ses valeurs et les définitions correspondes sont présentés au dessous :

Valeur	Définition
0	Pas de demande la liste d’utilisateurs connectés.
1	Demande la liste d’utilisateurs connectés.

III.Exemples de message d’erreur

Dans notre protocole, les message d’erreur sont considérés comme les message normaux, ils sont stockés dans le champ « contenu ». S’il y a une erreur se produit, le champ « contenu » ne porte que le message d’erreur. Nous notons ‘message d’erreur’ MDE à la suite de cette spécification.

MDE1 : Le nom d’utilisateur est déjà utilisé.
Ce message apparaît quand la demande de connexion d’un client est refusé par le serveur car un client connecté a pris le même nom que le nouvel client veut utiliser.

MDE2 : Le serveur ne peut plus accueillir un nouvel client.
Ce message apparaît quand la demande de connexion d’un client est refusé par le serveur car celui a un nombre maximal de client connectés.

MDE3 : Le message que vous venez d’envoyer n’est pas reçu par le récepteur.
Ce message apparaît quand l’émetteur n’a pas reçu le « ACK » ou « NACK » du récepteur après un temps d’attente fixé.

MDE4 : Le message que vous venez d’envoyer est rejeté par le récepteur.
Ce message apparaît quand l’émetteur reçoit un « NACK » du récepteur.

MDE5 : L’utilisateur que vous avez invité n’est pas sur ligne.
Ce message apparaît quand le nom d’utilisateur indiqué dans le message d’invitation n’existe pas dans la liste d’utilisateur connecté.

IV.Exemples d’échanges entre le serveur et les clients

Scénario 1 : Connexion et déconnexion d’un nouvel utilisateur

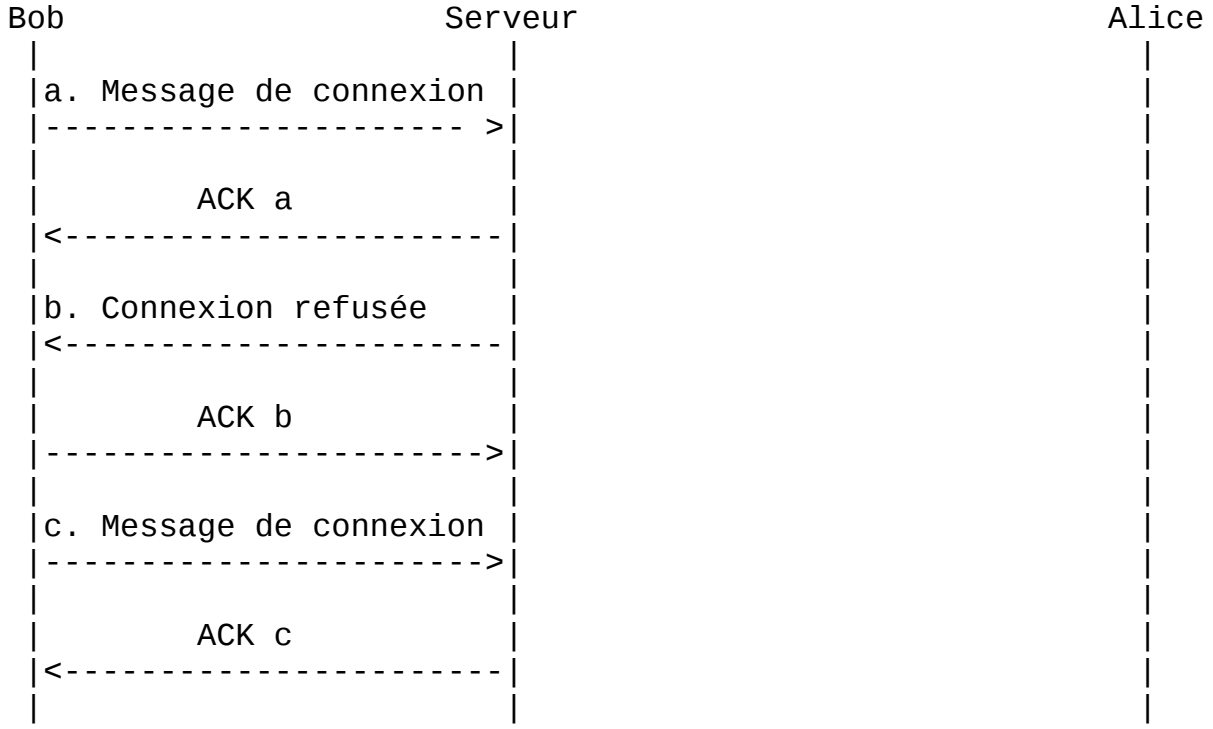
Dans le cas un nouvel utilisateur veut connecter au système, il faut préciser l’adresse IP et numéro de port de cet utilisateur ainsi que son nom, le dernier sera stocké dans le champ « Contenu ».
Le serveur va refuser la demande de connexion en retournant un message d’erreur si le nom d’utilisateur est déjà pris par un autre utilisateur ou le nombre d’utilisateur connectés a atteint le nombre maximal. (connexion = 10)

Si le serveur a accepté cette demande, il enverra un message retour au utilisateur pour confirmer sa connexion. (connexion = 01)

Si un utilisateur veut quitter le système, il faut envoyer une demande de déconnexion au serveur, et après le serveur a supprimé des data de cet utilisateur, le serveur renverra une message de confirmation.(connexion = 11)

Dès qu’un utilisateur a connecté au serveur, tous les messages qu’il envoie contient un champ de connexion égale 01.

Diagramme de séquence :



Les détaille de chaque message sont présentés au suivant :

Connexion	Type de messagerie	Numéro de messagerie
Numéro de message	ACK	Invitation	Demande de liste
Contenu		

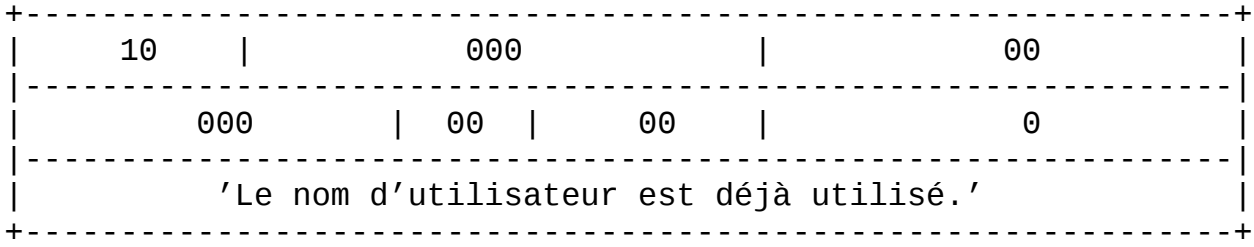
a. Message de connexion :

00	000	00
000	00	00	0
‘Alice’		

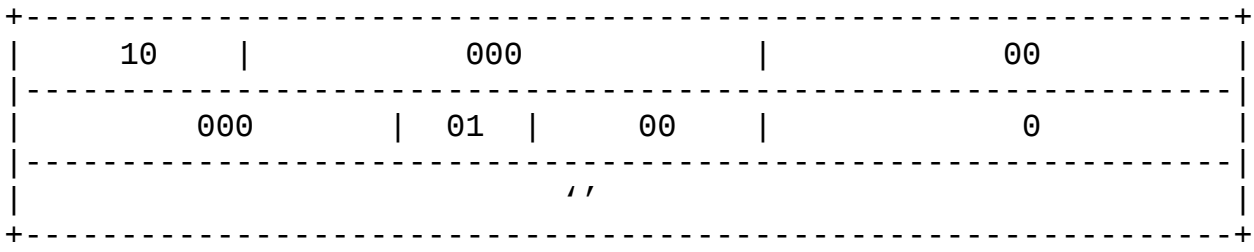
ACK a :

00	000	00
000	01	00	0
’’		

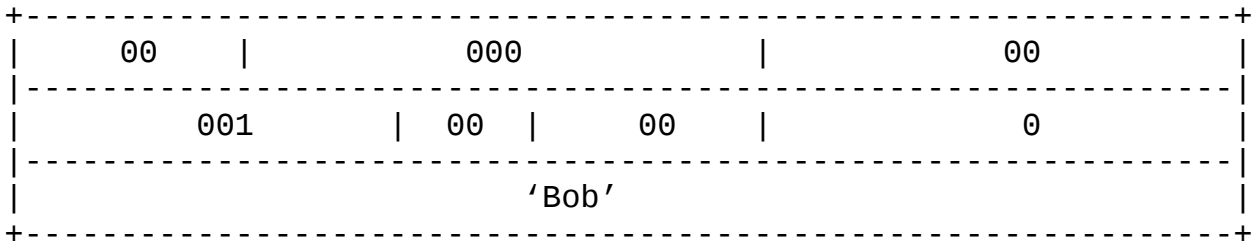
b. Connexion refusée :



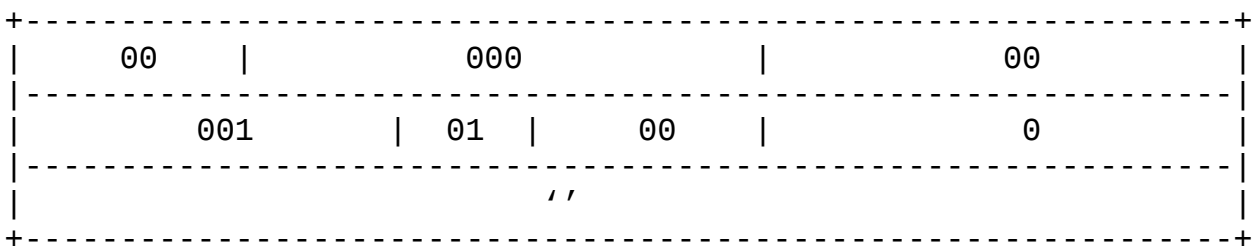
ACK b :



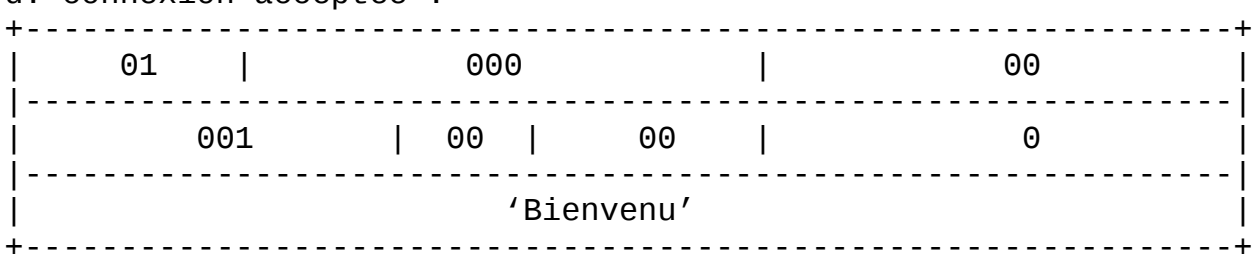
c. Message de connexion :



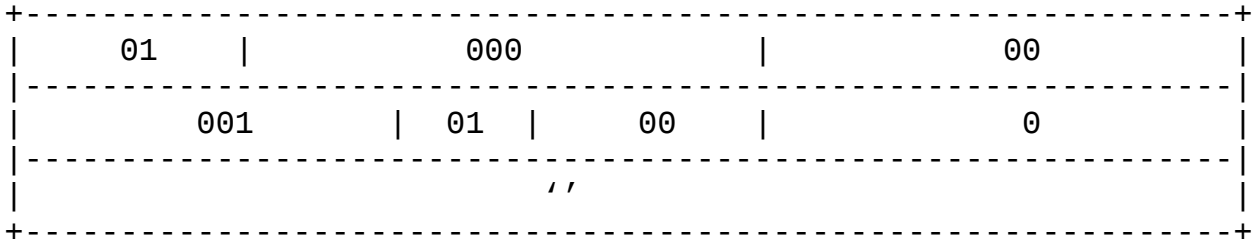
ACK c :



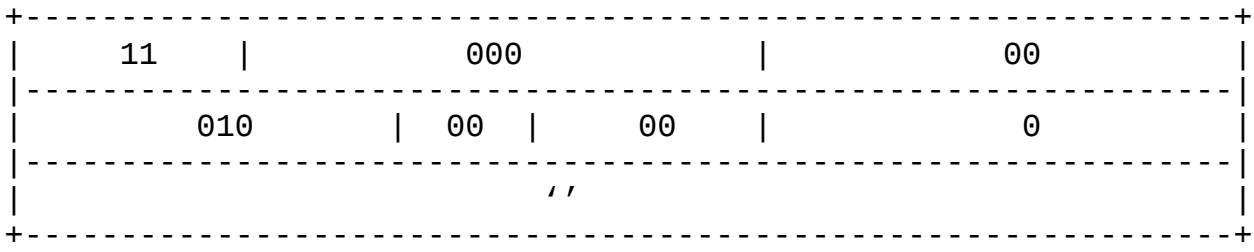
d. connexion acceptée :



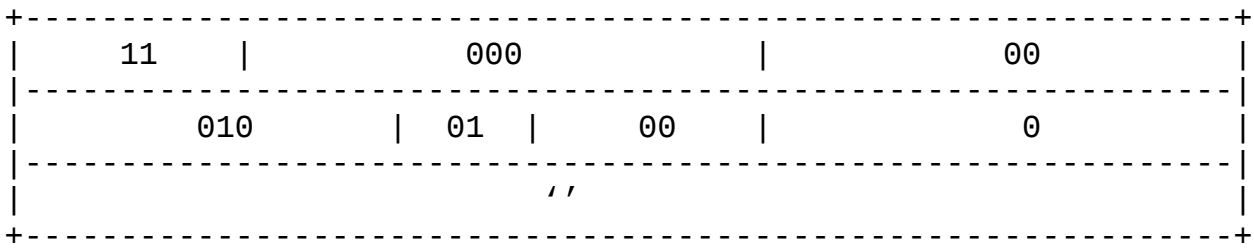
ACK d :



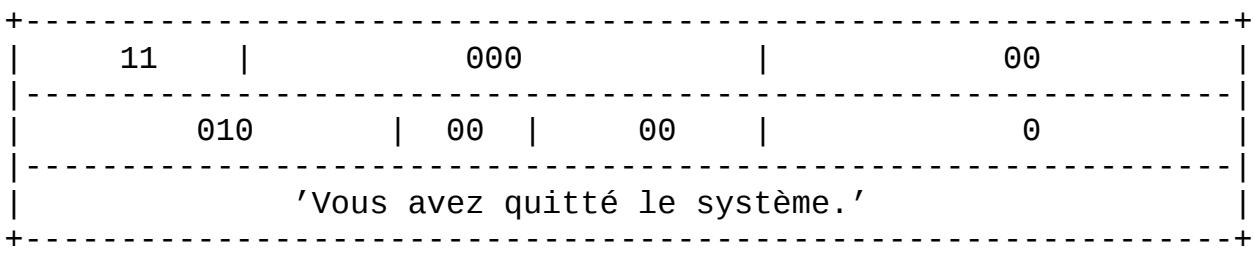
e. Message déconnexion :



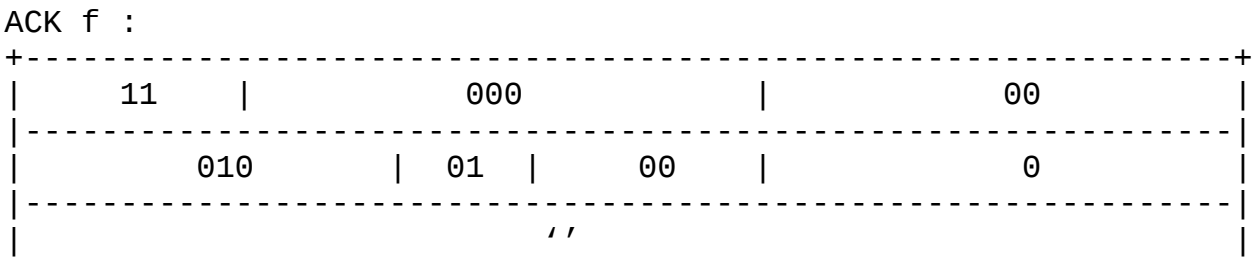
ACK e :



f. Conf de déconnexion :



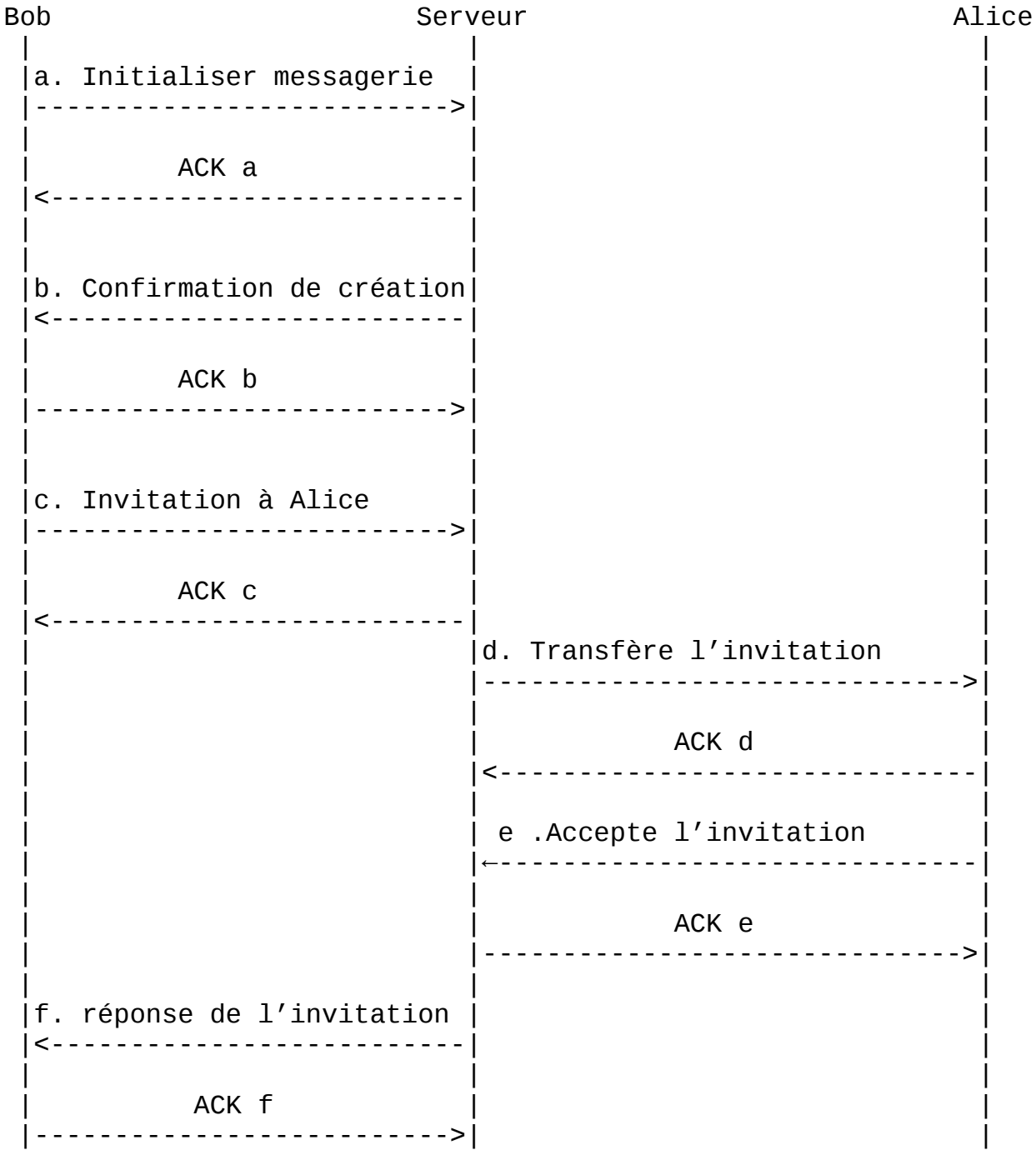
ACK f :

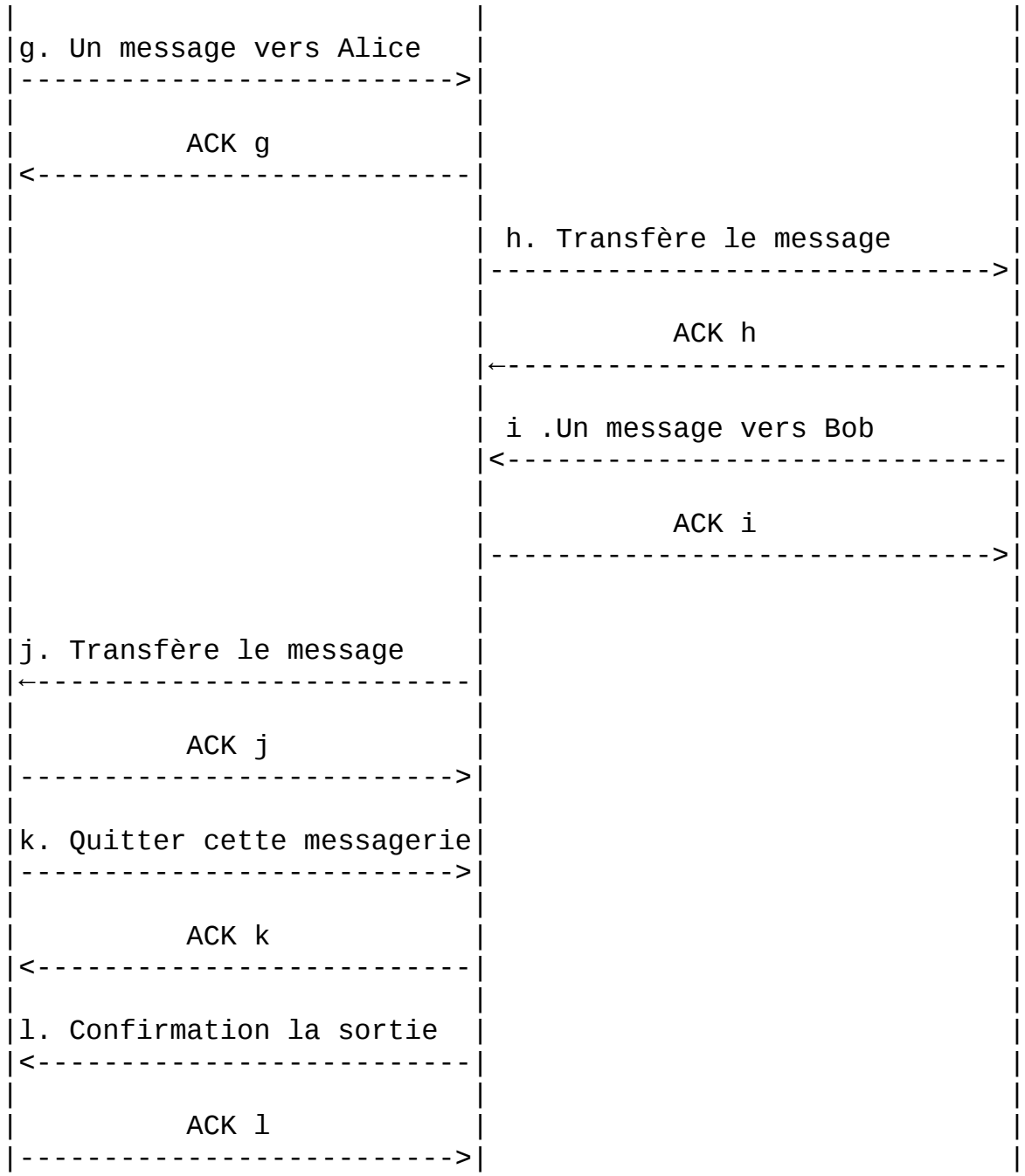


Scénario 2 : Invitation et messagerie privée

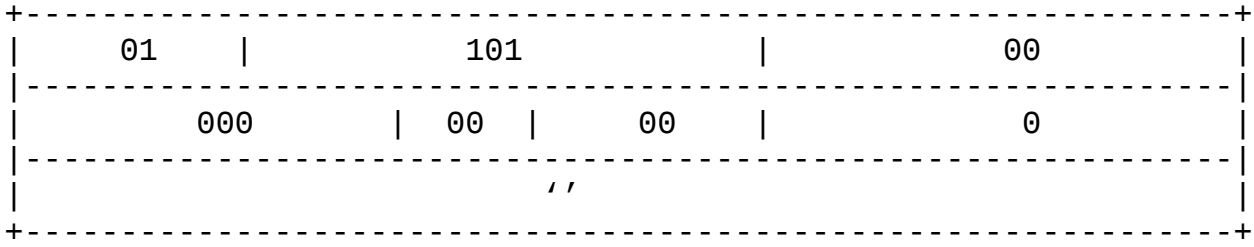
Nous supposons que l’utilisateur Bob et l’utilisateur Alice sont tous en ligne maintenant, et ils sont à la messagerie publique à l’instant, mais Bob va initialiser **une messagerie privée centralisée** et il veut inviter Alice cette messagerie. Après avoir reçu l’invitation, Alice l’accepte sans doute. 10 minutes après, Bob quitte la messagerie, par contre, Alice reste dans celle-ci.

Diagramme de séquence :

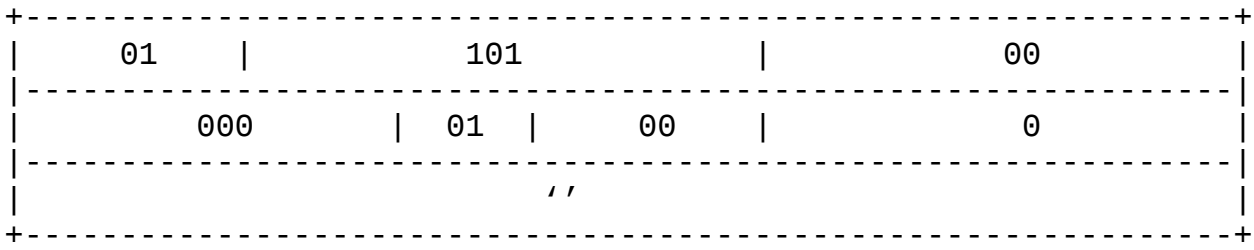



g. Un message vers Alice
ACK g
h. Transfère le message
ACK h
i .Un message vers Bob
ACK i
j. Transfère le message
ACK j
k. Quitter cette messagerie
ACK k
l. Confirmation la sortie
ACK l

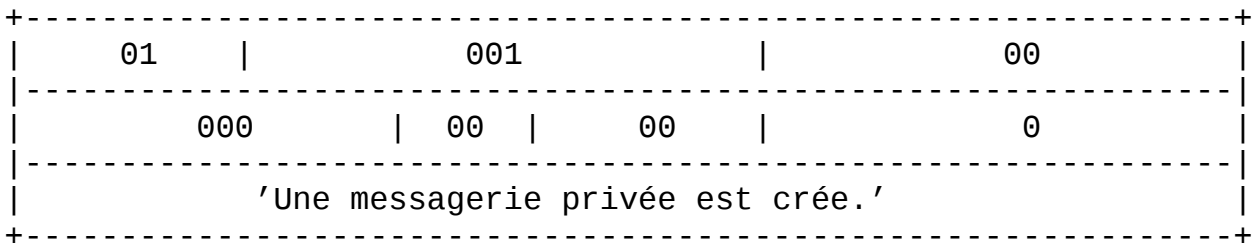
a. Initialiser messagerie :



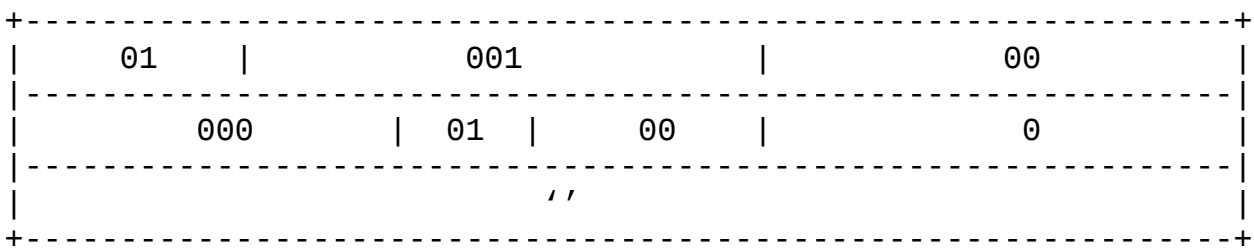
ACK a :



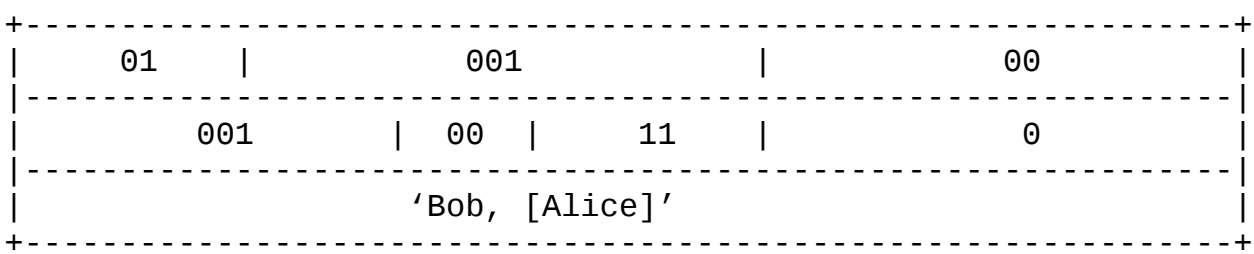
b. Confirmation de création :



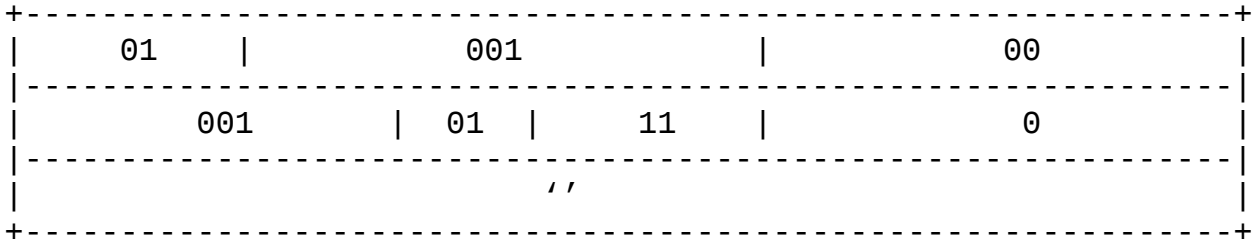
ACK b :



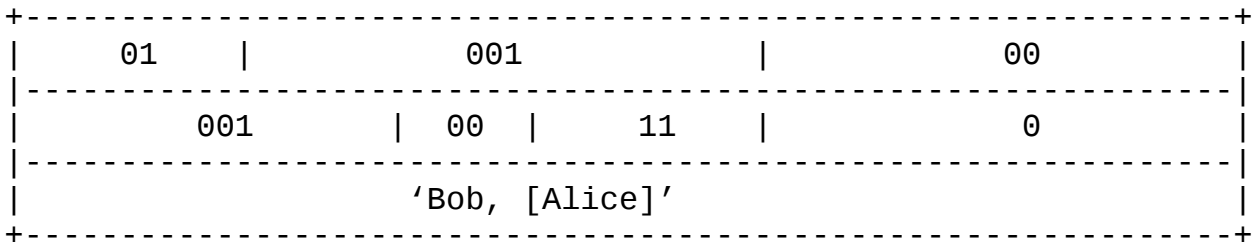
c. Invitation à Alice :



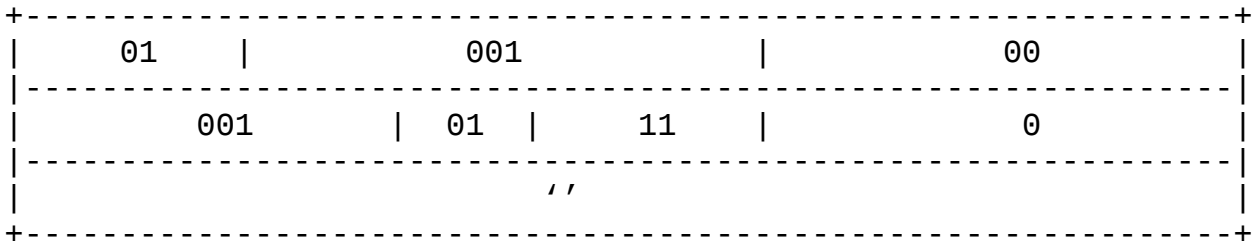
ACK c :



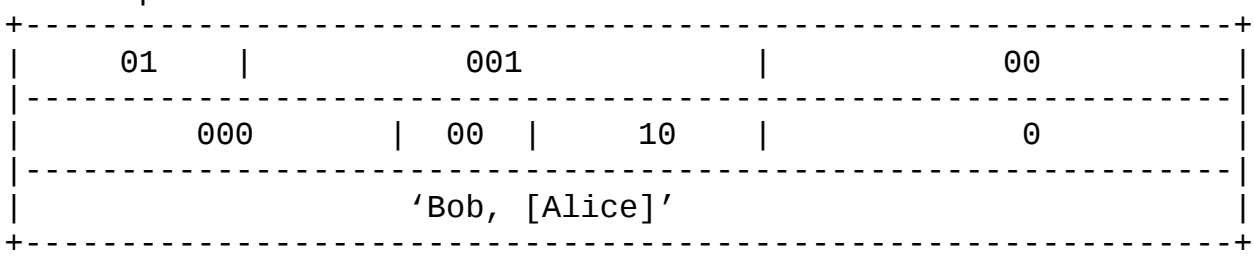
d. Transfère l’invitation :



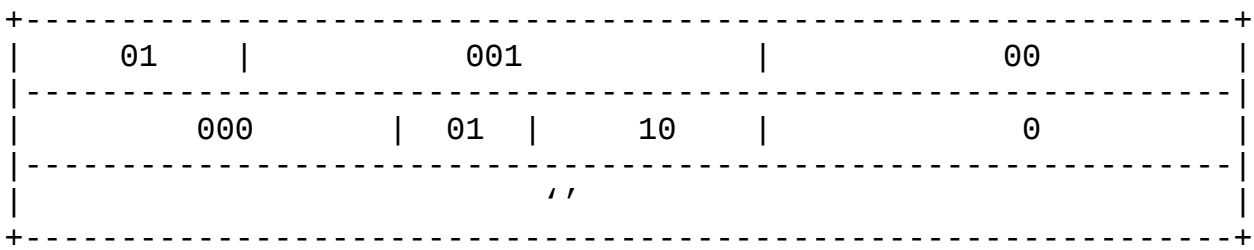
ACK d :



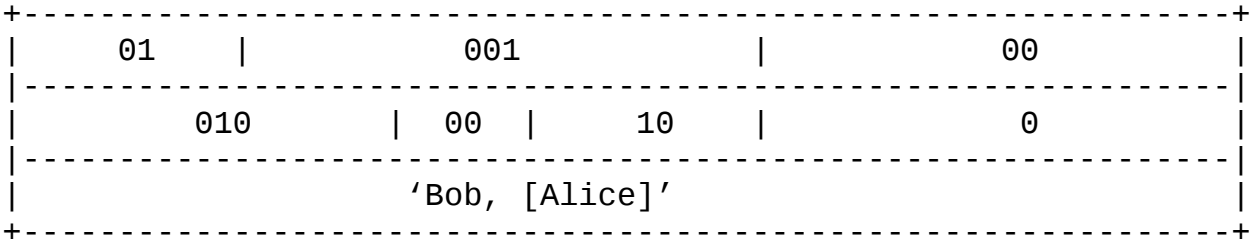
e .Accepte l’invitation :



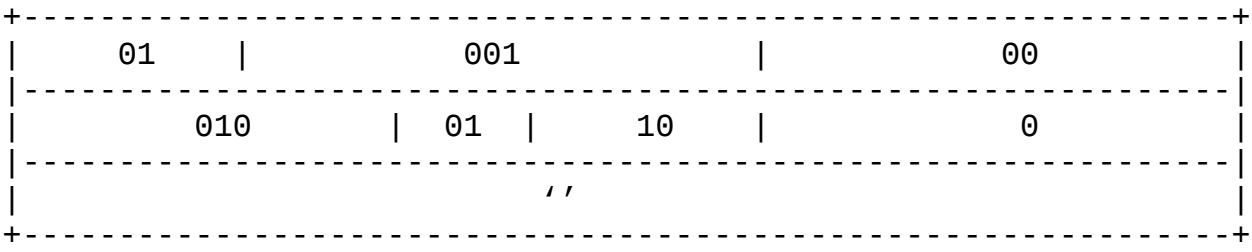
ACK e :



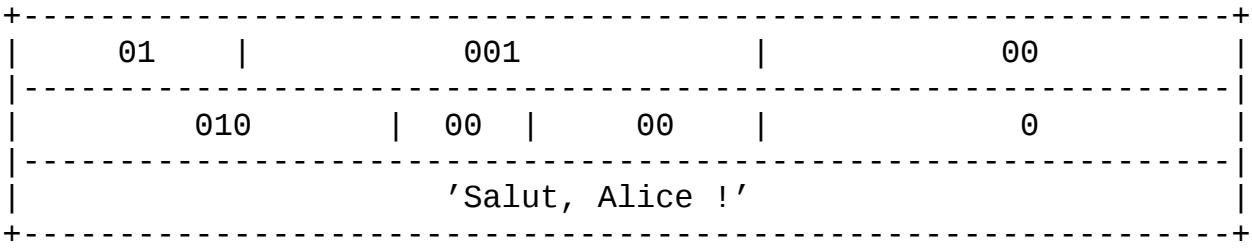
f. réponse de l’invitation :



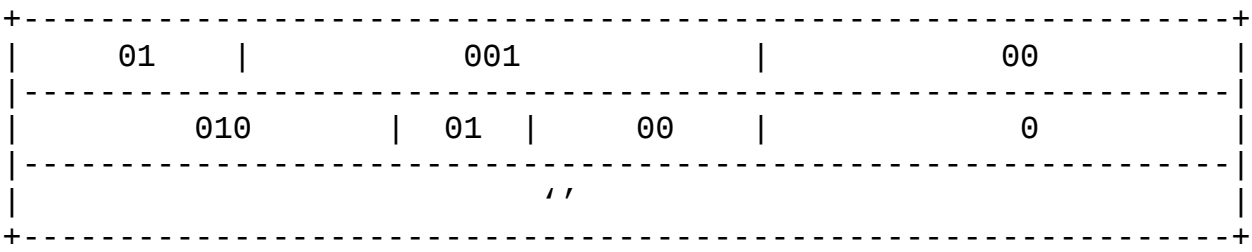
ACK f :



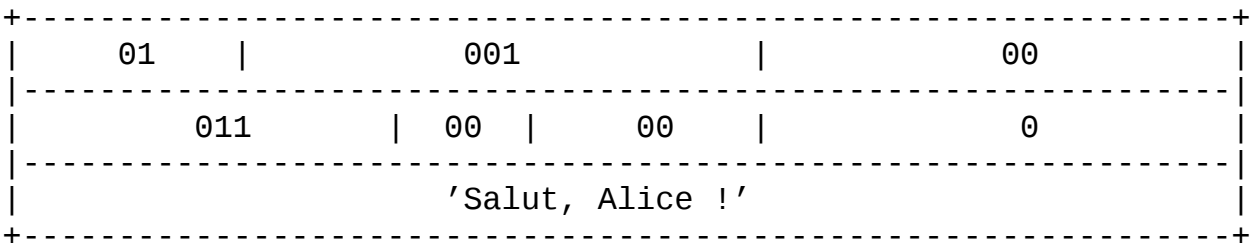
g. Un message vers Alice :



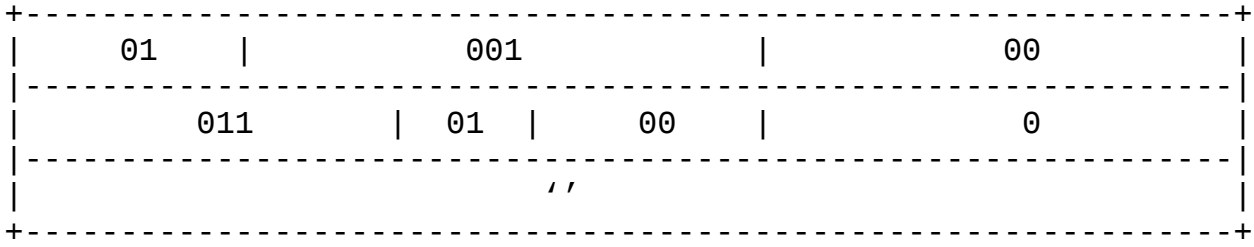
ACK g :



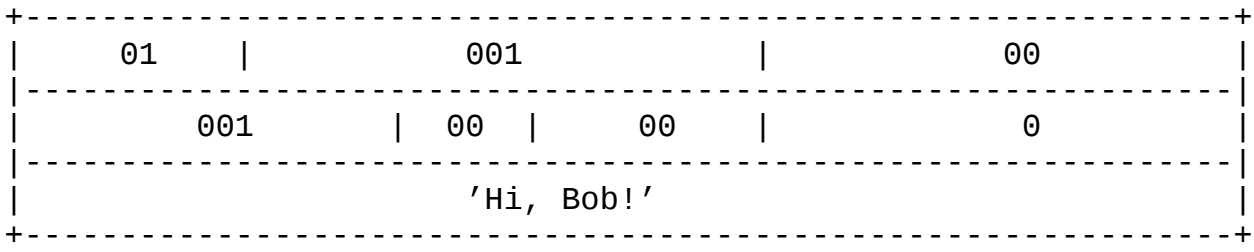
h. Transfère le message :



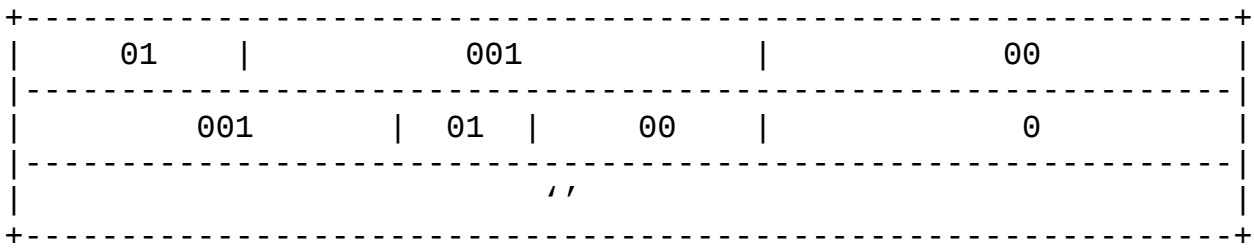
ACK h :



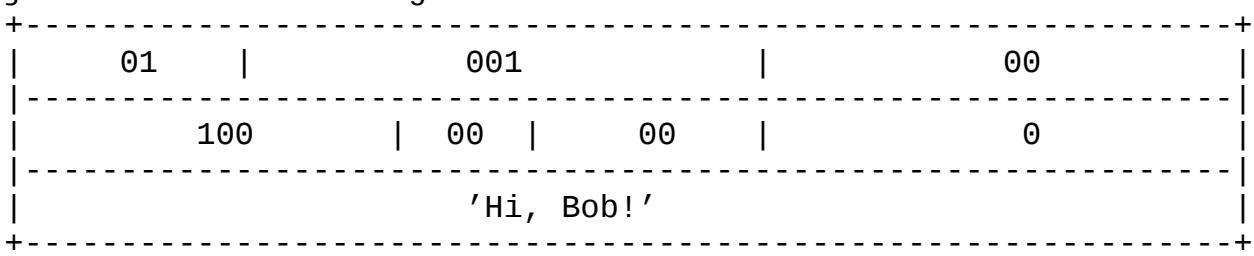
i .Un message vers Bob :



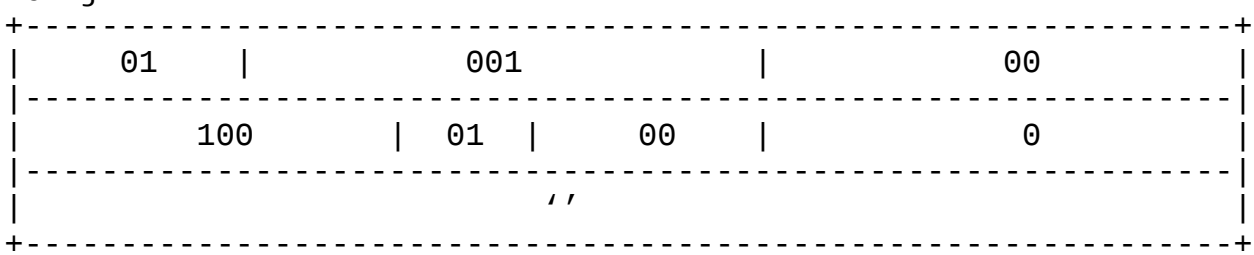
ACK i :



j. Transfère le message :



ACK j :



k. Quitter cette messagerie :

01	111	00
011	00	00	0
	’’	

ACK k :

01	111	00
100	01	00	0
	’’	

l. Confirmation la sortie :

01	000	00
101	00	00	0
	’Sortie Confirmée.’	

ACK l :

01	000	00
101	01	00	0
	’’	

Maintenant Alice est la seule personne qui reste dans la messagerie, et si elle a aussi quitté cette messagerie, celle va fermer tout de suite, autrement dit, il n’y a pas d’une messagerie qui contient aucune personne.

V.Résumé

C’est un document spécification qui propose un protocole du système de messagerie instantanée, centralisé sur un serveur. Chaque champ de message a une description détaille qui défini sa longueur et ses fonctionnalités. Les messages d’erreur sont donnés dans la section « Exemples de message d’erreur ». Nous développons deux scénarios qui vous montrent comment notre protocole marche. En mettant en œuvre ce protocole, nous pouvons réaliser un système de messagerie instantanée qui nous assure la fiabilité de la liaison.