

⑫ DEMANDE DE BREVET D'INVENTION

A1

⑫② Date de dépôt : 01.10.92.

⑫③ Priorité : 30.10.91 US 785139.

⑫④ Date de la mise à disposition du public de la demande : 07.05.93 Bulletin 93/18.

⑫⑤ Liste des documents cités dans le rapport de recherche : *Le rapport de recherche n'a pas été établi à la date de publication de la demande.*

⑫⑥ Références à d'autres documents nationaux apparentés :

⑦① Demandeur(s) : APPLE COMPUTER, INC. — US.

⑦② Inventeur(s) : Seitz Gregory William.

⑦③ Titulaire(s) :

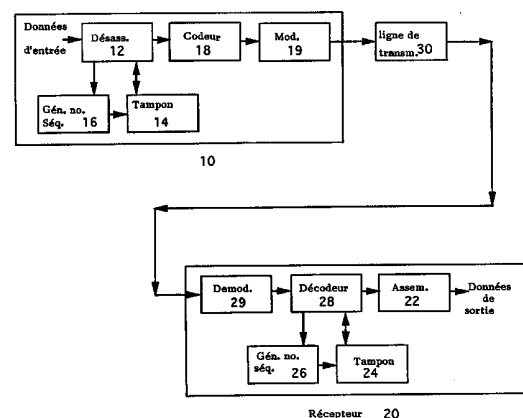
⑦④ Mandataire : Cabinet Regimbeau Martin Schrimpf Warcoin Ahner.

⑤④ Procédé et appareil pour réduire la transmission de données par la mise en antémémoire indexée de données.

⑤⑦ Ce procédé et cet appareil réduisent la transmission de données dans un réseau de communication par la mémorisation et le rappel indexé d'une partie des données à partir d'un cache au lieu de les transmettre.

Selon ce procédé, un cache indexé (14) est maintenu dans un émetteur (10) et un cache indexé (24) synchronisé avec le premier est maintenu dans un récepteur (20). Les caches contiennent des données de transmissions précédentes. Lorsque des informations identiques sont à transmettre ensuite, elles peuvent être remplacées par un drapeau et un index désignant les données mises en cache. La quantité de données à transmettre est ainsi réduite et la vitesse de transfert apparente est accrue.

Applicables en particulier au réseau Apple "AppleTalk".



Cette invention concerne un procédé pour réduire la transmission de données sur un réseau de communication de données par l'obtention de données d'un cache au lieu de par la retransmission des données, ainsi qu'un appareil pour la mise en oeuvre du procédé.

Il arrive souvent que des informations en double ou presque soient transmises entre un émetteur et un récepteur au cours d'une session de communication. De nombreux protocoles de communication utilisent des paquets de données comportant des informations d'en-tête largement identiques pour chaque paquet. Il en résulte une redondance qui peut être éliminée afin de réduire la quantité de données transmises, ce qui améliore le débit et la rapidité de réponse d'un système de communication interactif.

Selon l'invention, lorsque l'émetteur se prépare à transmettre des données, il indique au moyen d'un drapeau que certaines données sont à stocker dans une antémémoire ou cache chez le récepteur. Lorsque, par la suite, la même information est à transmettre, elle est remplacée par un drapeau et un index désignant les données mises en cache. Le récepteur interprète le drapeau et extrait les données stockées de son cache local. La quantité de données à transmettre est ainsi diminuée et la vitesse de transfert apparente est augmentée.

Selon l'invention, un procédé pour réduire la transmission de données entre un émetteur et un récepteur est caractérisé en ce qu'il comprend les étapes consistant à:

à l'émetteur, coder les données à émettre en groupes de données, chaque groupe de données comportant un drapeau et des données associées, une première configuration de données étant stockée dans un tampon d'émetteur et codée dans un groupe de données présentant une première valeur de drapeau, de nouvelles apparitions de cette première

configuration de données étant codées en une deuxième
valeur de drapeau et un numéro de séquence de la confi-
guration dans ce tampon;
la transmission des groupes de données de l'émetteur au
5 récepteur;
au récepteur, recevoir les groupes de données et, pour
chaque groupe de données, décoder les valeurs de drapeau
et effectuer les opérations indiquées sur les données
associées, les drapeaux ayant la première valeur de
10 drapeau ayant pour effet que les données associées sont
stockées dans un tampon de récepteur; des drapeaux ayant
ladite deuxième valeur de drapeau et ledit numéro de
séquence produisant l'extraction de données du tampon de
récepteur audit numéro de séquence, les données étant
15 destinées à être insérées, au récepteur, dans un flux de
données de sortie.

Un procédé comparable est caractérisé en ce
qu'il comprend les étapes consistant à:
déterminer la plus grande capacité de mémoire tampon
20 disponible à la fois dans l'émetteur et le récepteur;
établir un tampon indexé par les numéros de séquence
consécutifs et initialiser les numéros de séquence à 0 à
la fois à l'émetteur et au récepteur;
recevoir un flux de données d'entrée à l'émetteur, flux
25 qui est à transmettre au récepteur;
séparer le flux de données d'entrée en groupes de
données comportant un drapeau et des données associées;
une première valeur de drapeau indiquant que les données
associées sont stockées dans le tampon d'émetteur et
30 que, au récepteur, les données sont à stocker dans le
tampon de récepteur;
une deuxième valeur de drapeau indiquant que les données
associées contiennent un numéro de séquence auquel des
données peuvent être rappelées à partir du tampon de
35 récepteur pour être utilisées dans le flux de données de
sortie;

transmettre les groupes de données entre l'émetteur et le récepteur;

décoder les valeurs de drapeau des groupes de données et assembler un flux de données de sortie à partir des données stockées et des données transmises.

Selon un mode de mise en oeuvre de l'invention, un appareil pour réduire la transmission de données entre un émetteur et un récepteur comprend:

un émetteur possédant une unité de désassemblage, un tampon d'émetteur, un générateur de numéros de séquence, un codeur et un modulateur;

l'unité de désassemblage étant conçue pour recevoir un flux de données d'entrée dans l'émetteur et pour séparer ce flux de données d'entrée en parties à transmettre et en parties à stocker et transmettre;

le tampon d'émetteur étant couplé à l'unité de désassemblage pour le stockage des parties à stocker;

le générateur de numéros de séquence étant couplé à l'unité de désassemblage et au tampon d'émetteur pour la génération d'un index des parties à stocker;

le codeur étant couplé à l'unité de désassemblage en vue du codage d'un flux de données transmis, comportant des parties avec des première, deuxième et troisième valeurs de drapeau, indiquant respectivement des données à stocker, des données stockées précédemment et des données à utiliser directement;

le modulateur étant couplé au codeur en vue de l'application du flux de données transmis à une ligne de transmission; et

un récepteur possédant un démodulateur, un décodeur, un tampon de récepteur, un générateur de numéros de séquence et une unité d'assemblage;

le démodulateur étant couplé à la ligne de transmission pour recevoir le flux de données transmis;

le décodeur étant couplé au démodulateur pour reconnaître les première, deuxième et troisième valeurs de

drapeau et pour diriger ou acheminer les données en vue de leur stockage, leur appel ou leur utilisation directe respectivement;

le tampon de récepteur étant couplé au décodeur en vue
5 du stockage des données à stocker;

le générateur de numéros de séquence étant couplé au décodeur et au tampon de récepteur pour générer un index des données à stocker; et

l'unité d'assemblage étant couplée au décodeur et au
10 tampon de récepteur en vue de l'élaboration, à l'intérieur du récepteur d'un flux de données de sortie à partir des données rappelées et des données à utiliser directement.

D'autres caractéristiques et avantages de
15 l'invention ressortiront plus clairement de la description qui va suivre d'un exemple de mise en oeuvre non limitatif, ainsi que des dessins annexés, sur lesquels:

- la figure 1 est un schéma fonctionnel d'un émetteur et d'un récepteur de communication de données
20 comprenant des caches indexés selon l'invention;

- la figure 2A montre le désassemblage d'un premier paquet de recherche "AppleTalk NBP" (AppleTalk est une marque déposée de Apple Computer, Inc. et désigne notamment un réseau Apple);

25 - la figure 2B montre le contenu du cache indexé après le désassemblage du premier paquet tel que celui représenté sur la figure 2A;

- la figure 3A montre le désassemblage du deuxième paquet de recherche AppleTalk NBP;

30 - la figure 3B montre le contenu du cache indexé après le désassemblage du deuxième paquet tel que celui représenté sur la figure 3A;

- la figure 4A montre le désassemblage du troisième paquet de recherche AppleTalk NBP;

- la figure 4B montre le contenu du cache indexé après le désassemblage du troisième paquet tel que celui représenté sur la figure 4A;

- la figure 5A montre le désassemblage du
5 quatrième paquet de recherche AppleTalk NBP; et

- la figure 5B montre le contenu du cache indexé après le désassemblage du quatrième paquet tel que celui représenté sur la figure 5A.

La figure 1 est un schéma fonctionnel d'un
10 émetteur et d'un récepteur pour la communication de données avec des caches indexés selon l'invention. Il s'agit plus exactement d'un émetteur 10 et d'un récepteur 20 interconnectés par une ligne de transmission fiable 30. L'émetteur 10 et le récepteur 20 peuvent être
15 constitués par un équipement de communication de données ou, plus généralement, par des dispositifs informatiques tels que des ordinateurs personnels ou des postes de travail d'un ordinateur. La ligne de transmission 30 peut être une liaison de communication point à point
20 spécialisée telle qu'une ligne téléphonique à numérotation ou peut être un réseau de communication à multiples accès.

A l'intérieur de l'émetteur 10 se trouve une source de données d'entrée à transmettre. Ces données
25 d'entrée sont traitées par une unité de désassemblage 12 en vue de la sélection de séquences de données d'entrée à stocker dans une mémoire tampon d'émetteur 14. Toutes les données d'entrée stockées dans le tampon d'émetteur 14 sont indexées par un générateur de numéros de sé-
30 quence 16. La séquence de données à transmettre est préparée ensuite par un codeur 18 et mise en forme compte tenu des caractéristiques de la ligne de transmission 30 par un modulateur 19. Ce dernier pourrait être extérieur à l'émetteur 10 et être compris par
35 exemple dans un modem autonome externe.

A l'intérieur du récepteur 20 se trouve un démodulateur 29 destiné à recevoir les données codées arrivant par la ligne de transmission 30. Les données codées sont traitées par un décodeur 28 qui identifie
5 les drapeaux indiquant que des opérations spéciales sont à effectuer, par exemple que des données sont à stocker dans une mémoire tampon de récepteur 24 ou sont à rappeler de ce tampon. Les données stockées dans le tampon de récepteur 24 sont indexées par un générateur
10 de numéros de séquence 26. Les données venant du décodeur 28 et les données stockées extraites du tampon de récepteur 24 sont combinées dans une unité d'assemblage 22 servant à créer le flux de données de sortie à utiliser dans le récepteur 20.

15 Le procédé selon l'invention réduit les transmissions de données répétitives, telles que des en-têtes de paquets ou des paquets entiers. Le procédé peut être adapté pour la reconnaissance de séquences d'information entières ou partielles. L'émetteur met en
20 cache une quantité spécifiée d'information d'un flux de données d'entrée et ordonne au récepteur de mettre en cache la même information. L'émetteur contrôle ensuite les données futures sur des correspondances avec cette information mise en cache. Si des correspondances sont
25 trouvées, un drapeau et un jeton représentant les données concernées sont introduits dans le flux de données à la place des données effectives. Le flux de données transmis peut comprendre un mélange à la fois de drapeaux, de jetons et de données brutes.

30 Au récepteur, les drapeaux et les jetons sont interprétés comme des instructions pour stocker de l'information dans le cache du récepteur, pour extraire de l'information de ce cache ou pour utiliser les données arrivant directement par le flux de données
35 transmis. Le mélange de données extraites du cache du récepteur et de données brutes est assemblé pour

produire, dans le récepteur, un flux de données de sortie qui est identique au flux de données d'entrée dans l'émetteur.

Ce procédé est basé sur le synchronisme des
5 états des caches de l'émetteur et du récepteur. Une façon simple pour y parvenir consiste à maintenir des caches identiques à l'émetteur et au récepteur et à utiliser une ligne de transmission ou un service de transmission fiable. Cette solution évite une grande
10 complexité dans la gestion de caches de différentes capacités et d'indexages différents, ainsi que dans la détection et la réparation d'erreurs de transmission. Le procédé selon l'invention exige que l'émetteur et le récepteur restent synchronisés lorsque leurs caches sont
15 pleins et lorsque des procédures de débordement ou d'actualisation sont utilisées. Il est possible d'employer des caches ayant une capacité identique, laquelle peut être négociée entre l'émetteur et le récepteur dans un échange initial de communication de données. La
20 capacité de cache effectivement employée est la plus petite capacité des deux tampons formés par le tampon d'émetteur 14 et le tampon de récepteur 24. L'utilisation du plus petit des deux tampons assurera que les deux côtés finiront par avoir la même capacité de tampon
25 puisque le côté ayant le plus grand tampon peut toujours réduire temporairement la capacité de tampon qu'il met à disposition. Il est bien entendu possible d'employer des schémas de gestion de tampons plus compliqués à condition que les caches et leurs indexes restent synchro-
30 nisés.

Le choix des données à mettre en cache et la façon de les reconnaître lors de la recherche de correspondances dans les données à transmettre consécutivement, sont entièrement la responsabilité de l'émetteur.
35 Celui-ci détermine de quelle façon les données sont à transmettre et comment une partie des données est mise

en cache. Il envoie des drapeaux pour indiquer au côté
émission comment doivent être indexées, mises en cache
et plus tard rappelées les données du tampon. Des
données brutes sont également transmises en vue du
5 stockage dans le cache et pour être utilisées directe-
ment dans le flux de données de sortie.

Les techniques appliquées pour assurer que les
caches de l'émetteur et de récepteur restent toujours en
synchronisme sont essentielles à l'invention. Comme
10 mentionné, un premier pas pour assurer ce synchronisme
consiste à mettre à disposition une méthode par laquelle
les deux côtés peuvent déterminer la capacité de tampon
qui convient pour être utilisée. Afin d'y parvenir, ils
échangent leurs capacités de tampon préférées. Après
15 avoir été informé de la capacité de tampon de l'autre,
chacun choisit une capacité de tampon effective qui est
la plus petite des deux. Ainsi, si l'émetteur peut
mettre à disposition 10000 octets de mémoire et le
récepteur peut mettre à disposition 8000 octets de
20 mémoire, la capacité résultante, celle employée effec-
tivement, est la plus petite des deux, c'est-à-dire 8000
octets. Il est ainsi garanti que chaque opération
effectuée sur le tampon de l'émetteur peut, après
qu'elle a été effectuée, être reproduite exactement et
25 dans le même état du tampon sur le côté réception.

Un deuxième pas pour assurer que les deux
côtés restent en synchronisme est de faire en sorte
qu'ils utilisent des méthodes cohérentes pour ajouter
des articles (éléments d'information) en cache et pour
30 en retirer des articles. La méthode pour stocker des
données dans le cache correspondant doit être la même
des deux côtés. Selon un mode de mise en oeuvre préféré,
le tampon est organisé comme un tampon en anneau. Chaque
entrée ou information introduite dans le tampon, sous la
35 forme d'un paquet par exemple, a la structure suivante
exprimée dans le langage de programmation C:

```

deftype struct PaquetDonnéesT
{
    long          droite;    /*liais p recev, contr-
                             têtesomme p émettre */
5    long          gauche;   /*liaison pour recevoir*/
    court sansigne noséq;    /*n° séquence de `cette
                             entrée */
    court sansigne longdon;  /*longueur données qui
                             suivent*/
10   car sansigne   tamdon[]; /*données suivantes repé-
                             rées ici*/
} PaquetDonnéesT, *PaquetDonnéesTP;

```

Dans ce mode de mise en oeuvre, un bloc de
 15 données ajouté au tampon en anneau doit toujours pouvoir
 s'y insérer sans bouclage. En d'autres termes, s'il n'y
 a pas assez de place à l'extrémité haute du tampon pour
 ranger tout le bloc de données, le tampon doit être
 rebouclé à l'extrémité basse. Ceci est effectué afin de
 20 simplifier le mappage de structures de données sur les
 données à l'intérieur du tampon, au prix d'un léger
 sous-emploi de la place disponible. Il serait très
 incommode de traiter les données si une partie d'elles
 se trouvait à un endroit et l'autre partie quelque part
 25 ailleurs. S'il n'y pas assez de place pour insérer une
 entrée, des articles sont enlevés jusqu'à ce qu'il y ait
 assez de place. Il est important de se rappeler que
 l'émetteur et le récepteur doivent tous deux utiliser la
 même approche pour gérer leurs tampons parce que ceux-ci
 30 doivent toujours rester en synchronisme, avoir les mêmes
 contenus et être indexés avec les mêmes valeurs de
 numéros de séquence.

Dans ce mode de mise en oeuvre préféré, tel
 qu'il vient d'être défini, la zone "droite" est utilisée
 35 pour lier ensemble les enregistrements des totaux de
 contrôle qui décrivent ce bloc de données. La longueur

de l'ensemble de l'enregistrement, lequel comprend les données effectives, doit toujours correspondre à un nombre pair d'octets, ceci afin de garantir que les entrées consécutives aient toujours leurs en-têtes alignées sur des limites paires. Donc, s'il arrive que
5 la longueur effective de l'enregistrement correspond à un nombre impair, un octet est ajouté à la place utilisée dans le cache.

Afin de décrire les données dans ce tampon
10 sous la forme d'une structure pourvue d'un jeton, un numéro de séquence est stocké avec les données. Ce numéro de séquence sera reproduit côté récepteur lorsque celui-ci stocke les données en question. Il est crucial que l'émetteur et le récepteur appliquent la même
15 technique pour générer les numéros de séquence, afin de garantir que les deux côtés restent en synchronisme. Afin d'y parvenir, les deux côtés commencent toujours par le numéro de séquence zéro avant le stockage d'une donnée quelconque. Pour chaque nouvel article introduit
20 dans le tampon de données, un nouveau numéro de séquence est généré en ajoutant "un" au numéro précédent.

Pendant le fonctionnement, lequel commence à l'émetteur, les données d'entrée, habituellement sous la forme d'un paquet, sont désassemblées et leurs parties
25 en relation mutuelle sont séparées selon des définitions connues à l'émetteur. Après que tous les paquets partiels de données ont été ajoutés au tampon, le paquet entier peut être ajouté. L'adjonction de tout le paquet en dernier garantit que le récepteur sera capable de
30 reconstruire le paquet uniquement sur la base de l'information qui précède. Si le paquet entier était ajouté en premier, certaines entrées pourraient se référer au paquet lui-même, auquel cas le récepteur ne serait pas en mesure de reconstruire le paquet puisqu'il lui
35 faudrait pour cela le paquet lui-même.

Lors du codage, chaque paquet contient toujours au moins un groupe de données. Il peut également contenir plus d'un groupe de données. Un tel groupe est composé d'un drapeau suivi de données dont la signifi-
5 cation peut être découverte par le décodage du drapeau. Un groupe de données est une sous-unité du paquet, utilisable pour mélanger différents types de données. Par exemple, il peut être souhaitable de décrire un
10 paquet en utilisant à la fois des données correspondantes employant des numéros de séquence et des données nouvelles envoyées sous leur structure brute.

Selon un mode de réalisation préféré, afin de réduire à un minimum la quantité de données nécessaires pour décrire l'information qui suit le drapeau du groupe
15 de données, une représentation compacte d'un numéro court ou numéro abrégé sans signe a été définie. Cette représentation compacte est connue sous l'appellation "compactnum" et a les caractéristiques suivantes:

1) Si le bit de gauche du premier octet (premier s'ap-
20 plique à une lecture de gauche à droite dans le paquet) est positionné, le bit de poids fort est masqué pour obtenir le numéro. Cela signifie que si la valeur voulue est inférieure à 128, elle peut être décrite dans un octet.

25 2) Si ce bit n'est pas positionné, le numéro est à deux octets et la valeur de poids faible du numéro est obtenue de l'octet suivant. Ce schéma de numérotation limite la valeur maximale d'un "compactnum" à 32767.

Dans le flux de données codées à transmettre
30 de l'émetteur au récepteur, chaque groupe de données est composé d'un drapeau suivi de données dont la signification peut être découverte par le décodage du drapeau. Selon un mode de mise en oeuvre préféré, le drapeau du groupe de données est une valeur d'un octet décomposée
35 dans les zones suivantes:

Bit 7: arrangement -

Ce bit est positionné si les données faisant suite au drapeau en question sont des données d'arrangement. Le positionnement de ce bit est permis seulement
5 s'il est utilisé comme le tout premier drapeau. Lorsque ce bit est positionné, les 6 bits de poids faible représentent le code d'arrangement. Ce mode de mise en oeuvre réserve des codes 0 à 15 à un usage futur. Ce code est employé pour signaler que certains types de protocoles
10 concernent des cas spéciaux et peut servir à arranger des parties d'un paquet qui contiennent des marques de réservation dans le but d'améliorer les chances de trouver une correspondance lors des tentatives de faire correspondre des entrées. Par exemple, quelqu'un peut
15 détenir un paquet possédant typiquement toutes les mêmes données qu'un autre paquet exception faite de quatre octets quelque part au milieu du paquet. Dans ce cas, il serait possible de substituer des zéros à ces quatre octets lors de l'assortiment d'entrées. Le code
20 d'arrangement serait utilisé dans ce cas pour décrire les octets à écrire par-dessus ceux du paquet après que ce dernier a été ré-assemblé. Il est vraisemblable qu'un échange dynamique d'information sera ajouté avant l'envoi de paquets de données quelconques, ce qui
25 permettra à l'émetteur de décrire les codes d'arrangement qu'il utilise et la façon dont le récepteur doit les interpréter.

Bit 6: données paquet -

Ce bit est positionné si les données dans le
30 paquet en question sont effectivement des données. Si ce bit n'est pas positionné, il s'agit d'un message hors bande.

Bit 5: à jeton -

Ce bit est positionné si les données après ce
35 drapeau constituent un jeton. Dans ce cas, le numéro de séquence du jeton est contenu dans le mot qui suit (il s'agit toujours d'un mot et non pas d'un compactnum). Si

ce bit n'est pas positionné, il est supposé que des données brutes suivent. Les données brutes sont précédées d'une valeur de longueur comprimée à moins qu'il ne s'agisse du dernier groupe de données dans le paquet. Le

5 dernier groupe de données brutes dans un paquet représente un cas spécial pour lequel la longueur peut être tirée de la quantité de données restant dans le paquet.

Bit 4: dernier groupe -

Ce bit est positionné si les données qui

10 suivent forment le dernier groupe de données dans le paquet. Ce drapeau doit être présent dans le dernier groupe de données. Si la description concerne des données brutes et si ce drapeau est positionné, la longueur des données est calculée comme indiqué précédemment.

15

Bit 3: gamme -

Ce bit indique si oui ou non le numéro de séquence consécutif décrit une gamme. Si le bit est positionné, le numéro de séquence décrit une partie d'un

20 paquet séquencé (divisé en parties ou séquences) et non pas le paquet en entier. Les octets recherchés de ce paquet sont décrits par les deux numéros compacts qui suivent le mot du numéro de séquence.

Bit 2: à séquencer -

Ce bit indique si oui ou non le groupe de données décrit par ce drapeau est à séquencer, c'est-à-dire à diviser en parties ou séquences pourvues d'un numéro de séquence et à mettre en cache par le récepteur. Si le bit est positionné, le récepteur doit

25 effectuer la mise en cache et séquencer les données décrites dans ce groupe de données. Il peut s'agir, soit de données brutes, soit de données décrites par une séquence. Dans ce dernier cas, elles doivent être décodées avant que l'introduction dans le cache ne soit

30 opérée. Si le bit est mis à zéro, aucune introduction

35

n'est effectuée pour ce groupe de données dans le cache du récepteur.

Bit 1: paquet à séquencer -

5 Ce bit s'applique seulement au premier drapeau dans le paquet. S'il est positionné, tout le paquet doit d'abord être décodé puis introduit dans le cache du récepteur. Il est important que le paquet soit exploré et que toutes entrées demandées dans le paquet soient créées avant que le paquet lui-même ne soit séquencé.

10 Bits 0: réservés

Ces bits sont mis à zéro.

A titre d'exemple, on décrira ci-après le codage, la transmission et le décodage d'un paquet pour
15 un paquet dans le protocole de gestion de réseau "Apple Talk" . Pour plus de renseignements, on peut se reporter à Technical Introduction to the Macintosh Family, chapitre 9: The Macintosh Operating System, pages 181-183, AppleTalk Network Architecture. Voir
20 également l'ouvrage "Inside AppleTalk" de Apple Programmers' and Developers Association.

Cet exemple suit le parcours d'un paquet de recherche NBP lorsqu'il est retransmis et que son identité change. NBP est le protocole d'affectation de
25 noms (Name Binding Protocole) de AppleTalk. Pour l'exécution du protocole d'accès à distance (AppleTalk Remote Access), il est supposé que les zones de source et de destination LAP ou de protocole d'accès de liaison (Link Access Protocol) soient mises à zéro. Il est supposé
30 également que l'octet de longueur soit mis à zéro à l'émission, cet octet étant rempli par l'autre côté après la réception. On suppose qu'aucun paquet n'a encore été transmis. Les figures marquées "A", c'est-à-dire les figures 2A à 5A, montrent le désassemblage et
35 l'indexage des parties des paquets dans le tampon de l'émetteur. Les figures marquées "B", c'est-à-dire les

figures 2B à 5B, montrent les contenus du cache indexé après le désassemblage des premier à quatrième paquets représentés sur les figures 2A à 5A.

Le premier paquet de recherche NBP à émettre a
 5 la forme indiquée ci-après, dans laquelle DDP signifie traitement réparti pour "Distributed Data Processing" et AFP signifie protocole de rangement AppleTalk:

```

en-tête LAP/DDP -> 00 00 02 00 00 00 00 11 11 22 22 11
10                22 02 FE 02

fonction et
identité NBP      -> 21 74
n-uple NBP        -> 22 22 22 FE 00 01= 09 Serveur AFP 01*
```

15 La figure 2A montre le désassemblage du premier paquet de recherche AppleTalk NBP. La figure 2B montre le contenu du cache indexé après le désassemblage du premier paquet représenté sur la figure 2A. Comme il s'agit du premier paquet envoyé, il n'y a pas d'entrée
 20 pour lui ni pour l'une quelconque de ses parties dans le tampon de l'émetteur. Pour cette raison, une entrée est créée pour tout le paquet, avec le numéro de séquence 0, et le drapeau "groupe de données" est positionné pour indiquer que le récepteur doit également créer une
 25 entrée dans son tampon lorsqu'il reçoit ce paquet. Selon l'invention, les parties d'intérêt du paquet sont également séparées. Dans ce cas particulier, il s'agit de l'en-tête combinée LAP/DDP, du n-uple NBP et, bien entendu, du paquet en entier. Ces parties donneront la
 30 possibilité d'y faire correspondre la totalité ou une partie d'un paquet similaire consécutif. Le paquet résultant à envoyer consiste simplement en un drapeau suivi des données brutes dans ce cas:

drapeau groupé données -> 52 (donpaq, dernier groupe,
paquet à séquencer)

en-tête LAP/DDP -> 00 00 02 00 00 00 00 11 11
22 22 11 22 02 FE 02

5 fonction et identité

NBP -> 21 74

n-uple NBP -> 22 22 22 FE 00 01 = 09

Serveur AFP 01*

10 Un paquet a maintenant été mis en séquence et chargé en mémoire tampon. Le deuxième paquet de recherche NBP a une nouvelle identité et possède la forme suivante:

en-tête LAP/DDP -> 00 00 02 00 00 00 11 11 22 22 11
15 22 02 FE 02

fonction et

identité NBP -> 21 75 (à noter le changement
d'identité par rapport à 74)

n-uple NBP -> 22 22 22 FE 00 01 = 09

20 AFP 01*

La figure 3A montre le désassemblage du deuxième paquet de recherche AppleTalk NBP. La figure 3B montre le contenu du cache indexé après le désassemblage du deuxième paquet représenté sur la figure 3A. Cette fois-ci, l'émetteur vérifie s'il a déjà tout le paquet dans son cache et constate qu'il n'en est pas ainsi puisque l'octet d'identité a changé. Il vérifie ensuite s'il y a des parties quelconques qui correspondent, par exemple si les en-têtes LAP/DDP correspondent. L'unité de désassemblage découvre qu'elle a une correspondance pour cette partie du paquet dans la séquence numéro 0, octets 0 à 0 x 0 s (et les suivants) (hexadécimal). Elle délivre en sortie le drapeau "groupe de données" approprié, indiquant qu'un jeton est envoyé pour cette partie du paquet. Elle crée également une nouvelle entrée dans le tampon pour cette partie du paquet, de sorte qu'il

n'est pas nécessaire de la décrire comme une gamme par la suite, et positionne le bit "à séquencer" dans le drapeau "groupe de données"; ceci sera le numéro de séquence 1. Lorsqu'elle reçoit les octets de fonction et d'identité NBP, elle délivre simplement en sortie un drapeau "groupe de données" qui indique qu'il s'agit de données brutes. Elle ne tente pas de les indexer puisque ces octets changent assez fréquemment et on ne réaliserait aucun gain de place en affectant un jeton à deux octets. Ensuite, elle tente d'assortir le n-uple NBP et trouve qu'elle a une correspondance pour cette partie du paquet dans la séquence numéro 0, octets 0 x 12 à 0 x 24. Elle crée également une séquence pour cette partie du paquet et émet le drapeau "groupe de données" qui décrit la gamme trouvée; elle indique aussi au récepteur qu'il doit mettre en séquence cette partie du paquet, étant entendu qu'il s'agira de la séquence numéro 2. Enfin, le paquet en entier est introduit en tant que séquence numéro 3. Le résultat est que les données suivantes sont à envoyer:

	drapeau groupédonnées	-> 6E (donpaq, àjeton, gamme, àséquencer, paqàséquencer)
	numéro de séquence	-> 00 00
25	gamme	-> 80 8s(numéros compacts pour 00 0s)
	drapeau groupédonnées	-> 40 (donpaq)
	long données brutes	-> 82 (numéro compact pour 02)
30	fonction et identité NBP	-> 21 74 (les 2 octets de données brutes)
	drapeau groupédonnées	-> 7C (donpaq, àjeton, gamme, àséquencer)
	numéro de séquence	-> 00 00
35	gamme	-> 92 A4 (numéros compacts pour 12 24)

Le paquet résultant est long de 14 octets, alors que le paquet original était de 37 octets, ce qui correspond à une économie de 23 octets. Si encore un autre paquet de recherche NBP est envoyé et l'octet
 5 d'identité NBP change une nouvelle fois, on obtient le troisième paquet de recherche NBP suivant à traiter:

```

en-tête LAP/DDP  -> 00 00 02 00 00 00 11 11 22 22 11
                   22 02 FE 02
10 fonction et
   identité NBP    -> 21 76 (à noter le changement
                        d'identité par rapport à 75)
   n-uple NBP      -> 22 22 22 FE 00 01 = 09 Serveur
                        AFP 01*
15
   La figure 4A montre le désassemblage du
   troisième paquet de recherche AppleTalk NBP. La figure
   4B montre le contenu du cache indexé après le désassem-
   blage du troisième paquet représenté sur la figure 4A.
   L'émetteur passera une nouvelle fois par le processus de
20 tentative d'assortiment du paquet. Il ne trouvera pas de
   correspondance pour le paquet en entier puisque l'iden-
   tité NBP a de nouveau changé. Il trouvera une corres-
   pondance pour l'en-tête LAP/DDP mais, cette-fois-ci, il
   la trouvera dans la séquence numéro 1, c'est-à-dire la
25 séquence qu'il avait créé pour cette partie du paquet à
   partir du deuxième paquet de recherche NBP. L'avantage
   réalisé à ce moment est que cette séquence décrit
   exactement la partie du paquet qu'il recherche et qu'il
   pourra décrire sans les octets de gamme. Une description
30 en données brutes est délivrée ensuite en sortie pour
   les octets de fonction et d'identité NBP. On arrive
   ensuite au n-uple NBP, lequel correspond à la séquence
   numéro 2. Là encore, il s'agit d'une correspondance
   directe, de sorte qu'on n'a pas besoin des octets de
35 gamme. Enfin, tout le paquet est mis en séquence en tant

```

que séquence numéro 4. Les données résultantes à transmettre sont:

```

5   drapeau groupédonnées -> 62 (donpaq, àjeton, paqàsé-
    quence)
    numéro de séquence    -> 00 01
    drapeau groupédonnées -> 40 (donpaq)
    long données brutes   -> 82 (numéro compact pour 02)
    fonction et iden-
10  tité NBP               -> 21 74 (les 2 octets de données
    brutes)
    drapeau groupédonnées -> 70 (donpaq, àjeton, dernier
    groupe)
    numéro de série       -> 00 02
15  Cette fois-ci, le système a été capable de
    décrire le paquet en seulement 10 octets.
    Enfin, on suppose que l'émetteur doit ré-
    envoyer le paquet décrit ci-dessus sans changement. On
    obtient alors le quatrième paquet de recherche NBP
20  suivant à traiter:
    en-tête LAP/DDP       -> 00 00 02 00 00 00 00 11 11 22 22
    11 22 02 FE 02
    fonction et iden-
    tité NBP              -> 21 76 (à noter que c'est la même
25  qu'avant)
    n-uple NBP            -> 22 22 22 FE 00 01 = 09 Serveur
    AFP 01*
```

La figure 5 montre le désassemblage du quatrième paquet de recherche AppleTalk NBP. La figure 5B

30 montre le contenu du cache indexé après le désassemblage du quatrième paquet représenté sur la figure 5A. Cette fois-ci, l'émetteur vérifie s'il peut trouver une correspondance pour le paquet entier, et il découvre qu'il détient une telle correspondance dans la séquence

35 numéro 4. Donc, il peut décrire le paquet comme s'il

était composé de seulement une séquence et les données à transmettre sont les suivantes:

5 drapeau groupédonnées -> 70 (donpaq, àjeton, dernier
groupe)
numéro de séquence -> 00 04

10 On voit que le paquet original de 37 octets a pu être décrit en seulement 3 octets. Ainsi que cela est typique avec le trafic NBP, on arriverait à ce stade après seulement 2 paquets parce que, normalement, l'identité NBP ne change pas à chaque paquet envoyé. Par conséquent, dans un trafic typique entre l'émetteur et le récepteur, la quantité de données à transmettre est réduite par l'emploi de caches indexés à l'émetteur et
15 au récepteur.

On voit donc d'après cet exemple de quelle manière les transmissions de données sont réduites par la formation et le maintien de caches indexés pour les données qui ont été envoyées.

20 L'invention n'est pas limitée aux formes de réalisation décrites et l'homme de l'art pourra y apporter diverses modifications sans pour autant sortir de son cadre.

REVENDICATIONS

1. Procédé pour réduire la transmission de données entre un émetteur et un récepteur, caractérisé en ce qu'il comprend les étapes consistant à:
- 5 à l'émetteur (10), coder les données à émettre en groupes de données, chaque groupe de données comportant un drapeau et des données associées, une première configuration de données étant stockée dans un tampon d'émetteur (14) et codée dans un groupe de données
- 10 présentant une première valeur de drapeau, de nouvelles apparitions de cette première configuration de données étant codées en une deuxième valeur de drapeau et un numéro de séquence de la configuration dans ce tampon; la transmission des groupes de données de l'émetteur
- 15 (10) au récepteur (20);
- au récepteur (20), recevoir les groupes de données et, pour chaque groupe de données, décoder les valeurs de drapeau et effectuer les opérations indiquées sur les données associées, les drapeaux ayant la première valeur
- 20 de drapeau ayant pour effet que les données associées sont stockées dans un tampon de récepteur (24); des drapeaux ayant ladite deuxième valeur de drapeau et ledit numéro de séquence produisant l'extraction de données du tampon de récepteur (24) audit numéro de
- 25 séquence, les données étant destinées à être insérées, au récepteur (20), dans un flux de données de sortie.
2. Procédé pour réduire la transmission de données entre un émetteur et un récepteur, caractérisé en ce qu'il comprend les étapes consistant à:
- 30 déterminer la plus grande capacité de mémoire tampon disponible à la fois dans l'émetteur (10) et le récepteur (20);
- établir un tampon (14) indexé par les numéros de séquence consécutifs et initialiser les numéros de séquence
- 35 à 0 à la fois à l'émetteur (10) et au récepteur (20);

recevoir un flux de données d'entrée à l'émetteur, flux qui est à transmettre au récepteur;

séparer le flux de données d'entrée en groupes de données comportant un drapeau et des données associées;

5 une première valeur de drapeau indiquant que les données associées sont stockées dans le tampon d'émetteur (14) et que, au récepteur (20), les données sont à stocker dans le tampon de récepteur (24);

une deuxième valeur de drapeau indiquant que les données
10 associées contiennent un numéro de séquence auquel des données peuvent être rappelées à partir du tampon de récepteur (24) pour être utilisées dans le flux de données de sortie;

transmettre les groupes de données entre l'émetteur (10)
15 et le récepteur (20);

décoder les valeurs de drapeau des groupes de données et assembler un flux de données de sortie à partir des données stockées et des données transmises.

3. Procédé pour réduire la transmission de
20 données selon la revendication 2, dans lequel le tampon d'émetteur (14) et le tampon de récepteur (24) sont maintenus synchronisés du fait que les mêmes opérations sont effectuées sur chacun d'eux pendant que la transmission de données se poursuit.

25 4. Appareil pour réduire la transmission de données entre un émetteur et un récepteur, mettant en oeuvre le procédé selon l'une quelconque des revendications 1 à 3, caractérisé en ce qu'il comprend:

un émetteur (10) possédant une unité de désassemblage
30 (12), un tampon d'émetteur (14), un générateur de numéros de séquence (16), un codeur (18) et un modulateur (19);

l'unité de désassemblage (12) étant conçue pour recevoir un flux de données d'entrée dans l'émetteur (10) et pour
35 séparer ce flux de données d'entrée en parties à transmettre et en parties à stocker et transmettre;

le tampon d'émetteur (14) étant couplé à l'unité de désassemblage (12) pour le stockage des parties à stocker;

le générateur de numéros de séquence (16) étant couplé à
5 l'unité de désassemblage et au tampon d'émetteur (14) pour la génération d'un index des parties à stocker;

le codeur (18) étant couplé à l'unité de désassemblage (12) en vue du codage d'un flux de données transmis, comportant des parties avec des première, deuxième et
10 troisième valeurs de drapeau, indiquant respectivement des données à stocker, des données stockées précédemment et des données à utiliser directement;

le modulateur (19) étant couplé au codeur (18) en vue de l'application du flux de données transmis à une ligne de
15 transmission (30); et

un récepteur (20) possédant un démodulateur (29), un décodeur (28), un tampon de récepteur (24), un générateur de numéros de séquence (26) et une unité d'assemblage (22);

20 le démodulateur (29) étant couplé à la ligne de transmission (30) pour recevoir le flux de données transmis;

le décodeur (28) étant couplé au démodulateur (29) pour reconnaître les première, deuxième et troisième valeurs de drapeau et pour diriger ou acheminer les données en
25 vue de leur stockage, leur appel ou leur utilisation directe respectivement;

le tampon de récepteur (24) étant couplé au décodeur (28) en vue du stockage des données à stocker;

le générateur de numéros de séquence (26) étant couplé
30 au décodeur (28) et au tampon de récepteur (24) pour générer un index des données à stocker; et

l'unité d'assemblage (22) étant couplée au décodeur (28) et au tampon de récepteur (24) en vue de l'élaboration, à l'intérieur du récepteur (20) d'un flux de données de
35 sortie à partir des données rappelées et des données à utiliser directement.

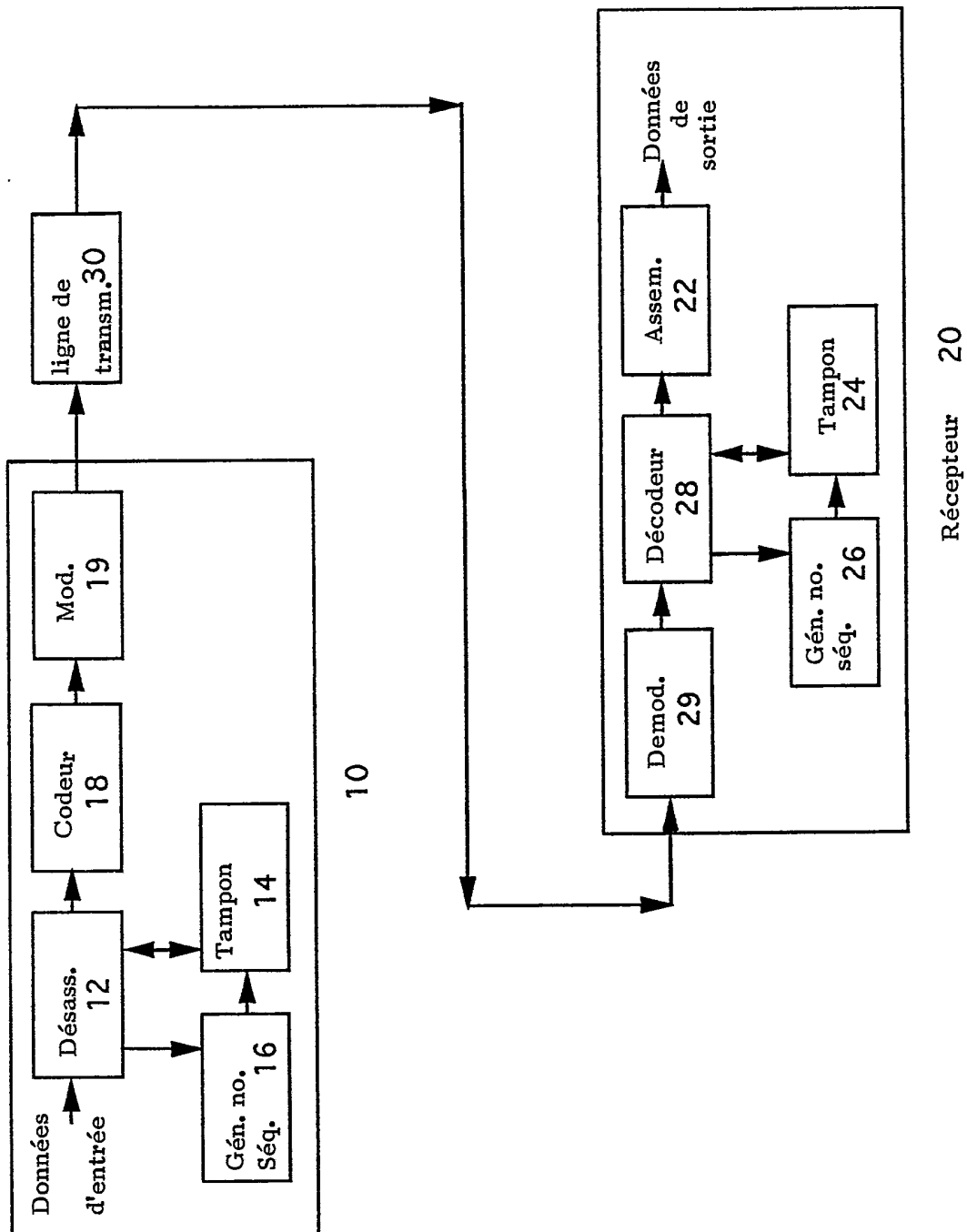


FIG. 1

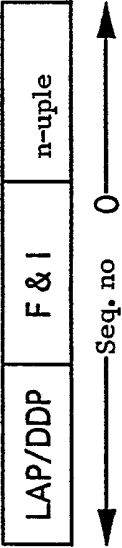


FIG. 2A



FIG. 2B

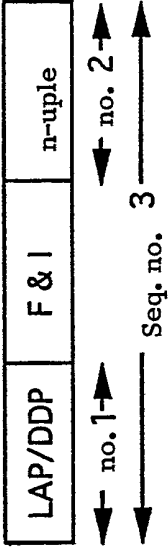


FIG. 3A

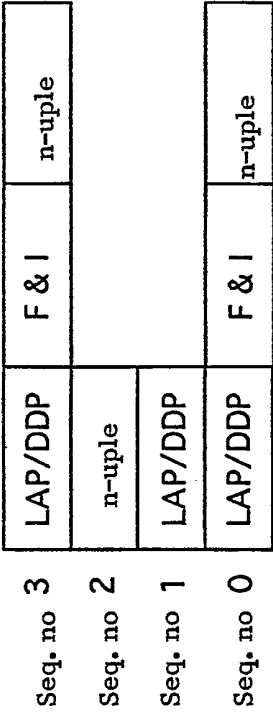


FIG. 3B

Seq. no.	4	LAP/DDP	F & I	n-uple
Seq. no.	3	LAP/DDP	F & I	n-uple
Seq. no.	2	n-uple		
Seq. no.	1	LAP/DDP		
Seq. no.	0	LAP/DDP	F & I	n-uple

FIG. 4B

Seq. no.	4	LAP/DDP	F & I	n-uple
Seq. no.	3	LAP/DDP	F & I	n-uple
Seq. no.	2	n-uple		
Seq. no.	1	LAP/DDP		
Seq. no.	0	LAP/DDP	F & I	

FIG. 5B

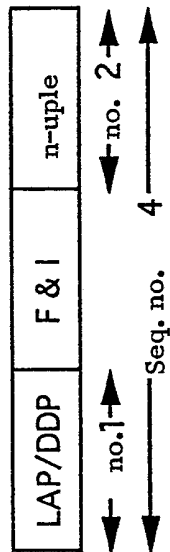


FIG. 4A

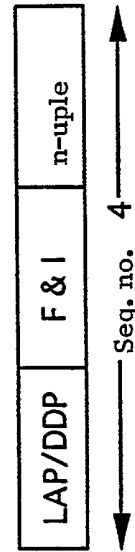


FIG. 5A

DOCUMENTS CONSIDERES COMME PERTINENTS		Revendications concernées de la demande examinée
Catégorie	Citation du document avec indication, en cas de besoin, des parties pertinentes	
X	EP-A-0 224 895 (ATT) * page 3, ligne 13 = page 7, ligne 23 * * page 11, ligne 22 - page 15, ligne 10 * * figures 6-15 * ---	1-4
A	EP-A-0 310 360 (HEWLETT-PACKARD) * page 7, ligne 25 - page 8, ligne 24 * -----	1-4
		DOMAINES TECHNIQUES RECHERCHES (Int.Cl.5)
		H04L
Date d'achèvement de la recherche		Examinateur
3 Novembre 1994		Canosa Arete, C
CATEGORIE DES DOCUMENTS CITES X : particulièrement pertinent à lui seul Y : particulièrement pertinent en combinaison avec un autre document de la même catégorie A : pertinent à l'encontre d'au moins une revendication ou arrière-plan technologique général O : divulgation non-écrite P : document intercalaire T : théorie ou principe à la base de l'invention E : document de brevet bénéficiant d'une date antérieure à la date de dépôt et qui n'a été publié qu'à cette date de dépôt ou qu'à une date postérieure. D : cité dans la demande L : cité pour d'autres raisons Δ : membre de la même famille, document correspondant		

06/95