



# Performances des applications TCP/IP sur GPRS

*Résumé de la présentation effectuée*

**Pierre Eisenmann**

**Nortel Networks**

Algotel 2003

[pierree@nortelnetworks.com](mailto:pierree@nortelnetworks.com)



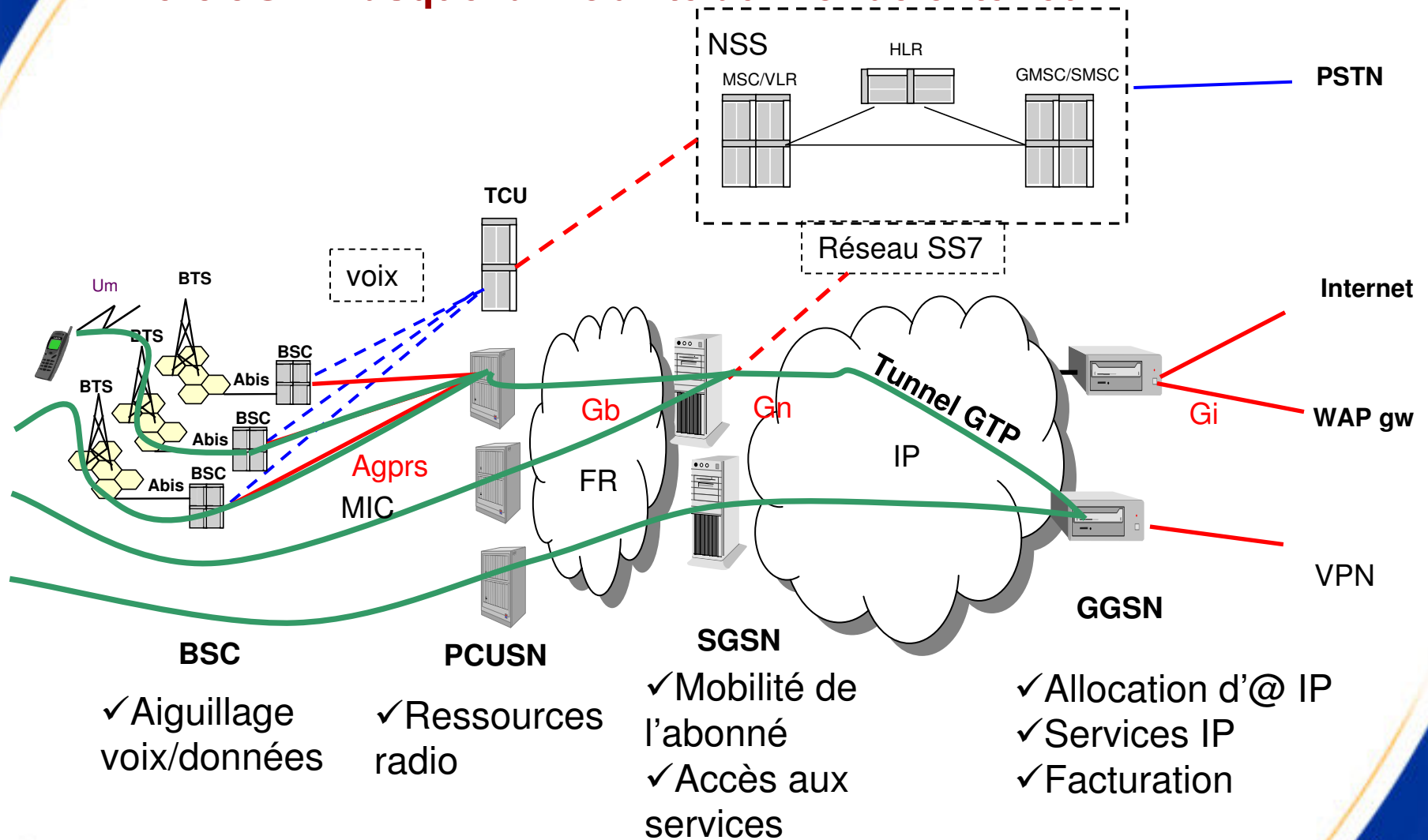


# Introduction

- **Le GPRS est-il un réseau d'accès IP banal?**
- **Un terminal GPRS est-il un modem plug&play?**
  
- **Architecture et Principes du GPRS**
- **GPRS et TCP**
  - GPRS vu par TCP
  - problèmes typiques de TCP sur des réseaux commerciaux
  - Solutions
- **Les applications sur GPRS**
  - Accès à l'intranet de l'entreprise
- **Conclusion**



# Architecture GPRS: le GGSN masque la mobilité au monde extérieur



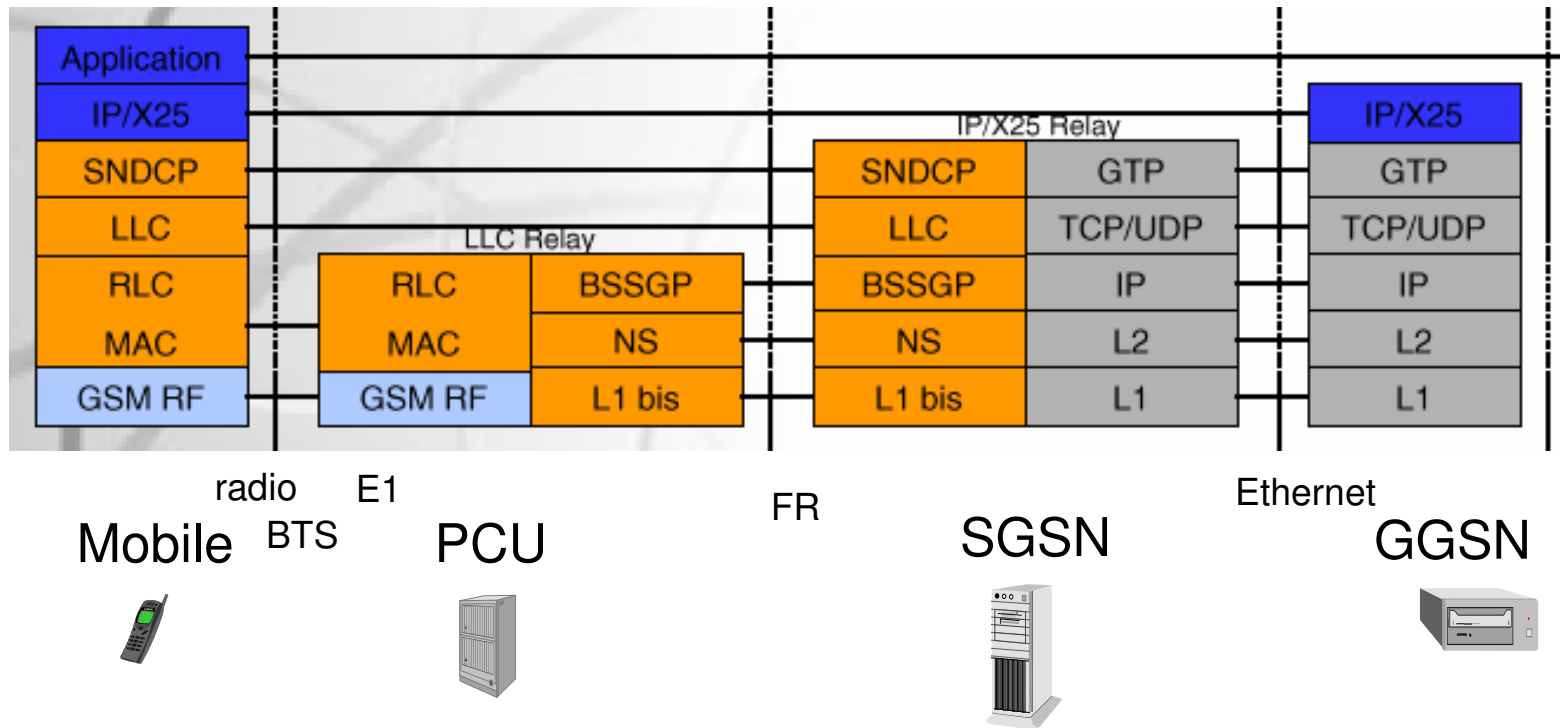
✓ Aiguillage  
voix/données

✓ Ressources  
radio

✓ Mobilité de  
l'abonné  
✓ Accès aux  
services

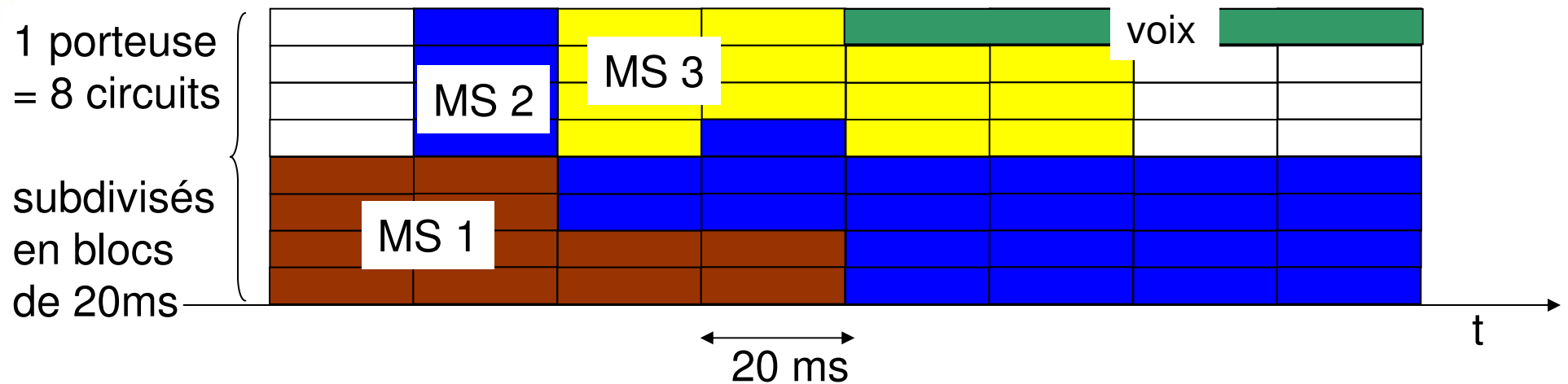
✓ Allocation d'@ IP  
✓ Services IP  
✓ Facturation

# Les couches protocolaires



# Le canal radio: le point névralgique

- Partagé (granularité=20ms), préempté par voix



- **Mode ARQ**
  - ⇒ débit fonction de la qualité du signal: Débit\_max.(1-BLER)
- **Changement de cellule : géré par le mobile (resélection)**
  - ⇒ interruption du canal de 2,5 secondes

# Procédure d'établissement de transfert

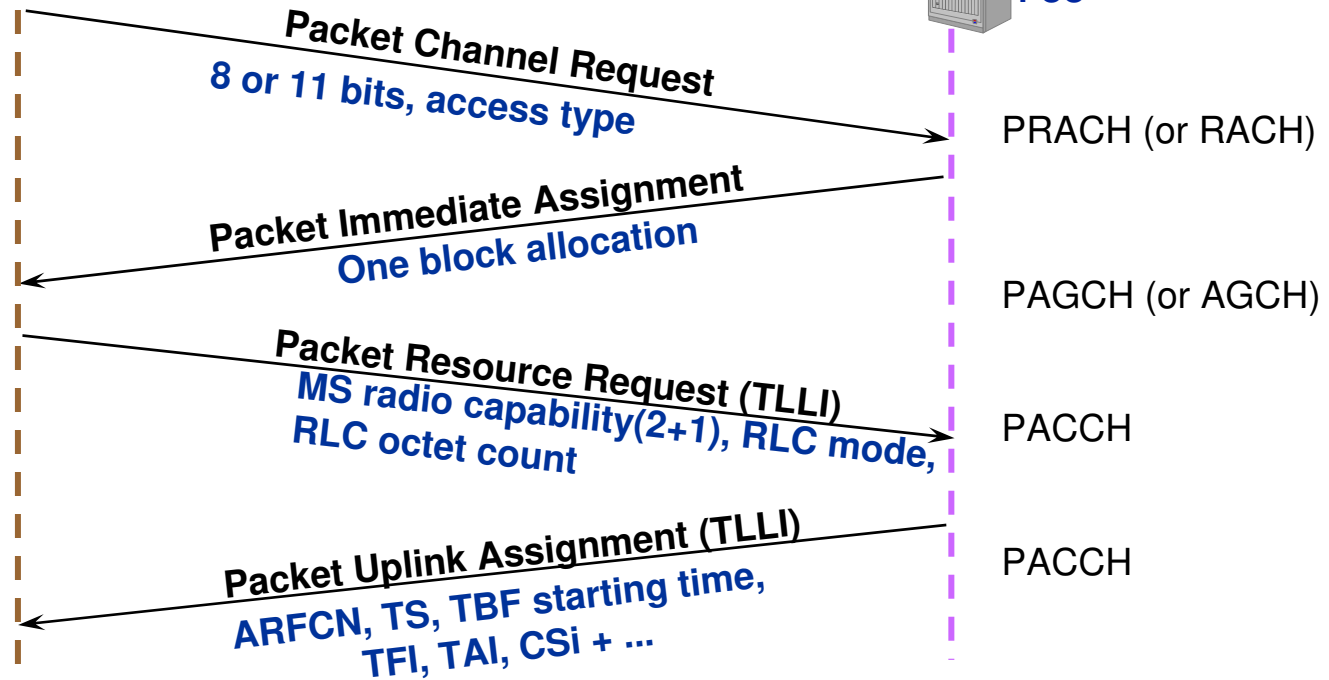
(exemple d'une procédure en 2 phases)  
plusieurs centaines de ms



MS

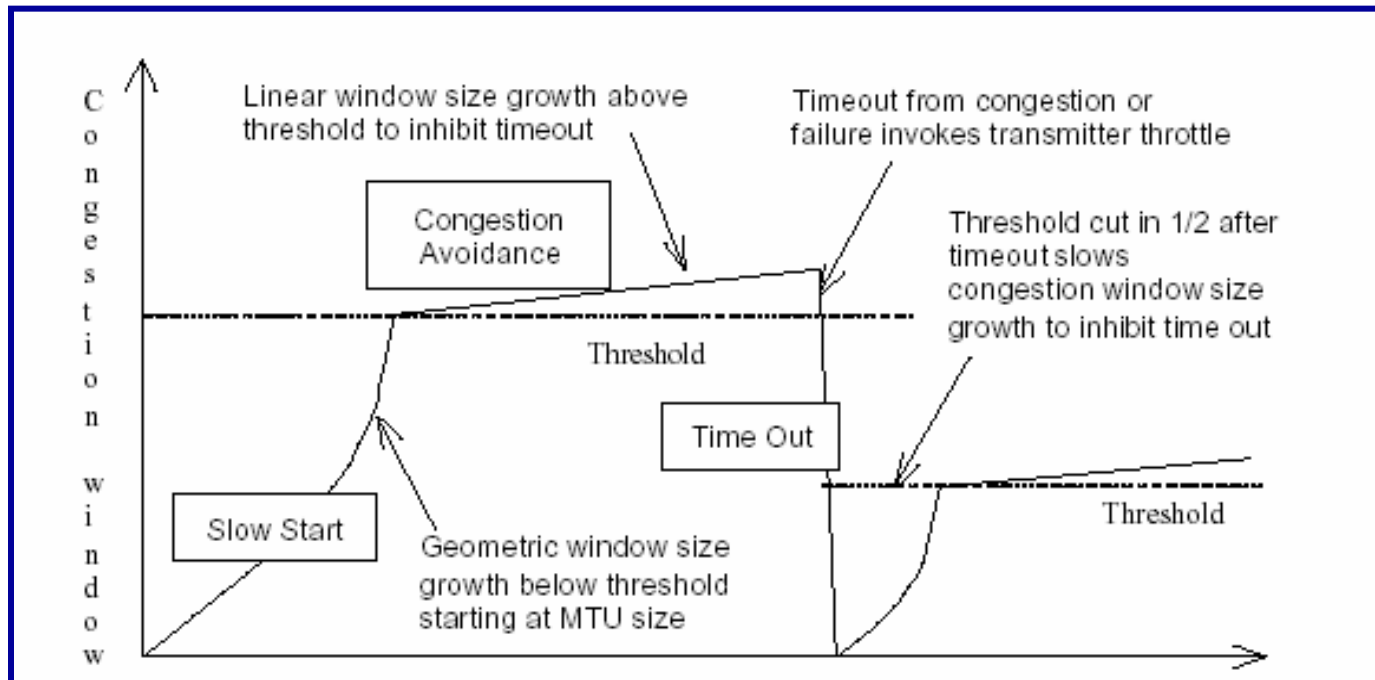


PCU



# TCP: fonctionnement

s'adapte automatiquement à la capacité du canal, du client (RWIN) et à la congestion



- Remplit le canal progressivement
- perte de paquet = signe de congestion
- => TCP adapté à l'Internet mais inadapté au GPRS!

# GPRS vu par TCP

- **Long: temps de latence important et gros buffers**
- **Fin: débit relativement faible**  
RFC 2757: "Long Thin Networks"
- **Fluctuant: débit (qualité radio, trafic)**
- **A pertes (resélections, congestion)**
- **Non-linéaire: oscillation du RTT par regroupement des ACKs**
- **Asymétrique**  
RFC 3481 « TCP over Second (2.5G) and Third (3G) Generation Wireless Networks »

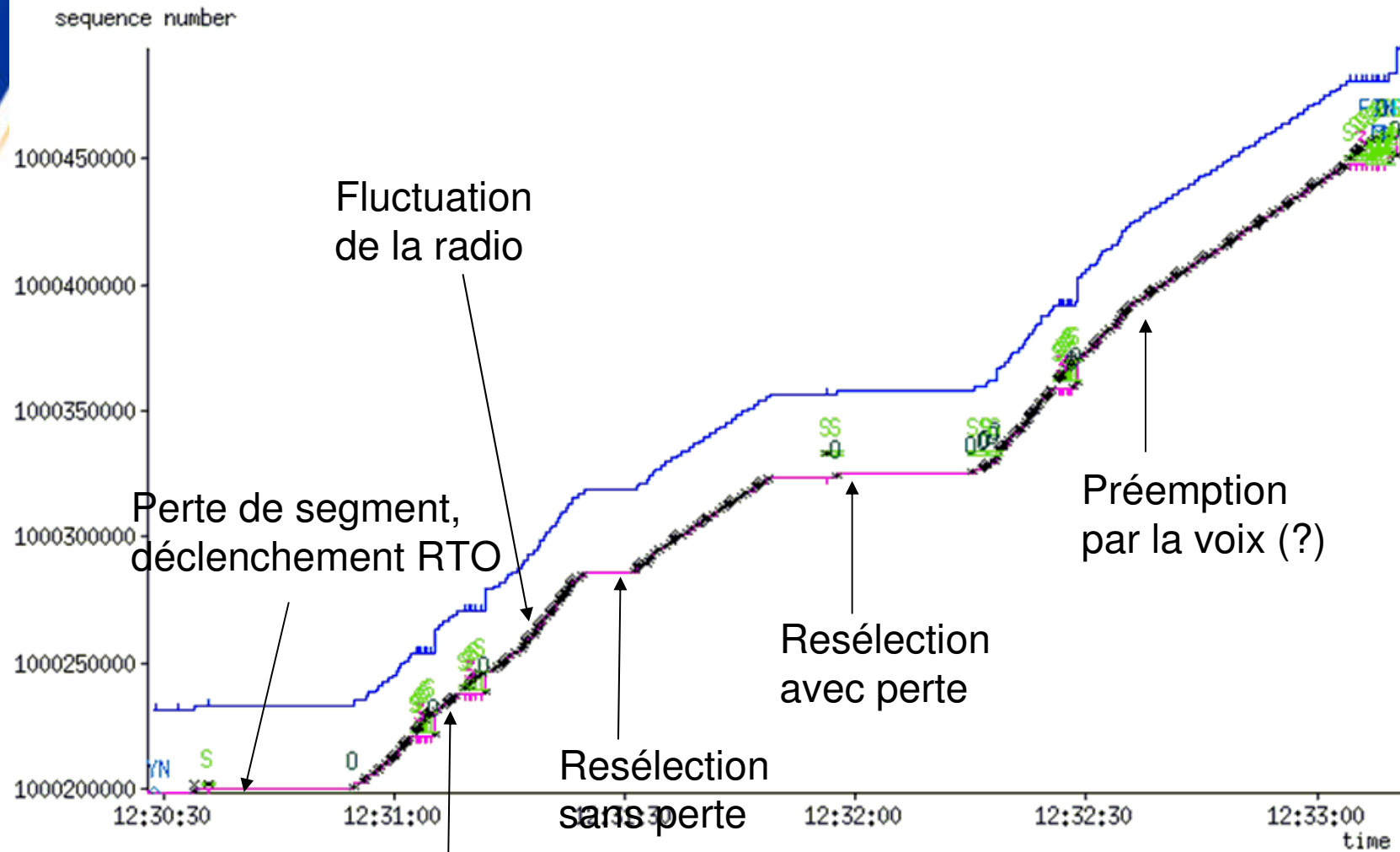
## Mais aussi:

- **Facturé au volume**
- **Ergonomie et usages spécifiques**





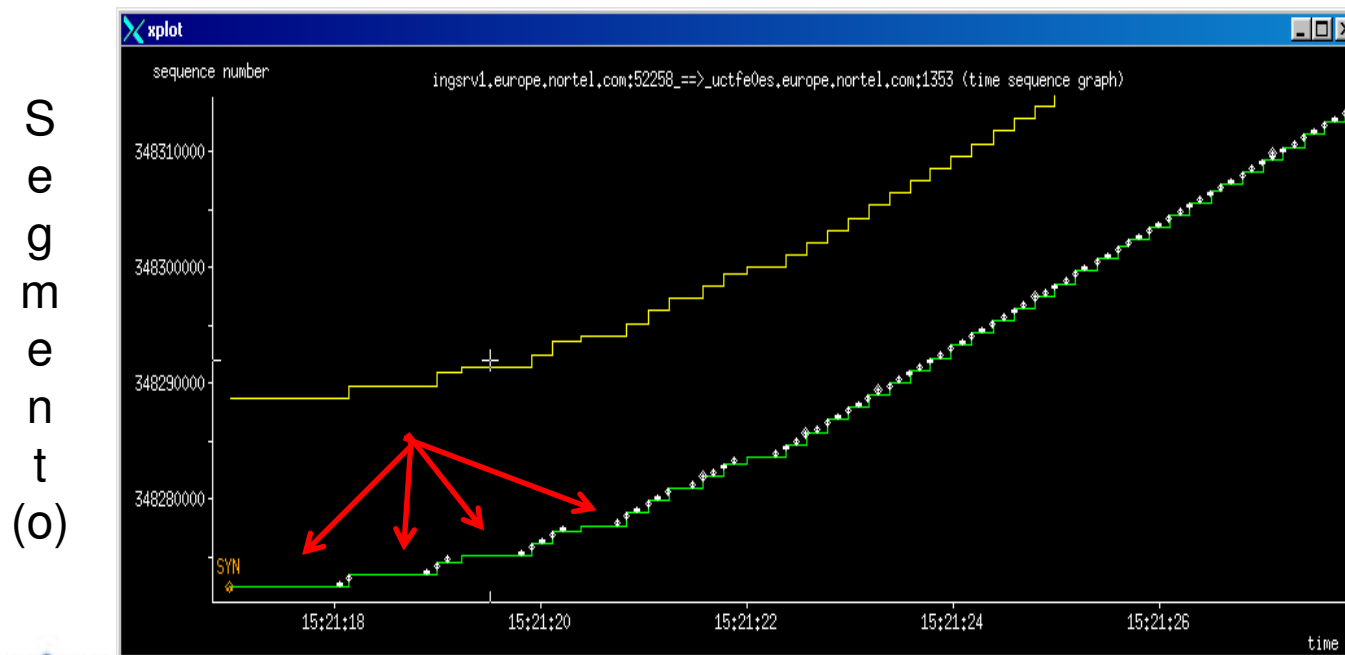
# Trace sur un réseau commercial (2001)



# perte de temps en phase slow-start

- **Slow-Start:**

- Envoi d'un segment, attente d'acquittement, envoi de 2 segments, attente d'acquittements, etc..



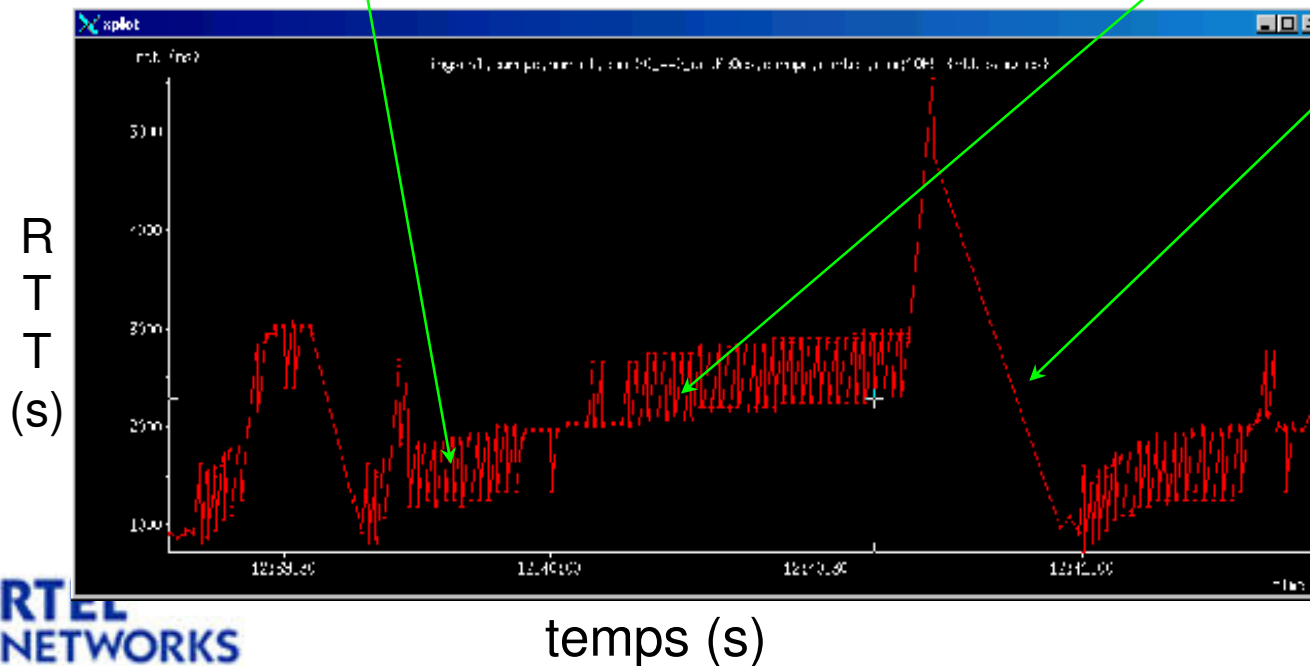
démarrage:  
au total plus  
de 2  
secondes  
perdues

# remplissage du buffer Gb et perte en resélection

- **Visualisation par graphe RTT  
(temps d'aller-retour calculé par TCP)**

Fluctuations court-terme dues  
au temps d'établissement de  
TBF montant

Remplissage progressif  
du buffer (congestion  
avoidance)



Changement de  
cellule: mémoire  
tampon vidée,  
données  
perdus

Redémarrage

## TCP en time-out car RWIN trop petite

- Le time-out de TCP est calculé itérativement

$$RTO_n = \overline{RTT}_n + \alpha \cdot \text{écart\_type}(RTT)_n$$

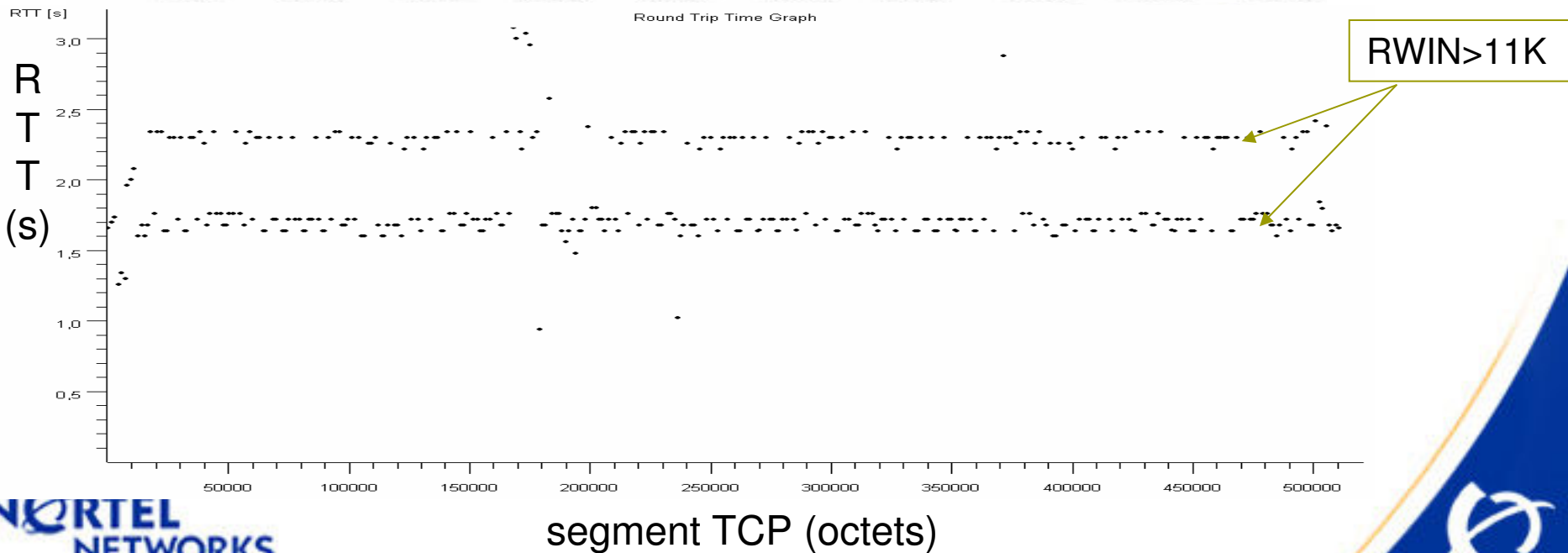
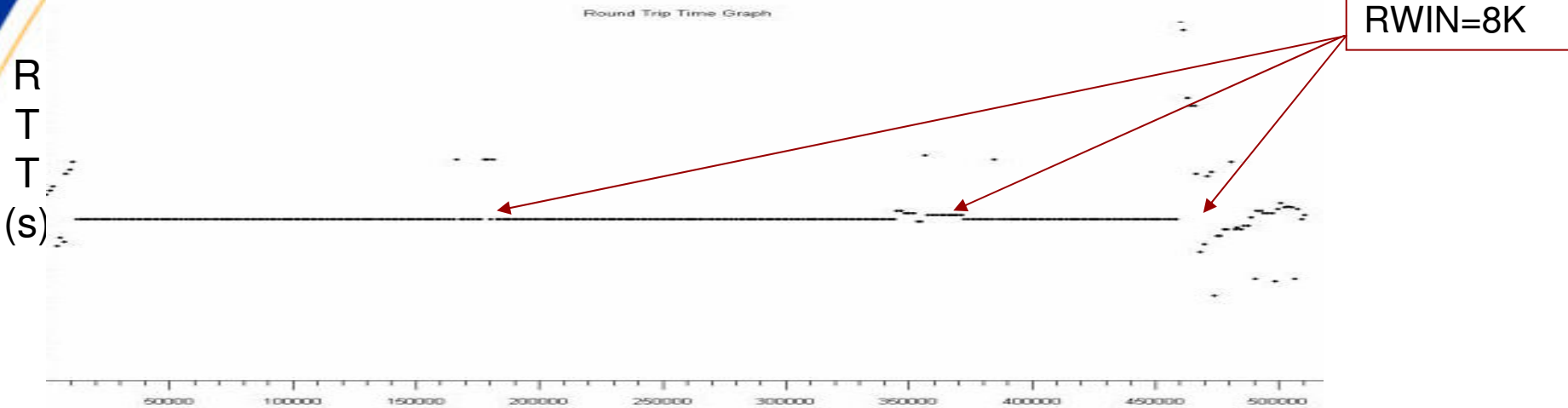
- RTT constant => TCP en time-out dès que RTT fluctuera

$$RTO_n \approx RTT$$

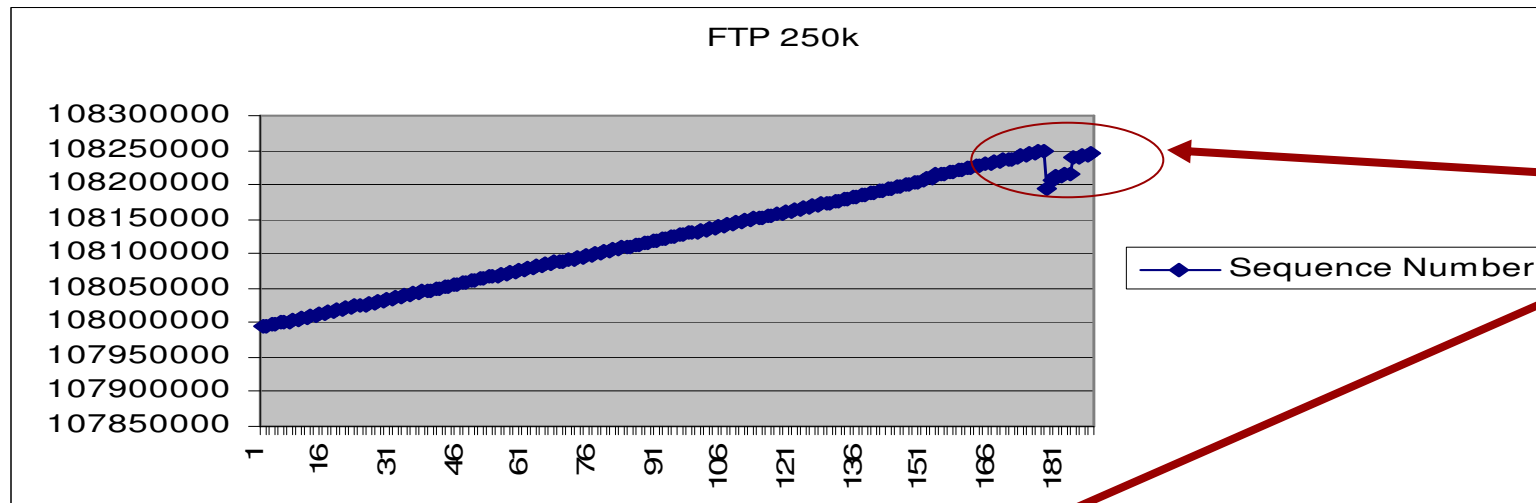
- **RWIN de 8K (par défaut dans W95):**
  - tous les ACKS dans le même TBF=> RTT constant
- **RWIN > 8K:**
  - ACKs sur plusieurs TBF => fluctuation de RTT => meilleure robustesse



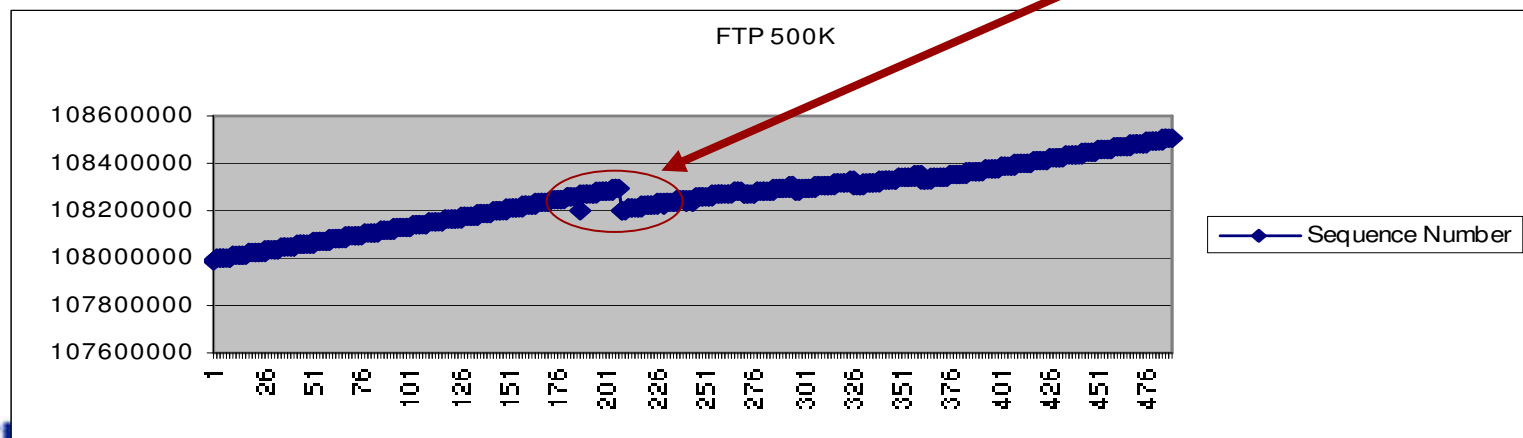
# RTT avec RWIN=8K et 11K : robustesse par ajout de fluctuations



# adaptation de débit imparfaite sur Gb : perte de segment en phase slow-start (RWIN=65K, envoi trop brutal de données sur le SGSN)



perte  
de  
trame



# Un bug résiduel de Mobile induit une retransmission

- Pendant la fermeture du TBF montant, le mobile perd les ACK TCP venant de la couche PPP. Le transfert est retardé de 2,6 s.

Timestamp	Source Address	Destination Address	Protocol	Message	Application	Syn	Ack	Fin	Sequence Number	Acknowledgement Number	Total Length
15:32:16 766'	10.4.128.10	10.224.64.6	TCP	226 Transfer complete - file 500KTEST sent successfully	FTP				1015496489	3440603	97
15:32:19 441'	10.4.128.10	10.224.64.6	TCP	226 Transfer complete - file 500KTEST sent successfully	FTP				1015496489	3440603	97
15:32:19 767'	10.4.128.10	10.224.64.6	TCP		FTP-DATA				1016008124	3495218	40
15:35:07 526'	10.4.128.10	10.224.64.6	TCP	221 Goodbye!	FTP				1015496546	3440609	54
15:35:07 532'	10.4.128.10	10.224.64.6	TCP	526'	FTP	1	1	1	1015496560	3440609	40
15:35:08 448'	10.4.128.10	10.224.64.6	TCP		FTP				1015496561	3440610	40



## Case 1: customer observations

- **DL throughput with 4+1 MS is below Operator KPI target**
  - DL\_101 (HTTP 2Mo) = **35.9** kbps
  - **expectation = 42.5 kbps**, corresponding to max observed end to end throughput minus 10%
  - Case opened: Weak DL throughput and DL breaks
- **Test Environment**
  - GPRS Access V12.4c / Core GPRS4.0 (SGSN patched / GGSN 2.0.3)
  - TCP settings Win98 default settings: RWIN 8Ko, MTU=1500 bytes
  - MS 4+1 brand #1 and #2
  - Linux Server (best-in class TCP stack, RWIN: 32 kByte)





# Case 1: Problem Analysis/Solution

## Issue 1: bugs/performances of two mobiles

- **Mobile #1 issue:**
  - MS#1 mis-interprets incoming RLC message “Packet TBF release” from PCU and releases both UL and DL channels, whereas it should only release the DL channel
  - Impact: PCU loss of com in case of stalled TCP window – takes 5s
- **Mobile #1 Solution: Use more recent software version**
- **Mobile # 2 issue**
  - MS#2 seems to experience packet loss when it has to handle many LLC frames or to re-assemble LLC. Nortel SGSN initial parameter setting on site was fragmenting LLC frames
  - Impact: Loss of TCP packets
- **Mobile # 2 Solution: Change Nortel SGSN configuration to disable LLC fragmentation by setting MaxUinfoFieldLength field to 1520 bytes. IP frames lost at PPP layer almost fully solved.**



# Case 1: Problem Analysis/Solution

## Issue 3: GGSN issue?

- **Problem**

- During fragmentation at Gn side due to MTU=1500, it was stated that the GGSN was losing IP fragments.

	Source	Destination	Protocol	src port	dst port	Size	Info
38:24.7979	62.180.105.41	10.41.0.12	GTP <HTTP>	80	1033	1514	Continuation
38:24.7979	62.180.77.3	62.180.77.17	IP			82	Fragmented IP protocol (proto=UDP 0x11, off=1480)
38:24.8679	10.41.0.12	62.180.105.41	GTP <TCP>	1033	80	102	1033 > 80 [ACK] Seq=13991372 Ack=487710289 Win=8760 Len=0
38:24.8779	62.180.105.41	10.41.0.12	GTP <HTTP>	80	1033	1514	Continuation
38:24.8779	62.180.77.3	62.180.77.17	IP			82	Fragmented IP protocol (proto=UDP 0x11, off=1480)
38:25.5279	10.41.0.12	62.180.105.41	GTP <TCP>	1033	80	102	1033 > 80 [ACK] Seq=13991372 Ack=487711749 Win=8760 Len=0
38:25.5379	62.180.77.3	62.180.77.17	IP			82	Fragmented IP protocol (proto=UDP 0x11, off=1480)
38:25.6079	10.41.0.12	62.180.105.41	GTP <TCP>	1033	80	102	1033 > 80 [ACK] Seq=13991372 Ack=487713209 Win=8760 Len=0
38:25.6179	62.180.105.41	10.41.0.12	GTP <HTTP>	80	1033	1514	Continuation
38:25.6179	62.180.77.3	62.180.77.17	IP			82	Fragmented IP protocol (proto=UDP 0x11, off=1480)
38:26.2879	10.41.0.12	62.180.105.41	GTP <TCP>	1033	80	102	1033 > 80 [ACK] Seq=13991372 Ack=487714669 Win=8760 Len=0
38:26.2979	62.180.105.41	10.41.0.12	GTP <HTTP>	80	1033	1514	Continuation

- **Solution:**

- It was a test artefact due to an issue on a Gn Ethernet switch.

# Case 1: Problem Analysis/Solution

## Final MS 4+1 DL throughput tests

MTU 1500		
	<i>Results (bits/s)</i>	
<b>RWIN</b>	<b>COM</b>	<b>APPL</b>
8k	40936	38892
16k	43545	42246
32k	43037	41820



# Inefficacité d'un serveur Web

transfert type d'une petite page en labo (1ko body+1ko image)

10: requête de la page

11-12: réponse du serveur par http 200 OK et TCP attend l'ACK

14-15: le client envoi TCP-ACK et le « get » pour l'image

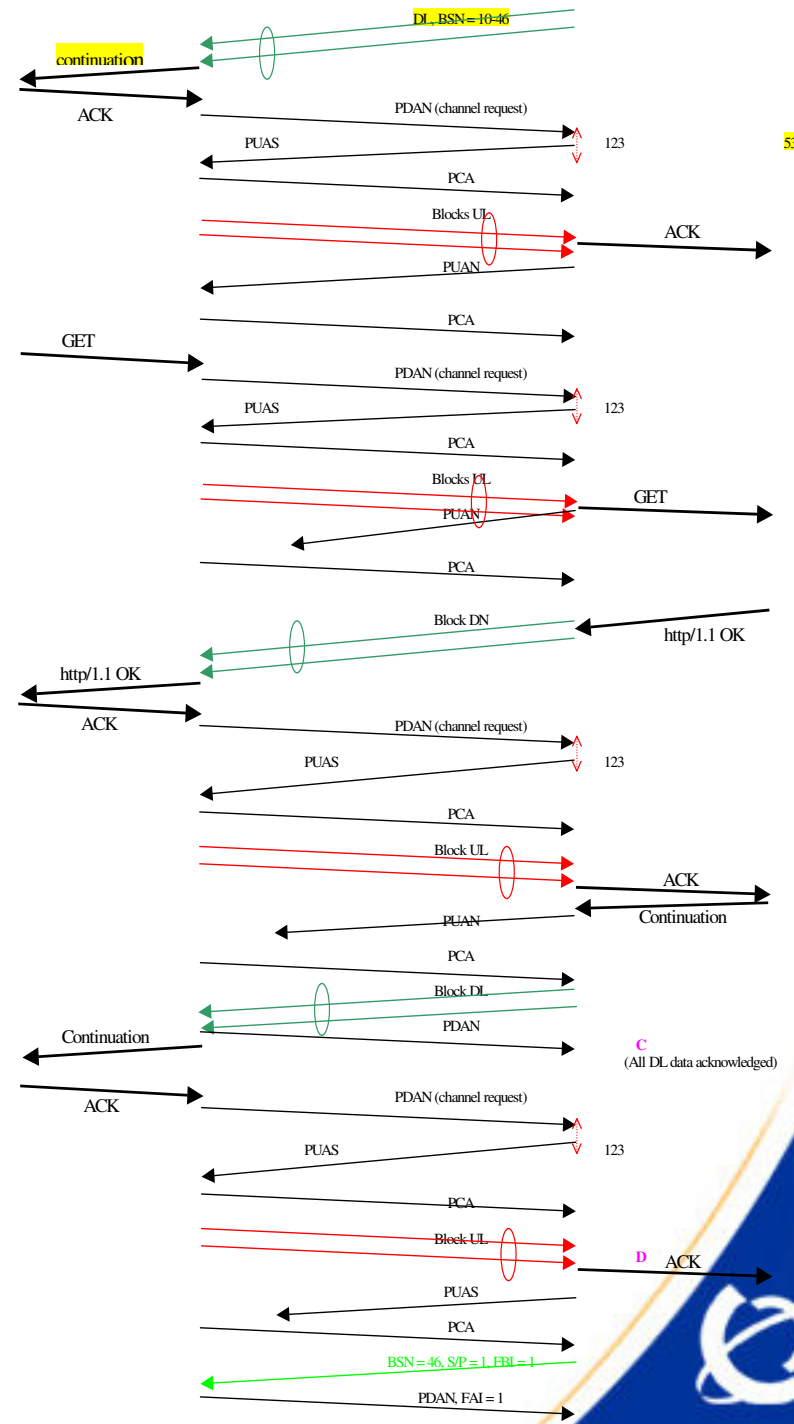
16-17: réponse serveur par http 200 OK et TCP attend l'ACK

No.	Time diff.	Source	Destination	length		
10	4.741979	45.21.23.24	121.45.21.56	275 [PSH, ACK] <i>get page.html</i>	Ack=0	Win=4096
11	0.057943	121.45.21.56	45.21.23.24	242 [PSH, ACK] <i>http 200 OK</i>	Seq=3418690	Ack=7490043
<b>12</b>	<b>0.978829</b>	<b>45.21.23.24</b>	<b>121.45.21.56</b>	<b>60 [ACK]</b>	Ack=3418690	Win=4096
13	0.000561	121.45.21.56	45.21.23.24	1078 [PSH, ACK] <i>page.html</i>	Seq=7490043	Ack=3418690
<b>14</b>	<b>1.099894</b>	<b>45.21.23.24</b>	<b>121.45.21.56</b>	<b>60 [ACK]</b>	Ack=7490265	Win=64240
15	0.181560	45.21.23.24	121.45.21.56	281 [PSH, ACK] <i>get image.gif</i>	Seq=3418690	Ack=7490265
16	0.067414	121.45.21.56	45.21.23.24	242 [PSH, ACK] <i>http 200 OK</i>	Ack=3418690	Win=4096
<b>17</b>	<b>1.191343</b>	<b>45.21.23.24</b>	<b>121.45.21.56</b>	<b>60 [ACK]</b>	Seq=3418690	Ack=7490265
18	0.000482	121.45.21.56	45.21.23.24	1078 [PSH, ACK] <i>image.gif</i>	Ack=3418690	Win=4096
19	1.059304	45.21.23.24	121.45.21.56	60 [ACK]	Seq=7490713	Seq=7490265

⇒ Perte cumulée de temps (12+17) = 2 secs.

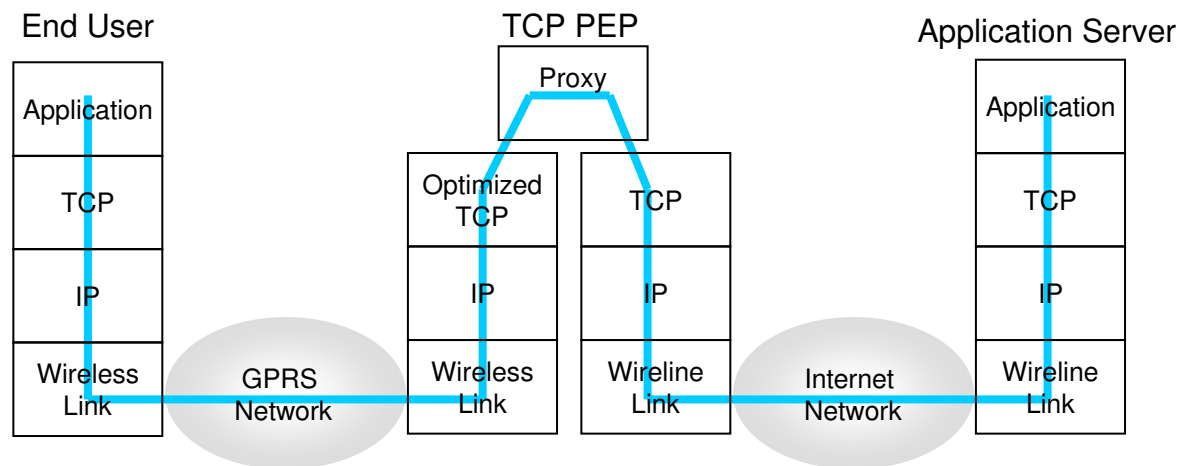
# NB: analyse radio RLC/MAC

- **Objectif:**
  - Voir si la radio est optimisée pour l'application
- **Conclusions:**
  - Les paramètres sont OK
- **améliorations:**
  - Les fonctionnalités à venir améliorant la réactivité sont évaluées et optimisées



## Utilité d'un Performance Enhancing Proxy

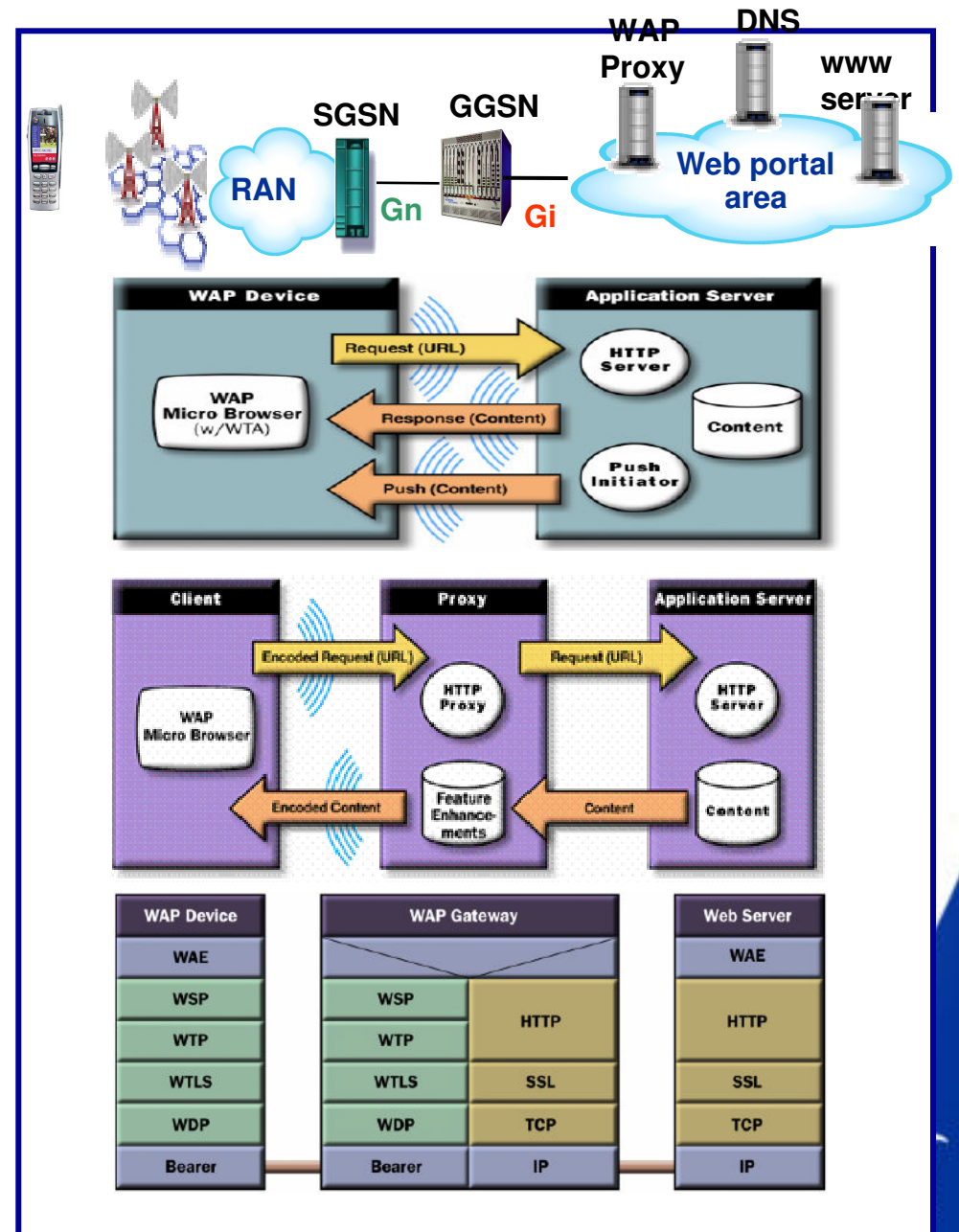
- La RFC 3135 définit le PEP et recommande son usage typiquement pour des WWAN (GPRS, Satellites,..)



- PEP: avec clients, sans clients, sens montant / descendant, etc..

# cas du WAP: PEP dans la gateway, système fermé

- Standard naming model – WWW-standard URLs are used to identify WAP content on origin servers.
- Content typing and format– All WAP content is given a specific type and format consistent with WWW.
- Standard communication protocols – WAP communication protocols enable the communication of browser requests from the mobile terminal to the network web server.
- WAP PROXY or GATEWAY provides
  - HTTP/WSP methods
  - Cookies management
  - Content encoder/decoder
  - PPG fonctionnalité (Push Proxy Gateway)
  - PUSH services as PushPAP and PushOTA (OTA-WSP or OTA-HTTP)
  - Over The Air (OTA) security (WTLS, WIM, PKI)
  - UAProf (user agent profile) negotiation
  - Reply Charging
  - And WTA capabilities

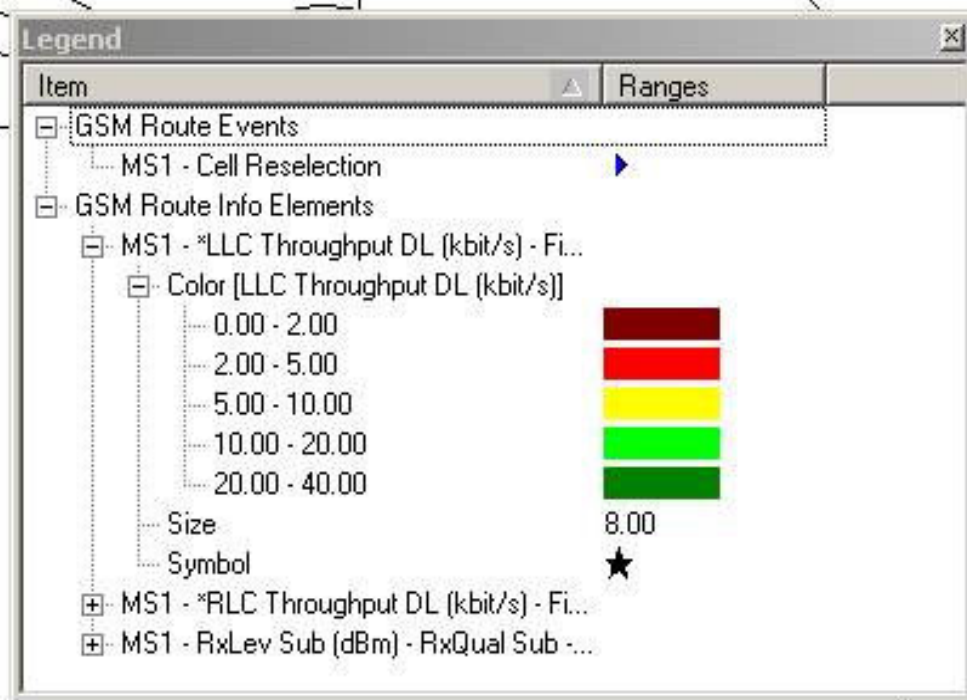


*end*

## 2MB download while driving in large city

(about 19 mins, 3+1 MS, live network work hours, no PEP used)

*start*



- ✓ reselections (▶) induce significant mutes (■)
- ✓ irregular throughput





# Les couches applicatives

- **Les applications peuvent aussi être optimisées**
- **HTTP**
- **accès à l'entreprise**
  - Le tunnel IP
  - Le trafic sur l'intranet
  - Outlook: une application d'entreprise





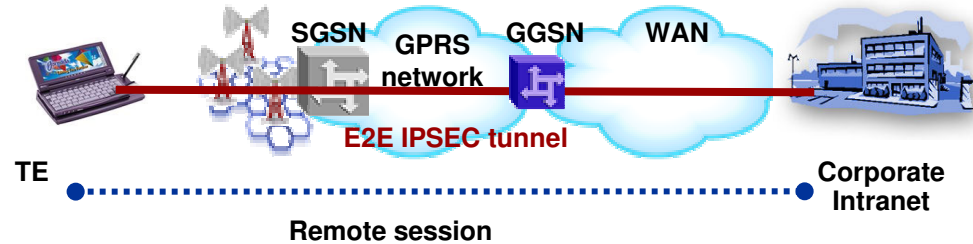
# http

- **Pour un remplissage optimal du tuyau, http 1.1 est indispensable**
  - Persistent connection: ne pas relancer de connection TCP à chaque image
  - Pipelining: faire les GET en // sans attendre la fin de la réception précédente et dans la même connection
  - Connections en parallèle

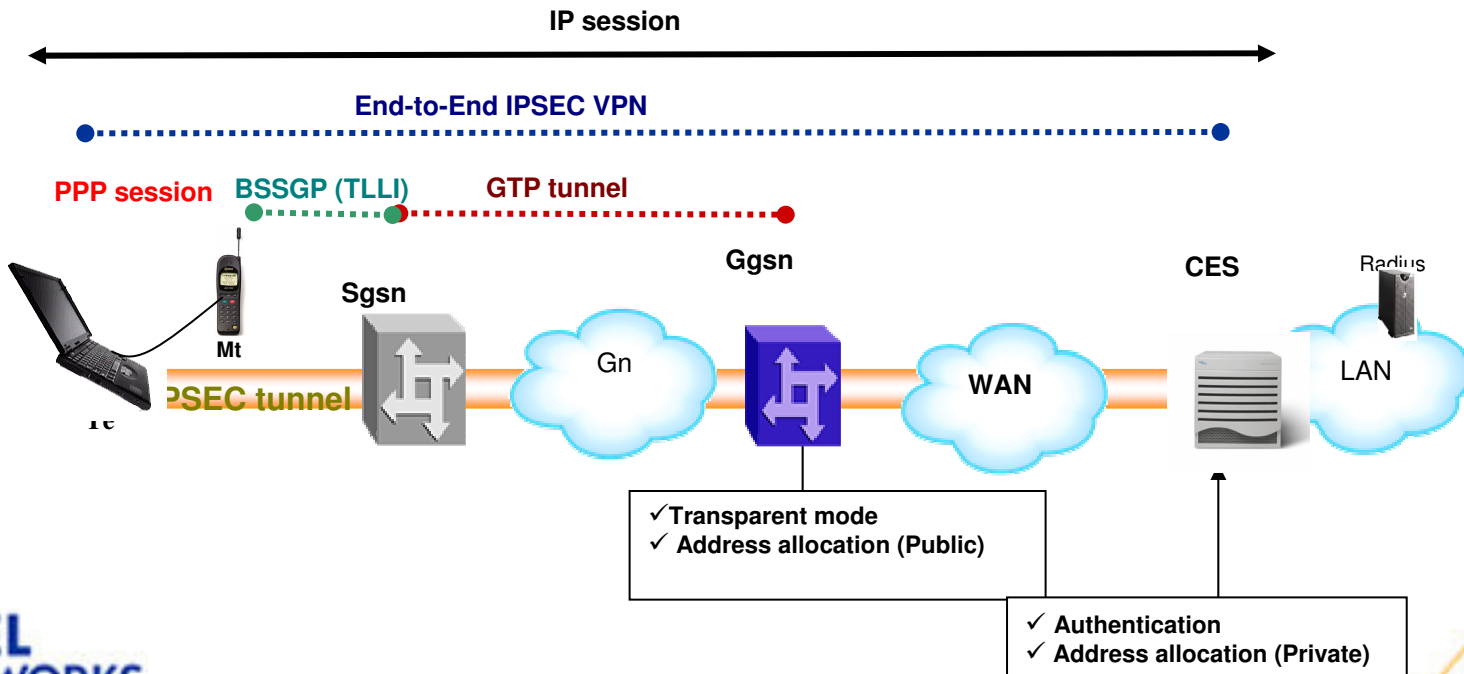


# VPN & GPRS: tunnel IPSEC

e2e Architecture:



VPN & GPRS interaction :



# IPSEC apporte de la compression

Bonne couverture radio

data rate (bits/sec) average (std dev)	Compressible File				Non Compressible File			
	120 kbyte		450 kbyte		120 kbyte		450 kbyte	
no VPN	42 608	(0)	43 921	(298)	42 958	(1 038)	43 840	(443)
VPN to lab CES	117 066	(7 458)	132 888	(4 925)	41 120	(964)	42 600	(0)
VPN to Corporate CES	80 584	(19 274)	84 621	(14 993)	39 954	(789)	39 847	(1 298)

download time (sec)	Compressible File		Non Compressible File	
	120 kbyte	450 kbyte	120 kbyte	450 kbyte
no VPN	23	82	22	82
VPN to lab CES	8	27	23	85
VPN to Corporate CES	12	43	24	90

Déplacement en bordure de cellule (fichier compressible)

attenuation (dB)	throughput (bits/sec)
no BLER	132 888
-5	132 888
-9	132 888
-16	128 000
-19	128 000
-21	102 400
-23	96 000
-29	64 000

Mobile 4+1, LZS activé

Débit effectif de 130kbits/s



# MS Outlook sur GPRS

## actions types et coûts correspondants

Outlook Test Description	Srv->Cli (Bytes)	Cli->Srv (Bytes)	Srv->Cli (Frames)	Cli->Srv (Frames)
Send Item 100K	3218.000	41326.000	20.000	34.000
Open Item 100K	44857.333	3967.333	33.333	25.667
Send Item 100K No Notification	2498.000	41342.000	20.000	34.000
Send Item 100K with Attachment	5326.000	113279.500	49.500	85.000
Open Item 100K with Attachment	6404.000	1328.667	6.000	6.000
Read Attachment 100K	109700.000	5060.000	80.333	57.000
Ambiguous Address Resolution (4 char)	764.000	520.000	2.000	3.000
Address Lookup	19518.667	1125.333	14.667	11.333
Open Calendar	914.000	958.000	3.000	4.000
Add a Calendar Item	1229.000	1545.000	4.000	5.000
Open Tasks	358.000	354.000	1.000	2.000
Open Contacts	406.000	386.000	1.000	2.000
Add a Contact Item	1023.000	2103.000	4.500	7.000

- Outlook compresses text message, not attached files (60% gain).
- Task, calendar, contact are always synchronizing (increasing user expense).
- An attached file can be sent twice.

- Outlook is not error recoverable (user may be double billed at network failure).
- Actions from line 7 in the table above increase user expense

# Conclusion

- **L'optimisation fine du GPRS apporte beaucoup**
- **TCP doit être optimisé:**
  - usage systématique d'un PEP TCP
  - Mais il dépend de la configuration (encryption, Internet, équipement réseau, client,..)
- **Les couches applicatives doivent être soignées:**
  - Éviter « single thread »
  - Minimiser le bavardage
  - Comprimer les données
- **L'infrastructure aussi**
- **L'optimisation pour un usage Plug&Play du GPRS doit être faite. Elle est compliquée par des facteurs techniques et économiques (compatibilité avec l'existant).**
- **L'utilisation de la connaissance détaillée du GPRS de bout en bout est essentielle.**





# NORTEL NETWORKS

NORTEL NETWORKS CONFIDENTIAL

