IPv6 Networking Over Satellite

For Mobile User Groups

TriaGnoSys GmbH

Axel Jahn, Angels Via

About TriaGnoSys GmbH

TriaGnoSys develops communication services for mobile networks, our advanced products are based on innovative research

Satellite Communications

Aeronautical Communications

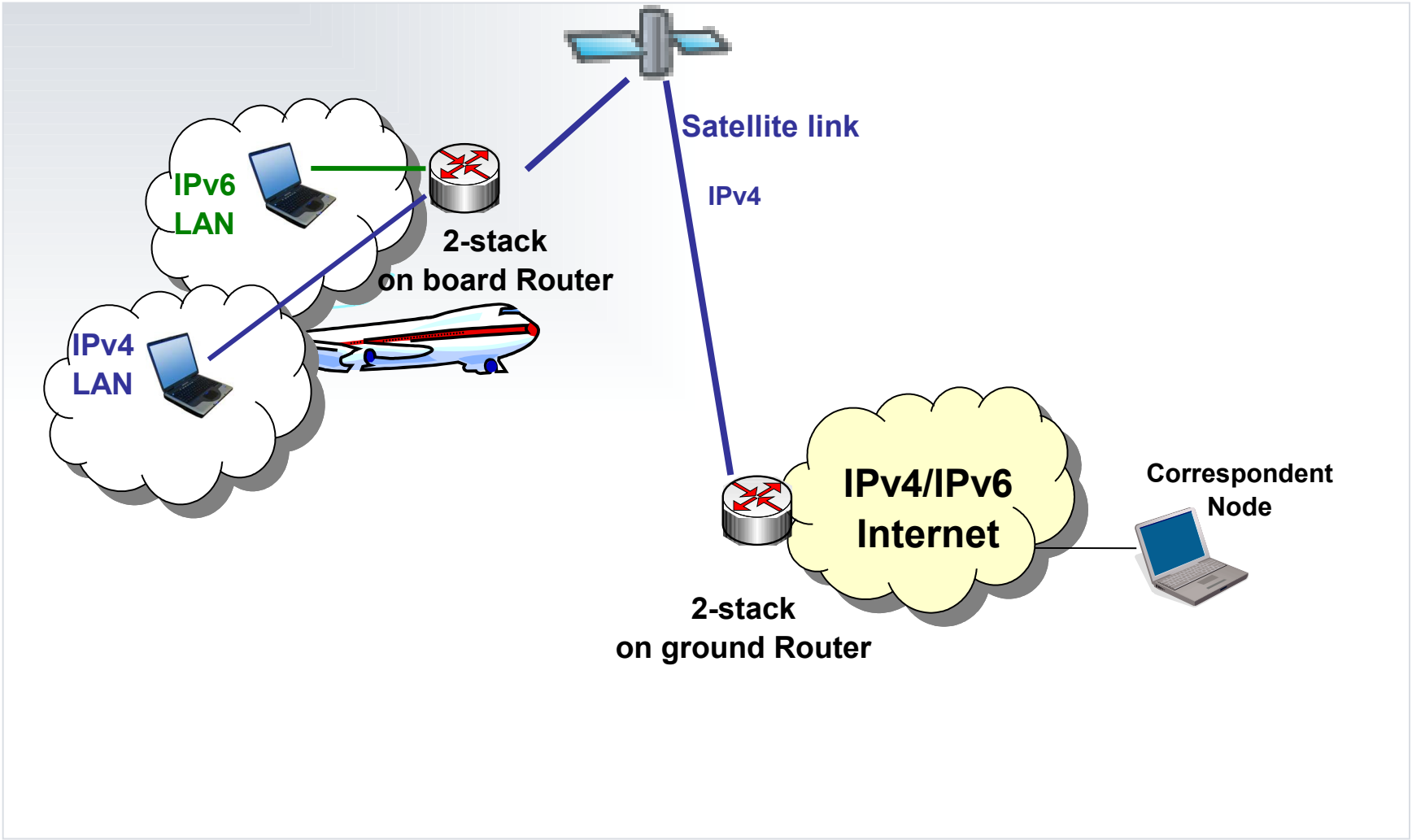
Maritime Communications

Navigation

ISO 9001 certified

Index

Scenario



Problem Statement

Satellite link only supports IPv4

Multiple IPv6 users inside the A/C

- **Packets must be sent through this link**

Aircraft moves

- **Mobility Problem**
  - **Every time the A/C changes AP, connection breakdown**
  - **Not addressed in this paper**

Possible Networking Solutions

Manual Tunnelling

- Overhead due to additional header
- Simpler

NAPT-PT modified

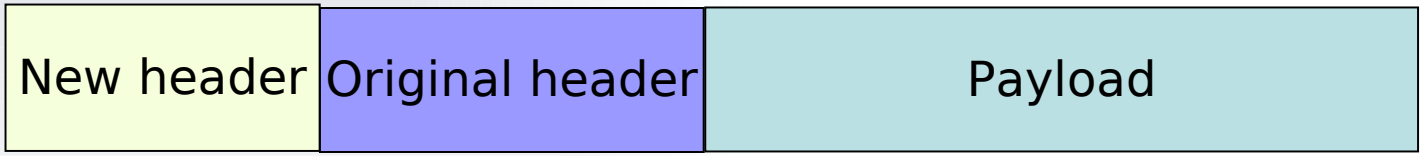
- Less bytes sent
- More complex at both ends of the Satellite Link

Automatic Tunnels

- Not addressed here

Tunnelling

Adding a new header



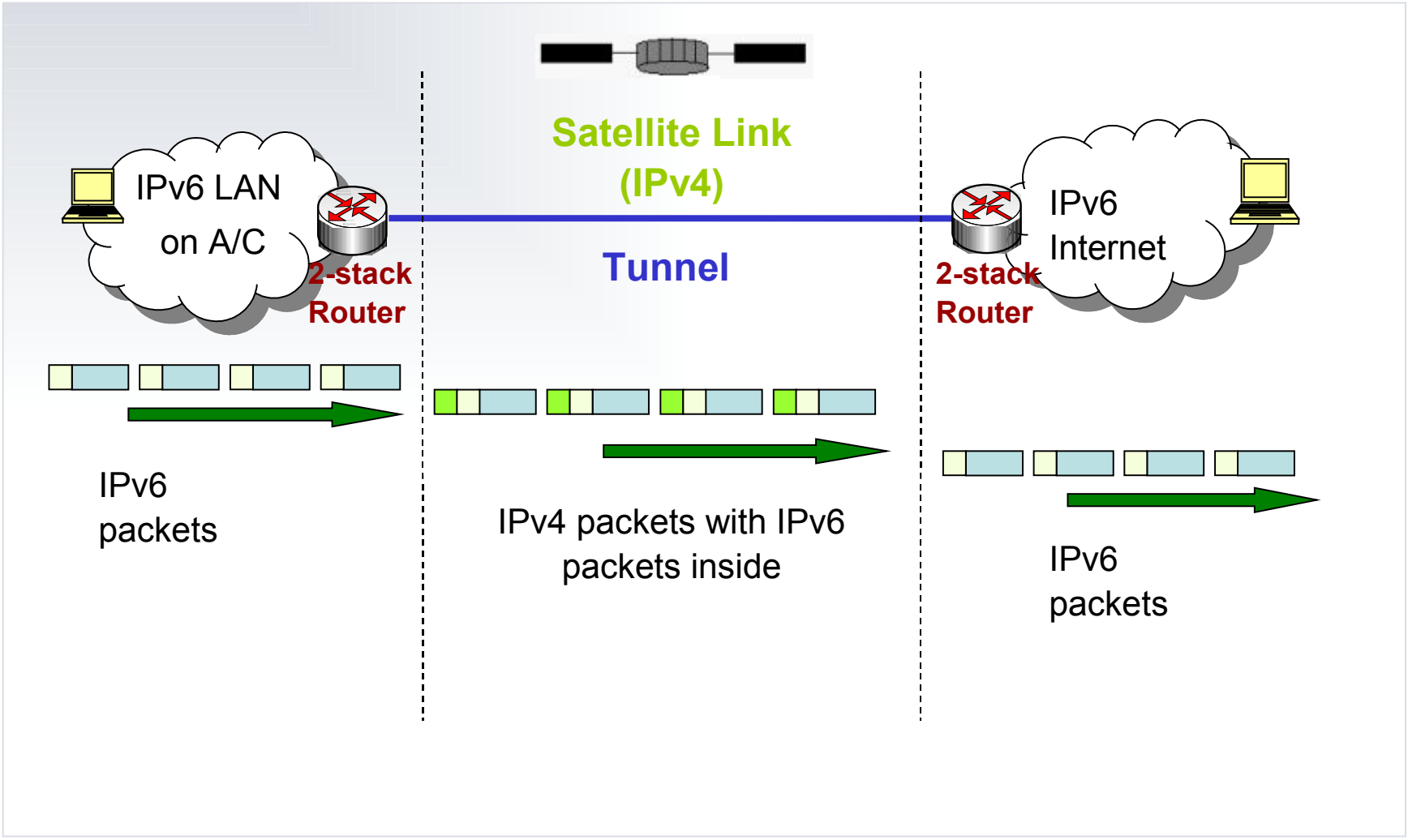
Advantages:

- **Simplicity**
- **Already implemented**

Disadvantages:

- **Overhead**

Manual Tunneling



Manual Tunneling

Both ends of tunnel are preconfigured

Encapsulation:

- **Adding IPv4 header with protocol type 41**
- **Fragmentation if MTU is smaller than 1280 Bytes**

Decapsulation:

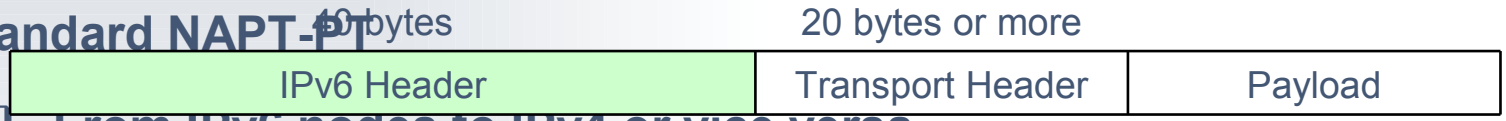
- **Security: accept packets only from configured tunnels**
- **Reassembling all fragments**

Translation of IPv4 error messages to IPv6

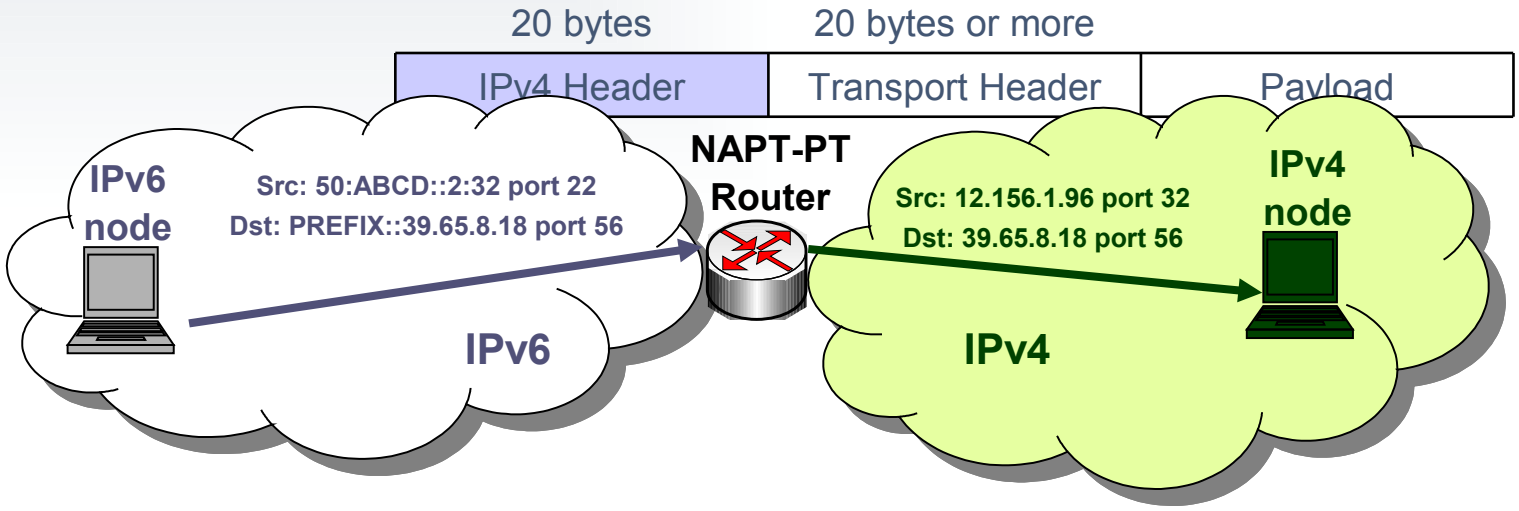
NAPT-PT

Protocol Translation (Header Translation)

Standard NAPT-PT



- From IPv6 nodes to IPv4 or vice versa

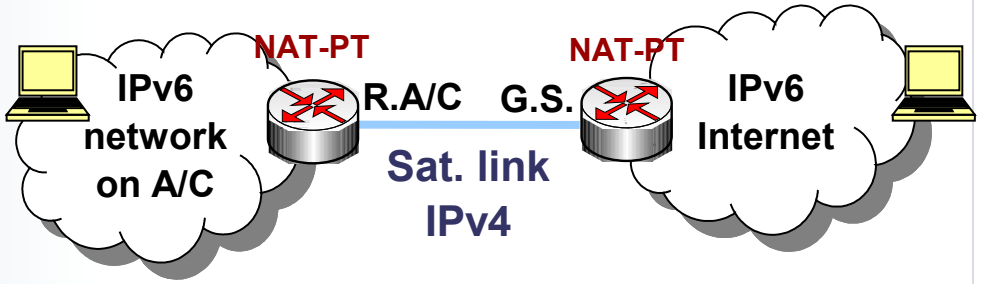


Some modifications needed

NAPT-PT modified

2 Translations:

- Router in A/C
- Router in Ground Station

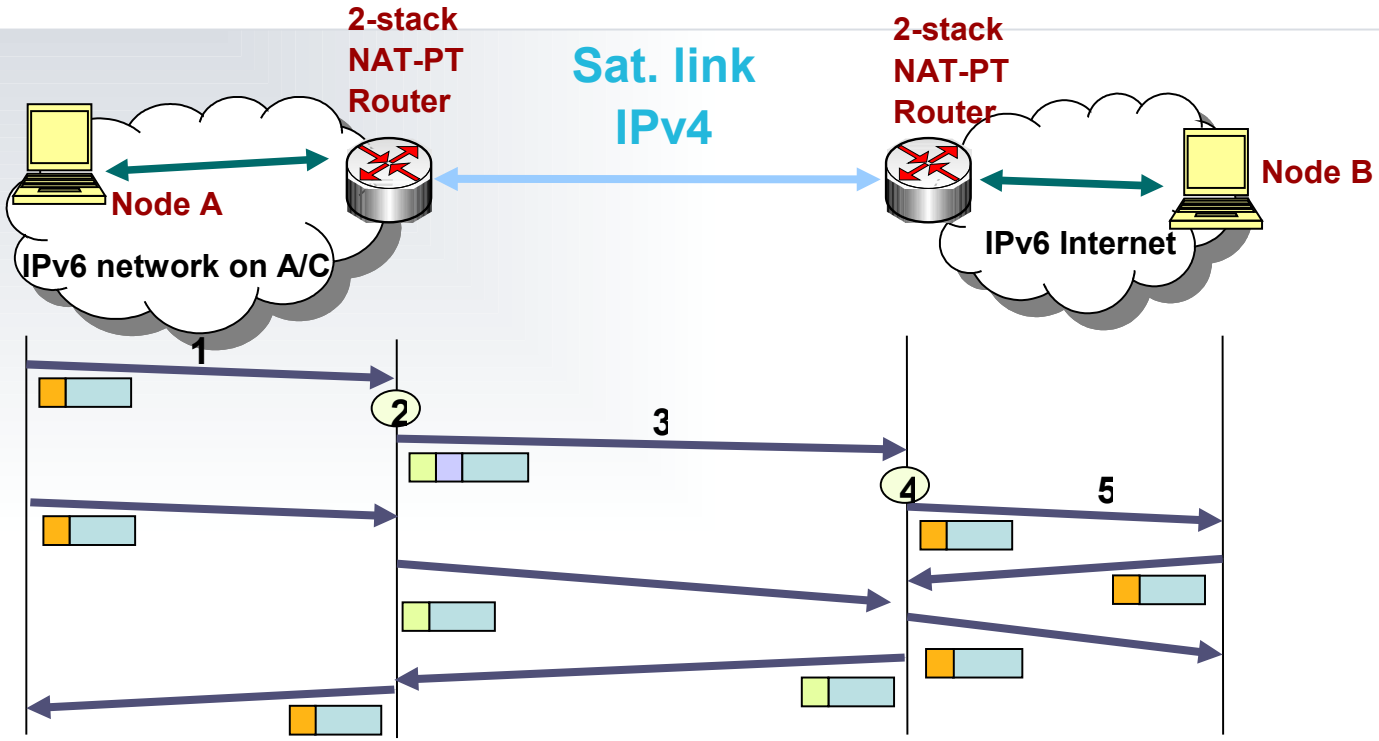


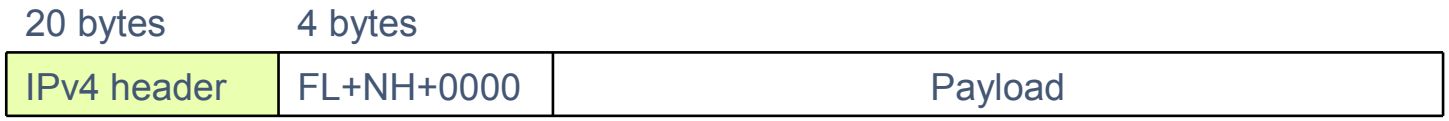
Dynamic tables for ports/addresses:

- Info sent from MR to GW or vice versa
- MR manages resources to be assigned

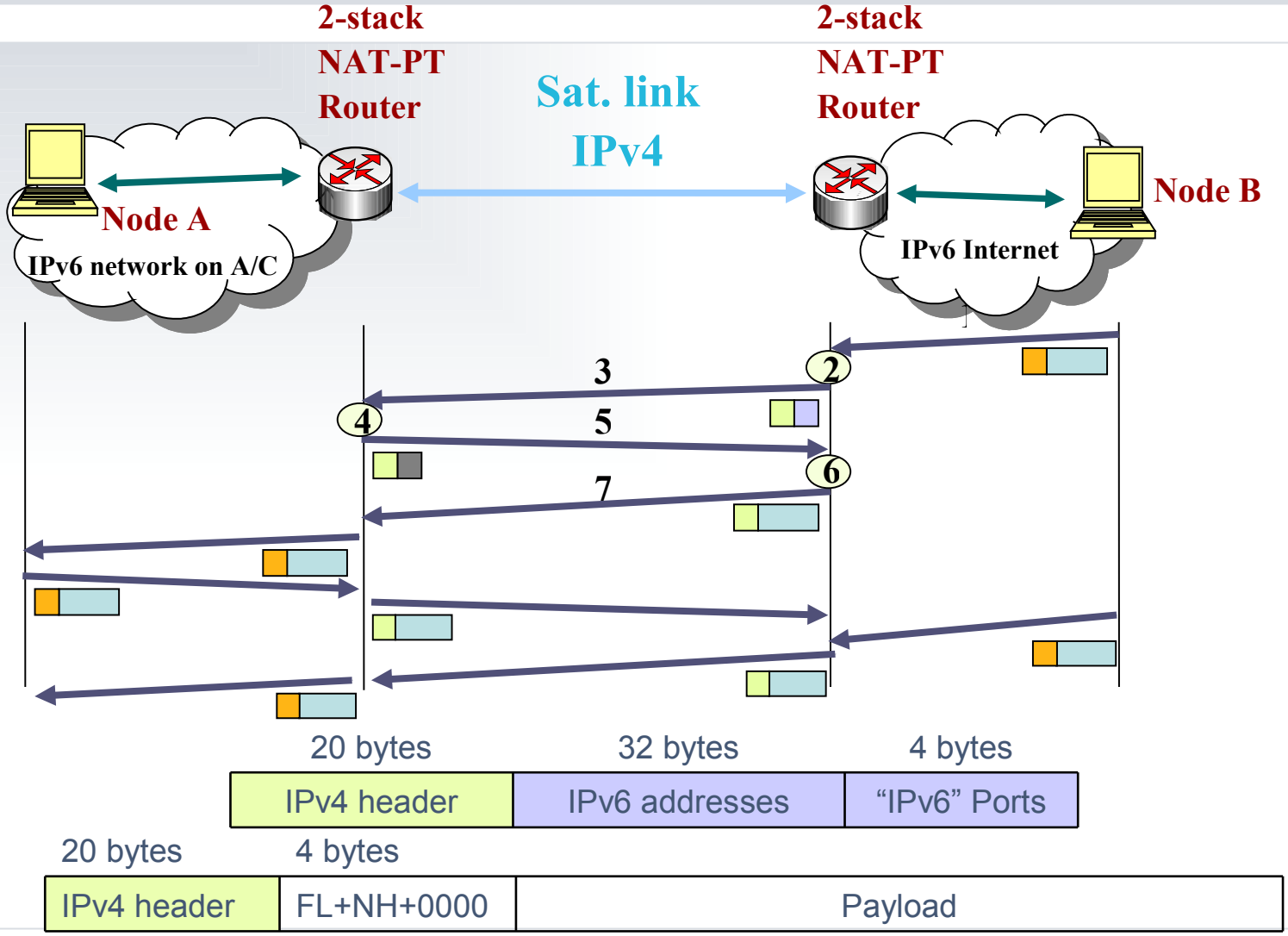
IPv6 Domain		IPv4 Domain	
Source	address port	Source	address port
Destination	address port	Destination	address port

NAPT-PT modified: cabin-originated packets

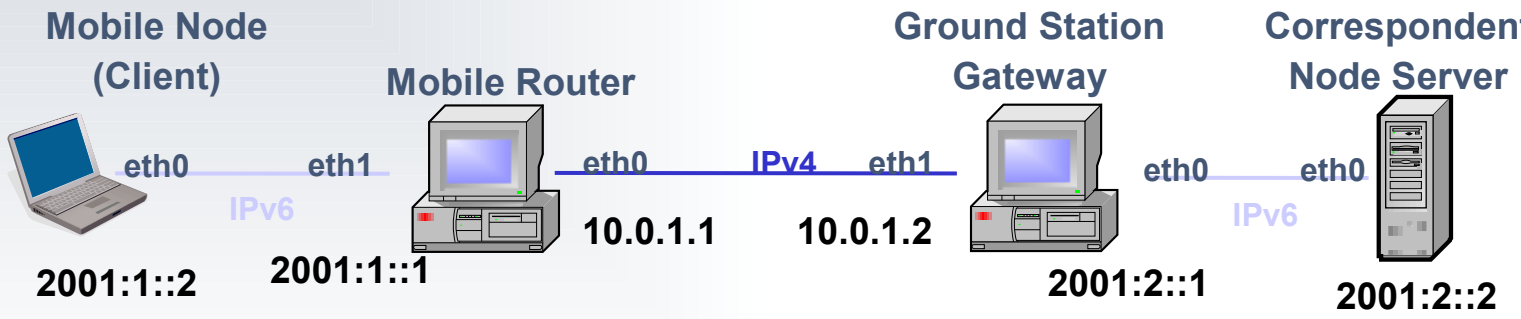




NAPT-PT modified: ground-originated packets



Simulation



All nodes implemented in XEN machine

Tunnel

- Existing module in Linux: ip tunnel

NAPT-PT modified

- Implemented in C/C++ in Linux user space

Measurements

Tunnel

		Sessions	Packet/Session	Bytes/packet	Payload
HTTP	**IPv4**	44	11	279	205
	IPv6			259	
FTP	**IPv4**	6	4645	1243	1169
	IPv6			1223	
VoIP	**IPv4**	1	18000	102	28
	IPv6			82	

NAPT-PT mod

		Sessions	Packet/Session	Bytes/packet			Payload (Bytes)
				First Packet	*Others*	*Mean*	
HTTP	**IPv4**	76	11	252	234	235	198
	IPv6			252			
FTP	**IPv4**	6	5118	1118	1102	1102	1064
	IPv6			1118			
VoIP	**IPv4**	1	18000	102	66	66,002	28
	IPv6			82			

Results

		HTTP	FTP	VoIP
NAPT-PT mod.	*Header Length (Bytes)*	25,44	24,03	24,002
	Overhead	2,50%	2,26%	86%
	Delay (ms)	0,79	0,75	0,75
Tunnel	*Header Length (Bytes)*	60	60	60
	Overhead	5,90%	5,60%	215%
	Delay (ms)	1,9	1,9	1,9
IPv6	*Header Length (Bytes)*	40	40	40
	Overhead	3,90%	3,70%	142%
	Delay (ms)	1,25	1,25	1,25

NAPT-PT better than Tunneling in terms of overhead

Conclusions

Problem:

- IPv4 satellite link with multiple IPv6 users in A/C

Best solution in terms of efficiency

2 Options:

- Tunneling
  - 2 headers: overhead
  - Simpler
- NAPT-PT mod
  - Only one header (IPv4): more efficient
  - More complex: translation necessary

Contact

TriaGnoSys GmbH

Axel Jahn, Angels Via

mail: Argelsrieder Feld 22
D-82234 Wessling

Tel: +49 8153 886780
Fax: +49 8153 886781

email: axel.jahn@triagnosys.com
angels.via@triagnosys.com

web: http://www.triagnosys.com