End-to-end Quality of Service support
over heterogeneous networks

EQSIP : SIP based QoS negotiation protocol

Stefano Salsano
University of Rome “Tor Vergata” / Pointercom

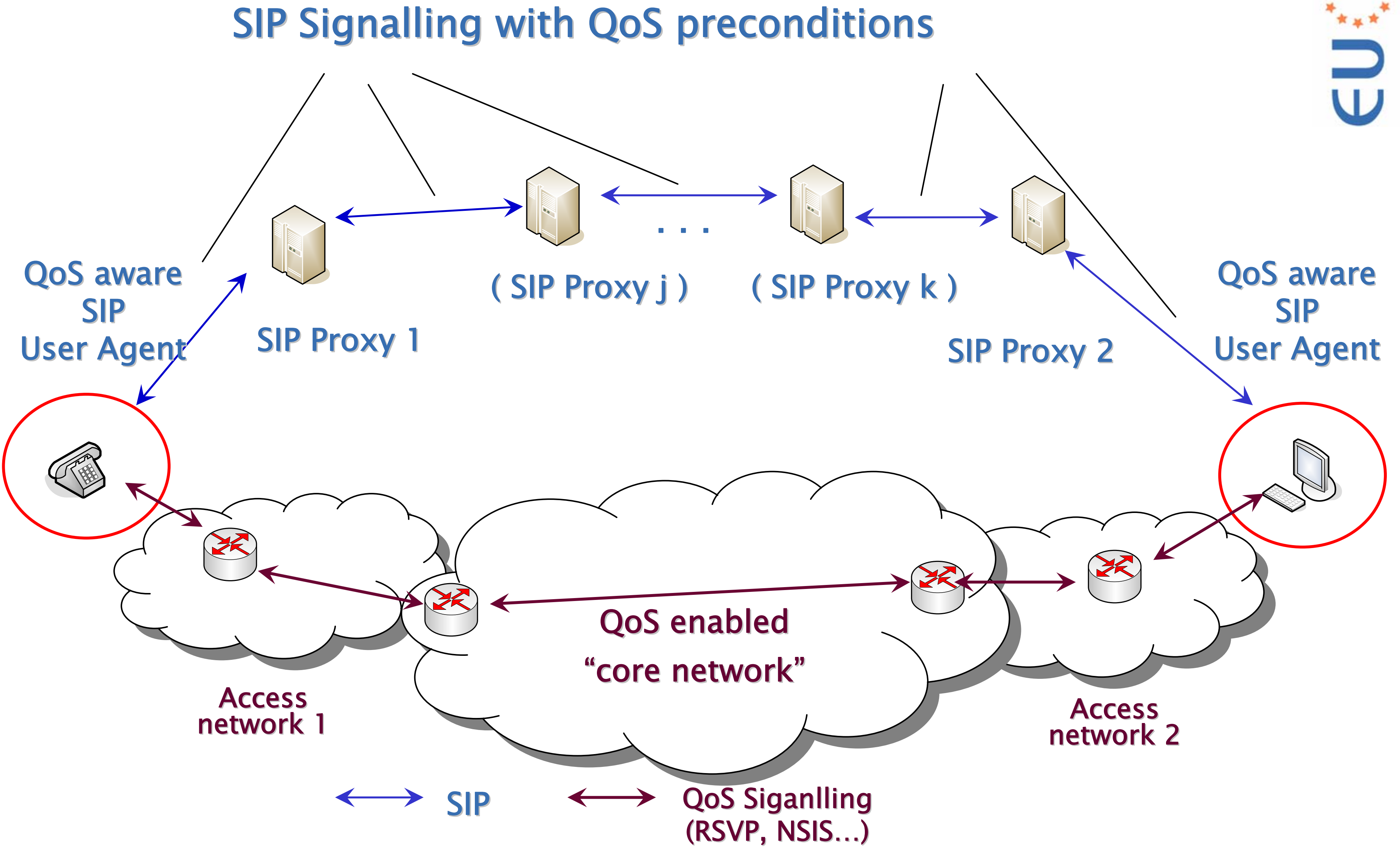
Acknowledgements

Luca Veltri (Univ. of Parma)
Gianluca Martiniello (Univ. of Roma “Tor Vergata”)
Donal Papalilo (PointerCom)
Roberto Marega (PointerCom)
Valeria Calcagni (PointerCom)

Summary

- SIP and QoS: current status (“preconditions” drafts)
- Issue 1 : terminals & QoS
- Issue 2 : coupling SIP and QoS negotiation
- The solution: EQSIP
- Standardization status and plans

SIP & QoS: current status



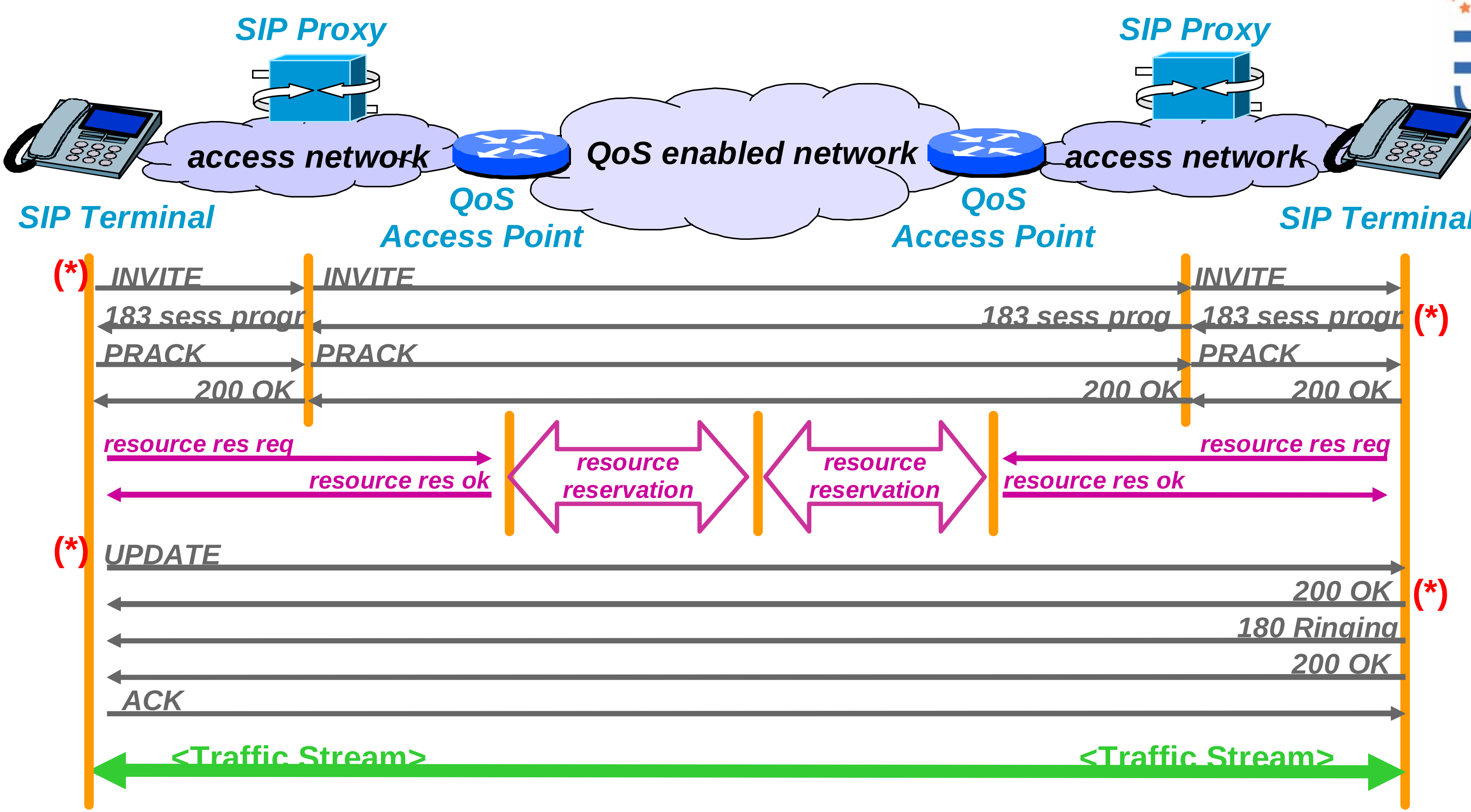
QoS preconditions

- Current standards (*RFCs 3312 and 4032*) introduce the concept of QoS precondition i.e. a set of constraints about the session to be established
  - the session cannot be established until QoS precondition are met
- Objective is to ensure that resources are made available before telephone rings
- Setup of the QoS reservation is a process originated by the user terminals
  - model not restricted to RSVP but originally designed with RSVP in mind
- Potential problem: resource reservation handled by the terminal increases complexity of the terminal itself
  - could be critical for very light terminals (small IP devices or other handheld IP based terminals)

QoS preconditions

- QoS preconditions are media stream specific
  - specified in SDP messages as attributes/parameters of the media
- Two media attributes enable the request for QoS:
  - current status attribute carries information about the current status of network resources reservation
  - desired status attribute carries information about the desired status of the network resources reservation (i.e. the QoS precondition!!!)
- Information carried in the status attributes:
  - status type: end-to-end, local and remote
  - precondition strength: mandatory, optional and none
  - direction of reservation: send, recv and sendrecv

QoS preconditions: message exchange



(*) Preconditions in SDP

Issues with the current solution

- QoS is requested by TERMINALS (Issue 1) and related to SIP via preconditions

- Preconditions only check that the QoS reservation has/has not been successful. It is not possible to negotiate QoS characteristics and agree on the QoS between the QoS aware entities in the SIP dialogue (Issue 2).

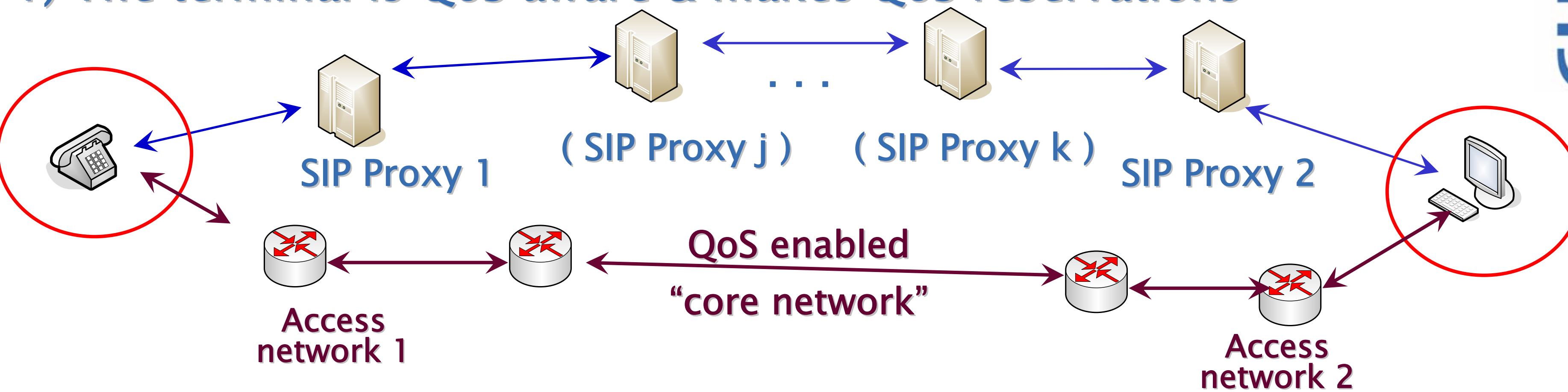
Issue 1: QoS & terminals

- In the current model the terminals must be "QoS aware" and capable to reserve the resource according to the "QoS language" spoken by the network

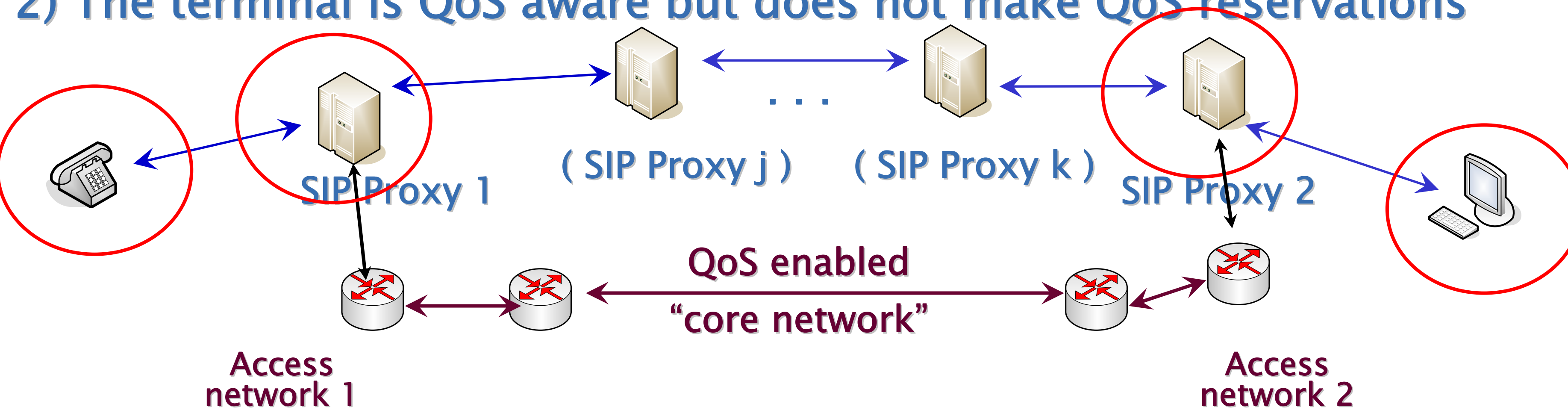
- We argue that a more flexible model should be supported, where the terminal can also be not aware of QoS and/or not able to speak the QoS language of the network

(Issue 1) Need for 3 scenarios

1) The terminal is QoS aware & makes QoS reservations

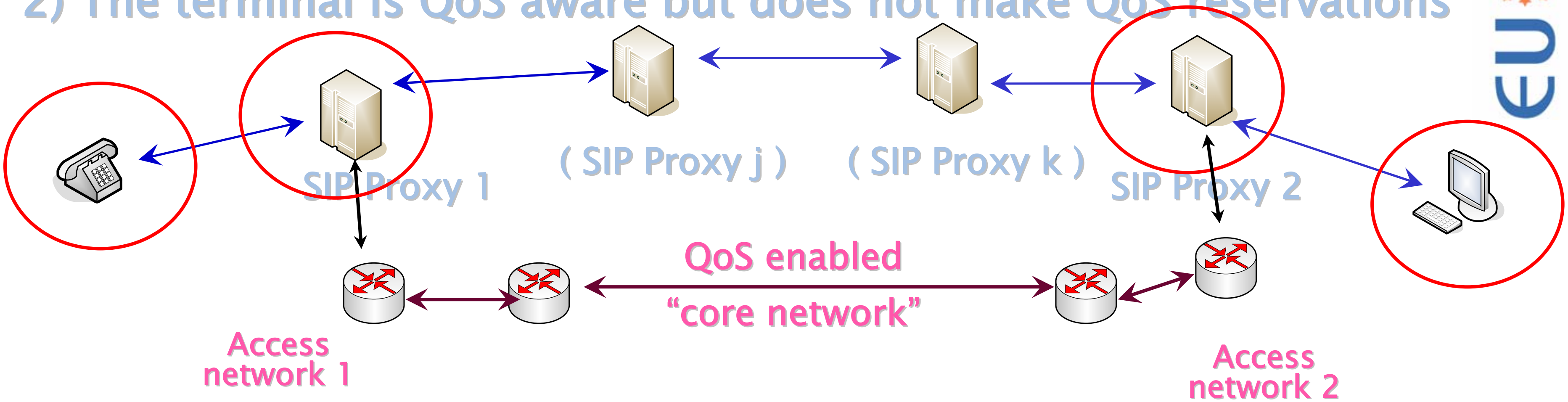


2) The terminal is QoS aware but does not make QoS reservations

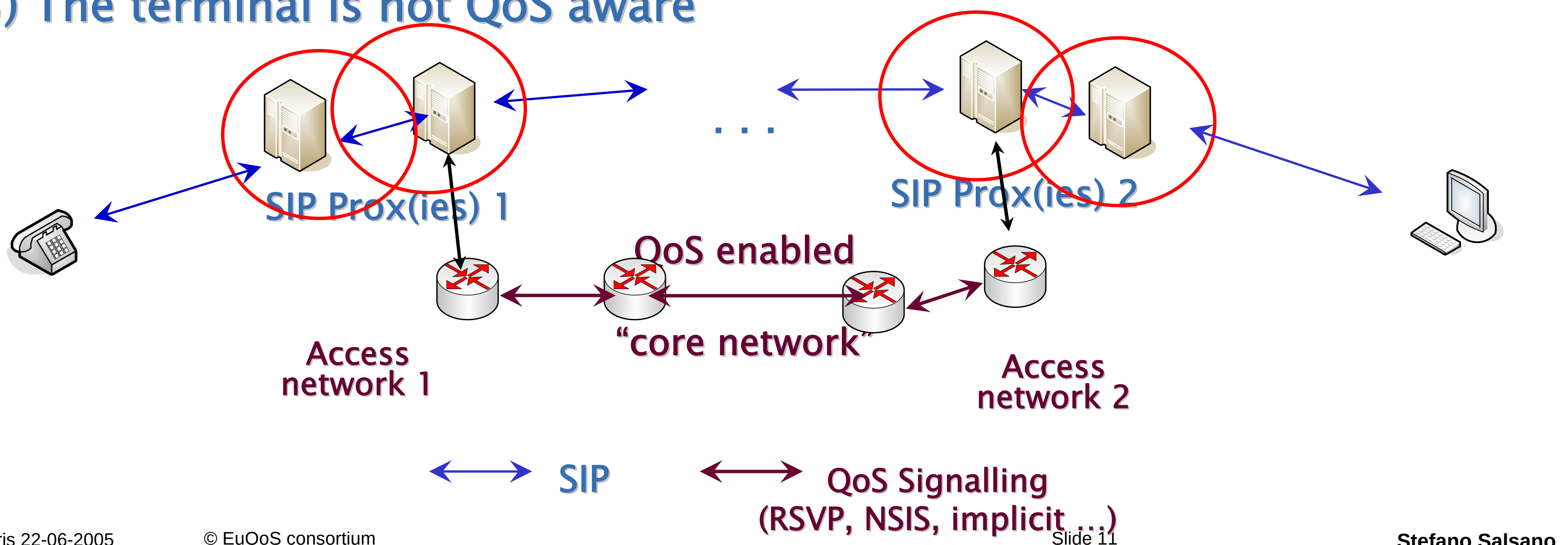


(Issue 1) Need for 3 scenarios

2) The terminal is QoS aware but does not make QoS reservations



3) The terminal is not QoS aware



(Issue 1) Comments on scenario 3

- Note that in the third scenario there are two functions that are performed by proxies on behalf of the terminal: the QoS negotiation and the resource reservation with the network.
- Two implementations are possible:
  - A “Gateway” performs the QoS negotiation and the proxy performs the resource reservation. This is very similar to scenario 2, with the “Gateway” that plays the role of user agent.
  - A single proxy can take care of both function.
- This solution has the advantage that terminals do not need to be upgraded and can be very simple. It also is in line with the current trend of having “Session Border Controllers”, to control SIP and media transmission at the border of a network.

(Issue 2) QoS negotiation in SIP

- With current QoS preconditions drafts the terminal can only say it wants QoS and if QoS has been reserved. There is no way to express which level of QoS per media stream is required or desired by the end-user.

- The QoS really needed by the user cannot be derived accurately from information available in the SDP: the codec and the <u>optional</u> bandwidth parameter ("b=").

(Issue 2) QoS negotiation in SIP

- End-users cannot reach a QoS agreement at session set-up using SIP.
- SIP proxies of local access/service provider cannot fully control the QoS requirements
- It is much more difficult for the service providers to deliver "predictable end-to-end QoS" to their customers (and charge for QoS…)

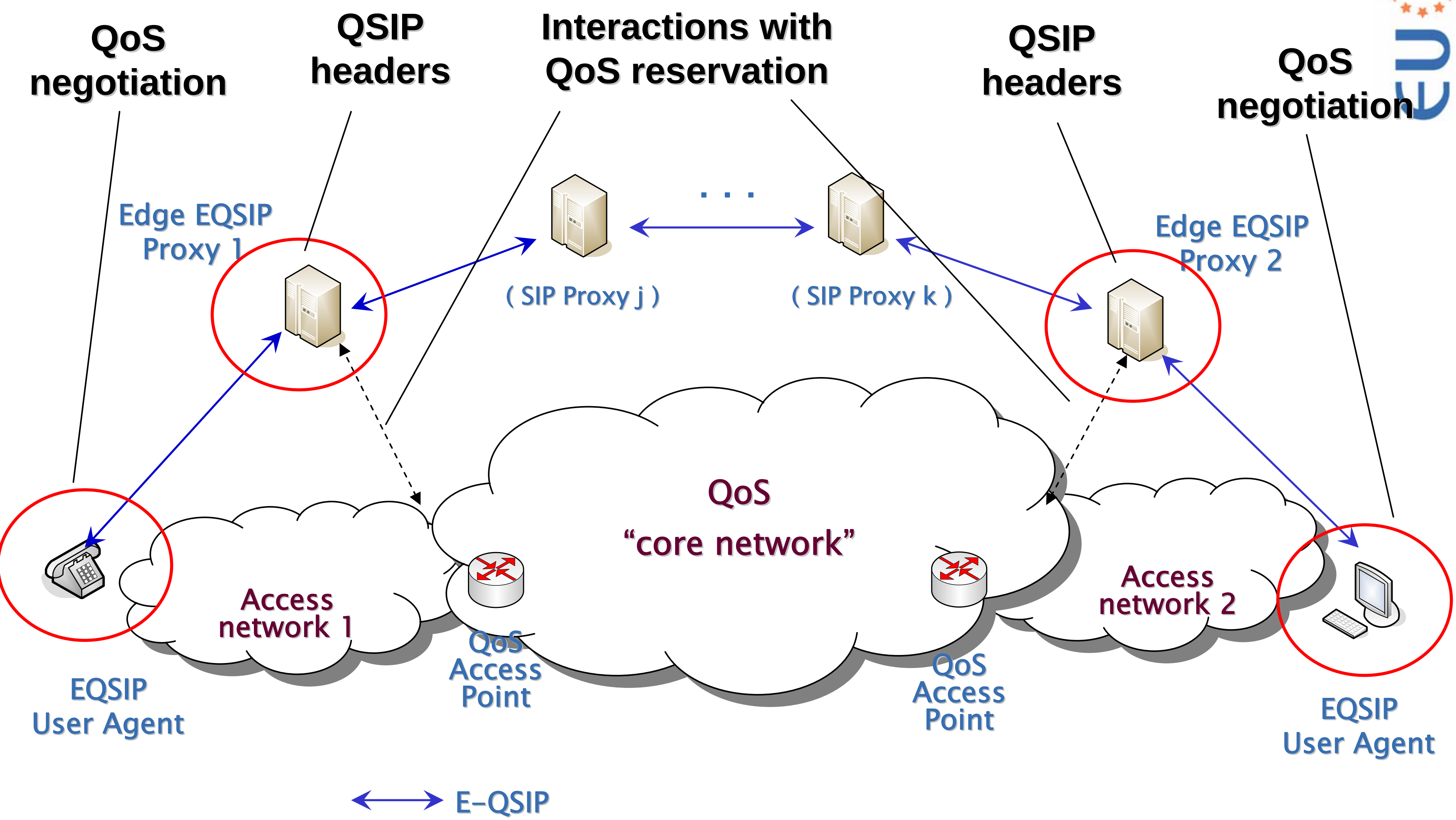
Summary

- SIP and QoS: current status (“preconditions” drafts)
- Issue 1 : terminals & QoS
- Issue 2 : coupling SIP and QoS negotiation
- The solution: EQSIP
- Standardization status and plans

The solutions

- (1) QSIP: adding QoS headers in SIP messages to allow resource reservation also from intermediate entities (i.e. SIP proxies)
- (2) Enhancing SDP to support QoS negotiation between QoS aware SIP entities
- $\Rightarrow$ EQSIP

EQSIP framework



Issue 1 solution: Q-SIP

- QSIP proposal (*draft-veltri-sip-qsip-01*) enhances SIP protocol to convey QoS related information
- The solution preserves backward compatibility and include the handling of QoS in the SIP signaling in a flexible way
- Resource reservation can be handled by terminals or by proxy servers according to the scenario
- Generic mechanism to exchange QoS information
  - it can be particularized for a specific QoS mechanism by defining specific information elements

QSIP header

```
INVITE sip:remotesystem@euqos.org SIP/2.0
Via: SIP/2.0/UDP 160.80.83.1:5060;branch=z9hG4bdte73k
Via: SIP/2.0/UDP 160.80.83.81:5065;branch=z9hG4bK74b43
From: Local User <sip:user@euqos.org:5065>;tag=9fxced76sl
To: Remote User <sip: remotesystem@euqos.org>
Call-ID: 3848276298220188511@160.80.83.81
CSeq: 1 INVITE
QoS-Info: rm-addr=160.80.82.1
Contact: <sip:160.80.83.81:5065>
Content-Type: application/sdp
Content-Length: 250

v=0
o=user 2890844526 2890844526 IN IP4 160.
s=-
c=IN IP4 160.80.83.81
t=0 0
m=audio 49170 RTP/AVP 0 3
a=curr:qos e2e none
a=des:qos mandatory e2e send
a=rtpmap:0 PCMU/8000
a=qos:char1 RT 64 10
a=qos:char2 RT 32 20
a=rtpmap:3 GSM/8000
```

QSIP header added by QSIP proxy who wants to be in charge of resource reservation

Issue 2 solution: QoS negotiation with SIP

- Involve SIP in the QoS negotiation defining a framework in which:
  - end-users applications negotiate QoS requirements and characteristics of the media components in a session
  - EQSIP proxies are able to derive QoS requirements and characteristics of the applications
- Application QoS information expressed within SDP body of SIP messages
- The QoS negotiation is modeled after the existing Offer/Answer negotiation of codecs

Issue 2 solution: QoS negotiation with SIP

- This issue has already been identified in the context of SDPng work within mmusic IETF WG.
- It has been proposed to support QoS negotiation in SDP new generation, with a set of SDPng extension fields.
- The need of implementing a simplified solution for current SDP was identified, but to our knowledge there has been no follow up.

How it might look like ?

```
INVITE sip:remotesystem@euqos.org SIP/2.0
Via: SIP/2.0/UDP 160.80.83.81:5065;branch=z9hG4bK74b43
From: Local User <sip:user@euqos.org:5065>;tag=9fxced76sl
To: Remote User <sip: remotesystem@euqos.org>
Call-ID: 3848276298220188511@160.80.83.81
CSeq: 1 INVITE
Contact: <sip:160.80.83.81:5065>
Content-Type: application/sdp
Content-Length: 250

v=0
o=user 2890844526 2890844526 IN IP4 160.80.83.81
s=-
c=IN IP4 160.80.83.81
t=0 0
m=audio 49170 RTP/AVP 0 3
a=curr:qos e2e none
a=des:qos mandatory e2e send
a=rtpmap:0 PCMU/8000
a=qos:char1 RT 64 10
a=qos:char2 RT 32 20
a=rtpmap:3 GSM/8000
a=qos:char1 RT 13 5
```

How it might look like ?

```
INVITE sip:remotesystem@euqos.org SIP/2.0
Via: SIP/2.0/UDP 160.80.83.81:5065;branch=z9hG4bK74b43
From: Local User <sip:user@euqos.org:50          76sl
To: Remote User <sip: remotesystem@euqos
Call-ID: 3848276298220188511@160.80.83.8
CSeq: 1 INVITE
Contact: <sip:160.80.83.81:5065>
Content-Type: application/sdp
Content-Length: 250

v=0
o=user 2890844526 2890844526 IN IP4 160.80.83.81
s=-
c=IN IP4 160.80.83.81
t=0 0
m=audio 49170 RTP/AVP 0 3
a=curr:qos e2e none
a=des:qos mandatory e2e send
a=rtpmap:0 PCMU/8000
a=qos:char1 RT 64 10
a=qos:char2 RT 32 20
a=rtpmap:3 GSM/8000
a=qos:char1 RT 13 5
```

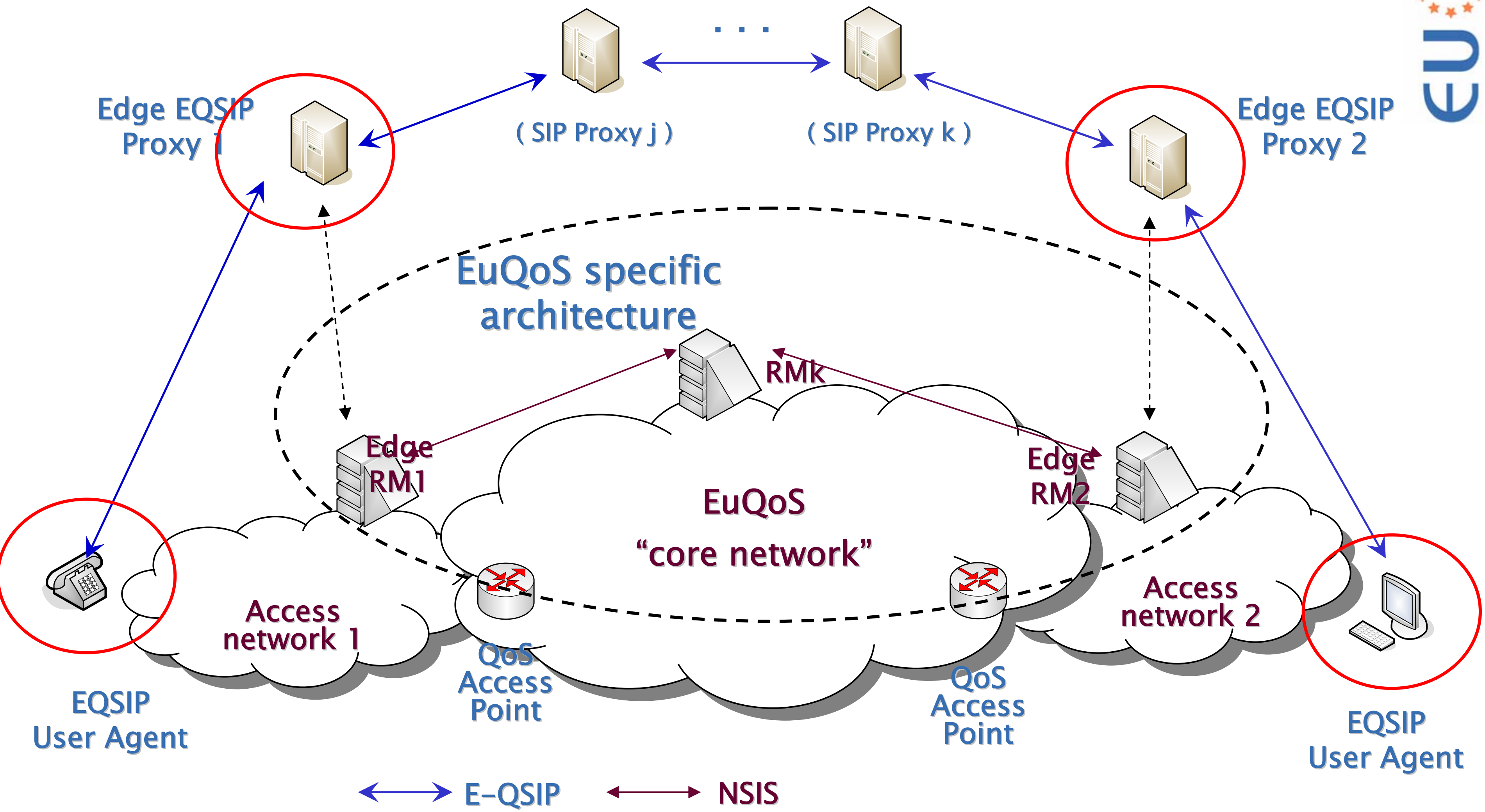
SDP

Preconditions

QoS negotiation offer/answer attibutes:

a=qos:<descriptor> <network/class service id> <pbr> <pbrt>

EQSIP framework & EuQoS



Summary

- SIP and QoS: current status ("preconditions" drafts)
- Issue 1 : terminals & QoS
- Issue 2 : coupling SIP and QoS negotiation
- The solution: EQSIP
- Standardization status and plans

QSIP status

- First version of QSIP was proposed in 2001
  - L. Veltri and S. Salsano, “QoS Support for SIP Based Applications in DiffServ Networks,” <draft-veltri-sip-qsip-00.txt>, Oct. 2001, work in progress, (submitted as internet draft, but never pushed nor presented in IETF)
  - S. Salsano, L. Veltri, “QoS Control by means of COPS to support SIP based applications”, IEEE Networks, March/April 2002

QSIP status

- An updated version was prepared in 2002
  - L. Veltri, S. Salsano, D. Papalilo, "SIP Extension for QoS support in Diffserv Networks", Oct 2002 (actually, this was not submitted to IETF, but it was made available at www.coritel.it/projects/qsip)
  - L. Veltri, S. Salsano, D. Papalilo, “QoS Support for SIP Based Applications in a Diffserv Network”, IEEE Softcom 2003, October 7-10, 2003, Split, Dubrovnik (Croatia), Ancona, Venice (Italy)

Next steps

- We think it could be worth submitting an internet draft on the proposed solutions:

1) Updating the QSIP draft
2) Writing a draft on QoS negotiation in SIP (currently only described in EuQoS documents)

- **Authors welcome comments, contributions and support from EuQoS partners (and from anybody interested) !**

- Next IETF meeting is in Paris soon…

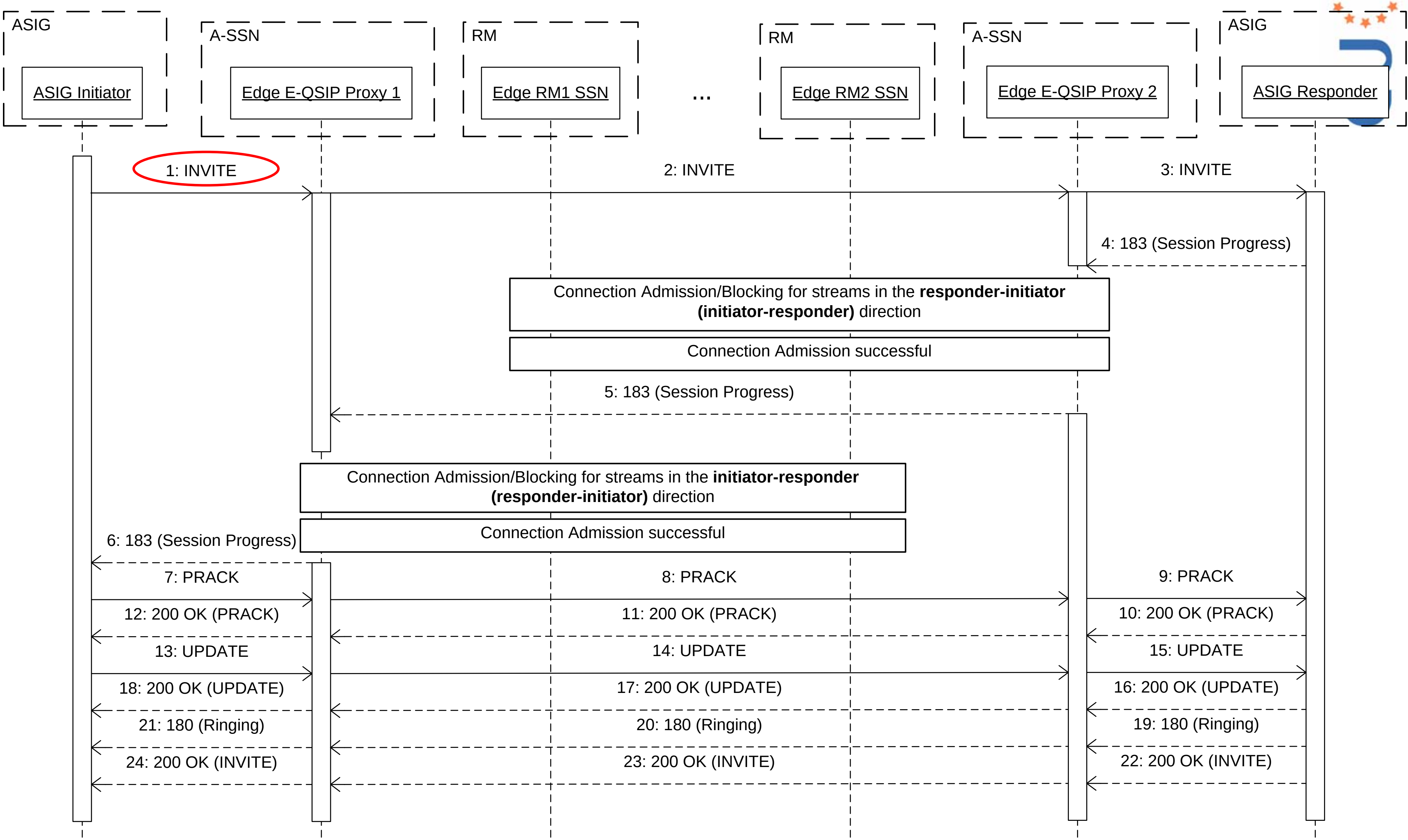
End-to-end Quality of Service support
over heterogeneous networks

Thank you for your attention !

Backup material

- Message exchanges from current EuQoS implementation (work in progress)

Message flows (1/3)



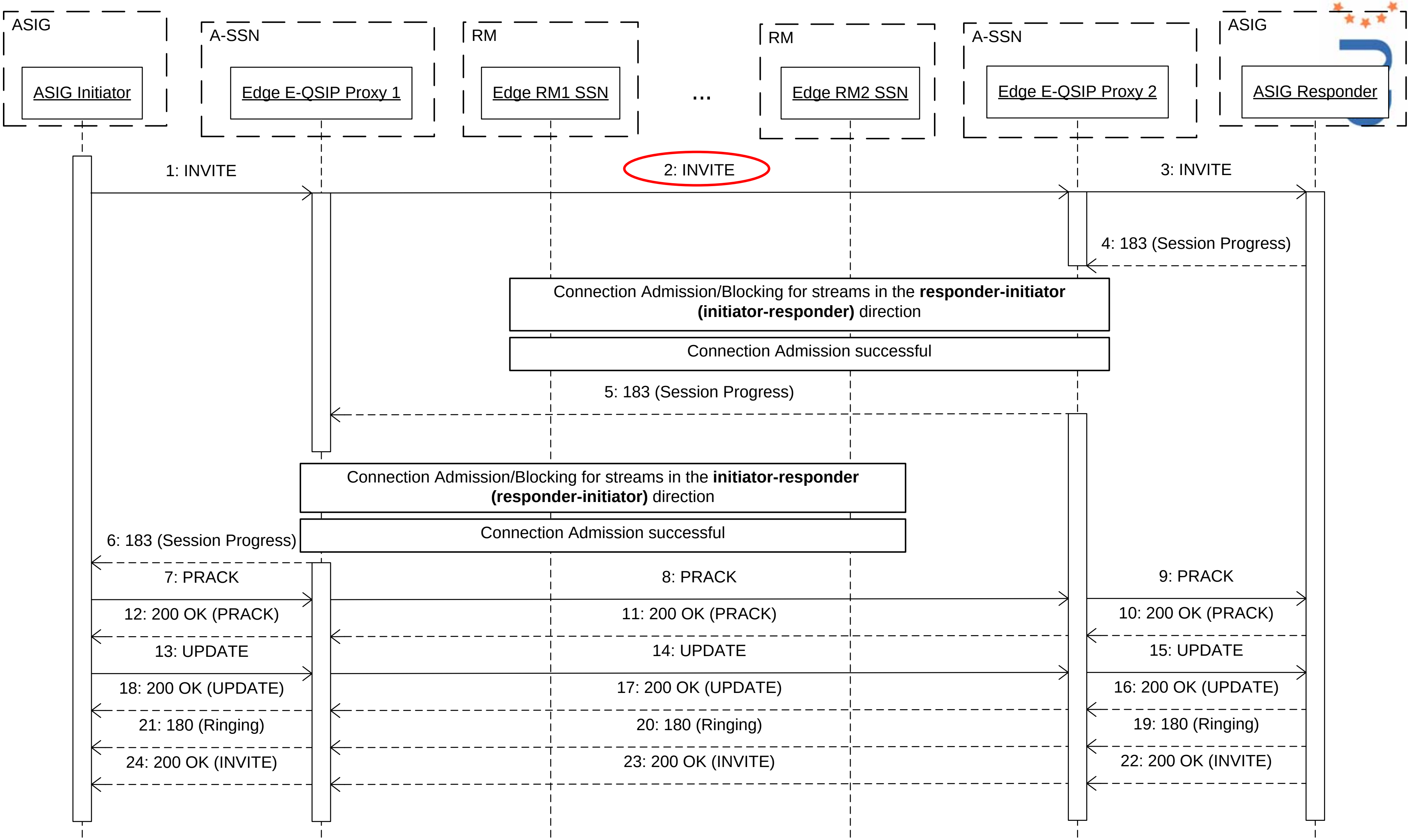
Message flows (1 / 3)

```
INVITE sip:remotesystem@euqos.org SIP/2.0
Via: SIP/2.0/UDP 160.80.83.81:5065;branch=z9hG4bK74b43
From: Local User <sip:user@euqos.org:5065>;tag=9fxced76sl
To: Remote User <sip: remotesystem@euqos.org>
Call-ID: 3848276298220188511@160.80.83.81
CSeq: 1 INVITE
Contact: <sip:160.80.83.81:5065>
Content-Type: application/sdp
Content-Length: 250

v=0
o=user 2890844526 2890844526 IN IP4 160.80.83.81
s=-
c=IN IP4 160.80.83.81
t=0 0
m=audio 49170 RTP/AVP 0 3
a=curr:qos e2e none
a=des:qos mandatory e2e send
a=rtpmap:0 PCMU/8000
a=qos:char1 RT 64 10
a=qos:char2 RT 32 20
a=rtpmap:3 GSM/8000
a=qos:char1 RT 13 5
```

a=qos:<descriptor> <network/class service id> <pbr> <pbrt>

Message flows (2/3)

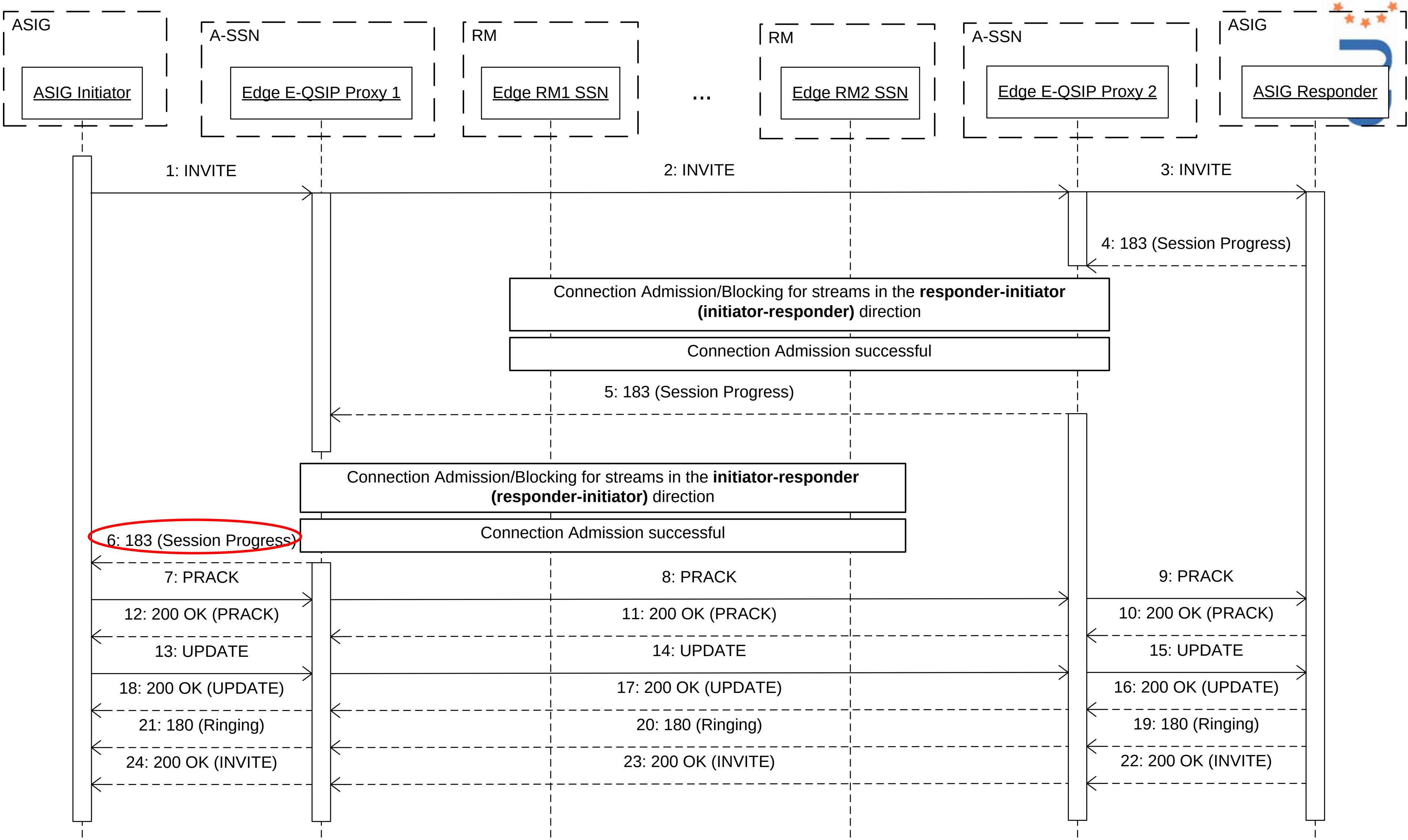


Message flows (2/3)

```
INVITE sip:remotesystem@euqos.org SIP/2.0
Via: SIP/2.0/UDP 160.80.83.1:5060;branch=z9hG4bdte73k
Via: SIP/2.0/UDP 160.80.83.81:5065;branch=z9hG4bK74b43
From: Local User <sip:user@euqos.org:5065>;tag=9fxced76sl
To: Remote User <sip: remotesystem@euqos.org>
Call-ID: 3848276298220188511@160.80.83.81
CSeq: 1 INVITE
QoS-Info: rm-addr=160.80.82.1
Contact: <sip:160.80.83.81:5065>
Content-Type: application/sdp
Content-Length: 250

v=0
o=user 2890844526 2890844526 IN IP4 160.80.83.81
s=-
c=IN IP4 160.80.83.81
t=0 0
m=audio 49170 RTP/AVP 0 3
a=curr:qos e2e none
a=des:qos mandatory e2e send
a=rtpmap:0 PCMU/8000
a=qos:char1 RT 64 10
a=qos:char2 RT 32 20
a=rtpmap:3 GSM/8000
```

Message flows (3/4)



Message flows (3/3)

```
SIP/2.0 183 Session Progress
Via: SIP/2.0/UDP 160.80.83.81:5065;branch=z9hG4bK74b43
Record-Route: <sip:eqproxy1.euqos.org;lr>, <sip:eqproxy2.euqos.org;lr>
From: Local User <sip:user@euqos.org:5065>;tag=9fxced76sl
To: Remote User <sip: remotesystem@euqos.org>;tag=9103874
Call-ID: 3848276298220188511@160.80.83.81
CSeq: 1 INVITE
Contact: <sip:151.123.92.100:5072>
Content-Type: application/sdp
Content-Length: 220

v=0
o=remotesystem 2890844527 2890844527 IN IP4 127.0.0.1
s=-
c=IN IP4 151.123.92.100
t=0 0
m=audio 49172 RTP/AVP 0
a=curr:qos e2e sendrecv
a=des:qos mandatory e2e send
a=conf:qos e2e recv
a=rtpmap:0 PCMU/8000
a=qos:char2 RT 32 20
```