“Classical” RSVP over IP over ATM

Steven Berson

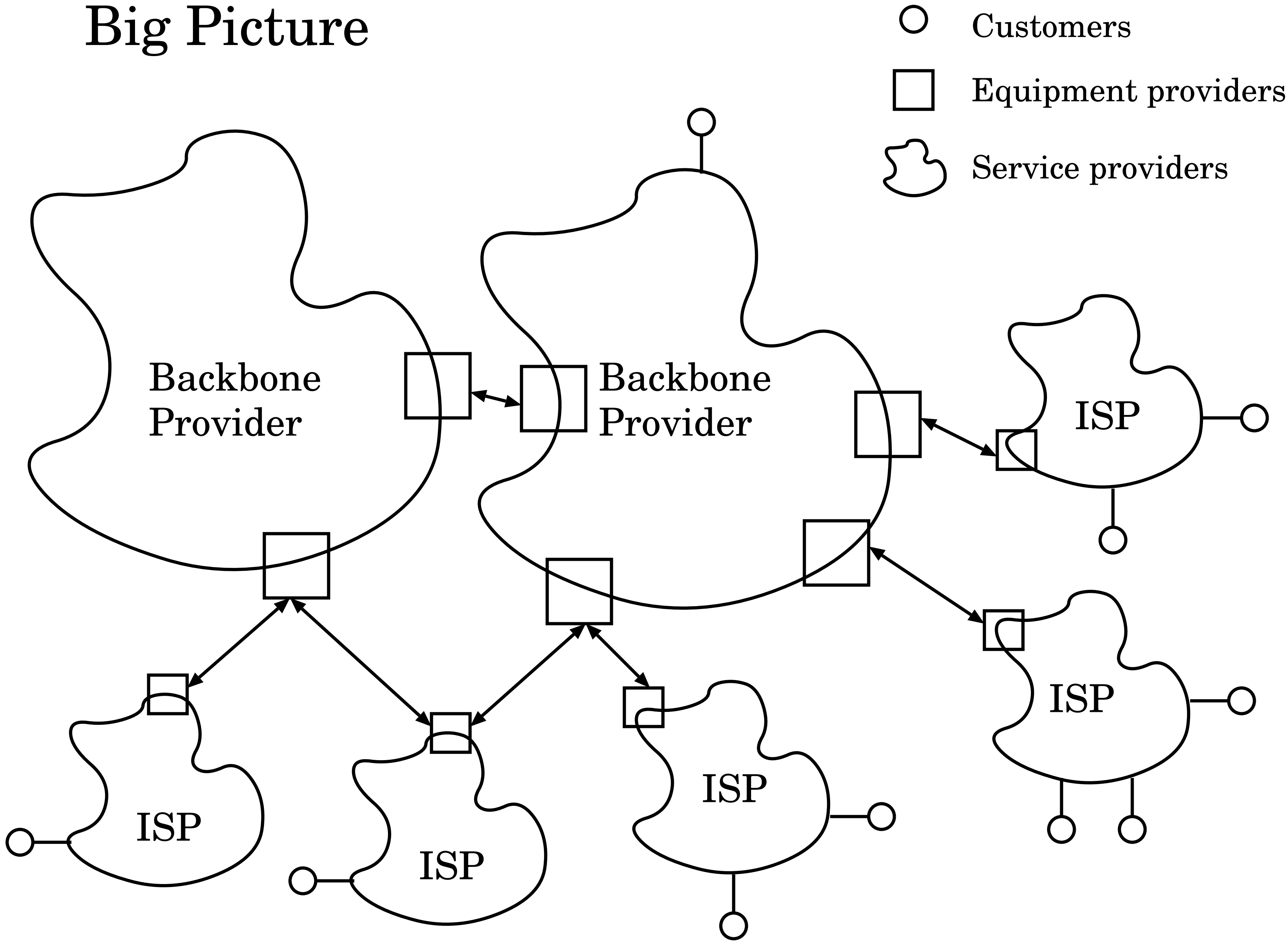
USC Information Sciences Institute

June, 1996

Outline

- Motivation
- Data VCs
- Control VCs
- Miscellaneous issues

Big Picture


Customers
Equipment providers
Service providers
Backbone Provider
Backbone Provider
ISP
ISP
ISP
ISP
ISP
ISP

Goals

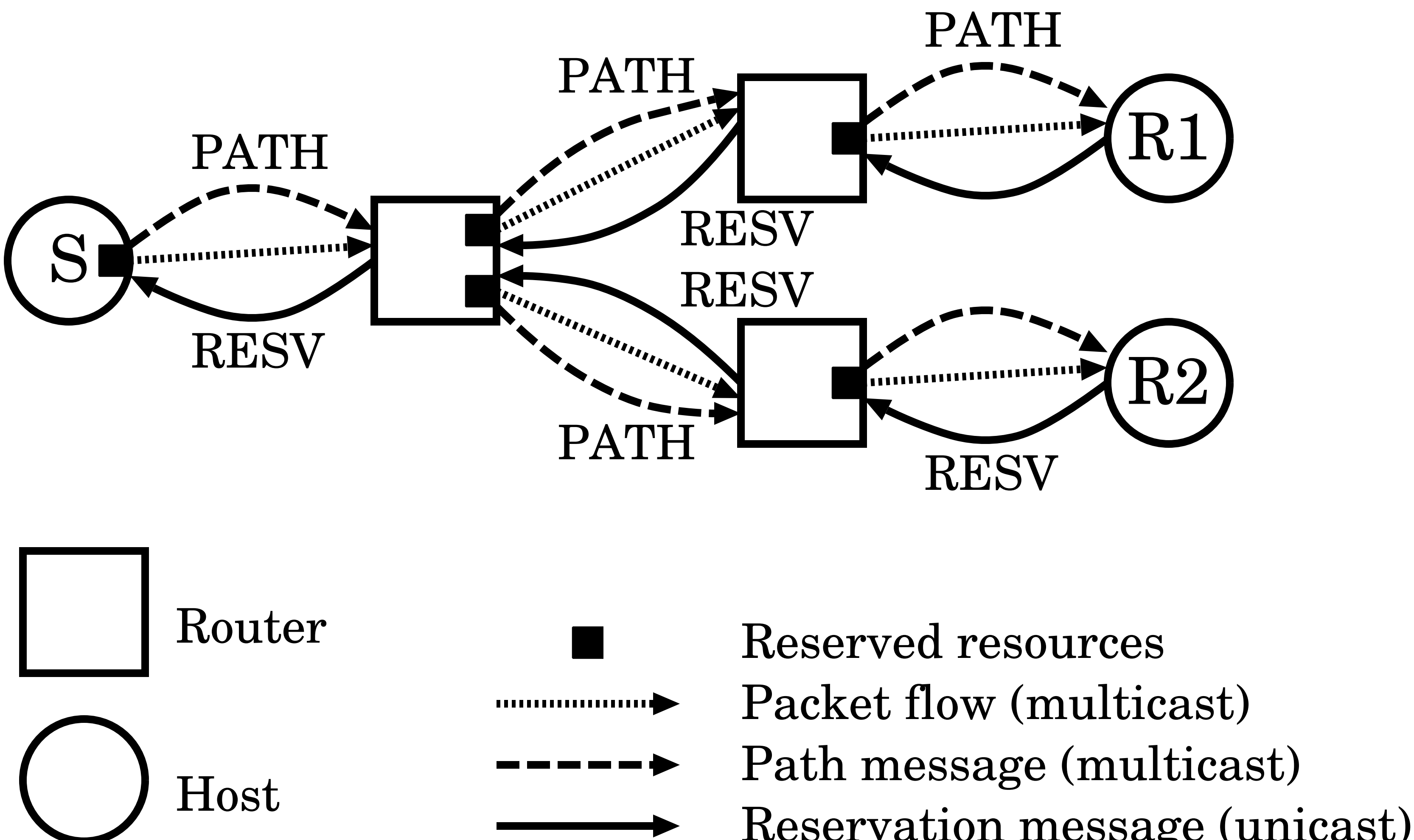
Manage data and control VCs effectively for

- Interoperability
- Maximum flexibility in services for
  - Customers
  - Internet Service Providers (ISPs)
  - Equipment Vendors

Assumptions

1. IP multicast (and unicast)
2. IP "best effort service" is default service
   - BE should not be adversely affected if someone else makes a reservation
3. Classical IP over ATM (RFC1577, etc)
4. ATM UNI 3.0/3.1
5. VC mesh (not MCS) unless otherwise noted
6. Segregation of traffic into flows done by RSVP filtering

Types of flows (control & data)



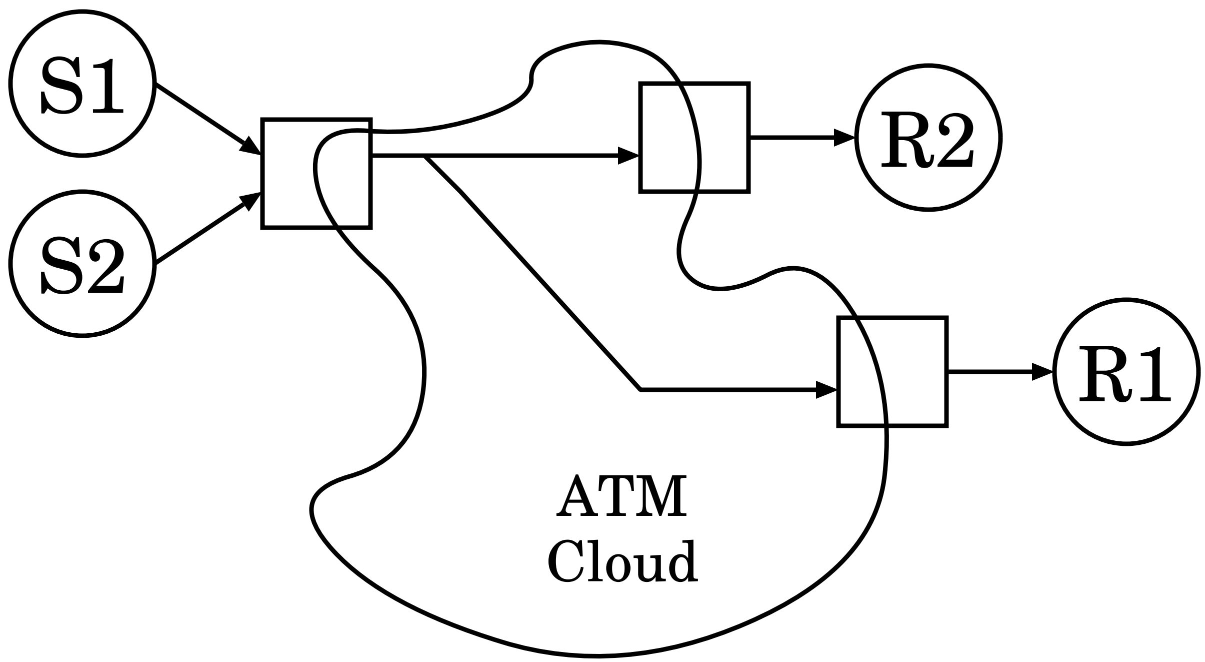
Outline

- Motivation
- Data VCs
- Control VCs
- Miscellaneous issues

Mapping data flows into VCs

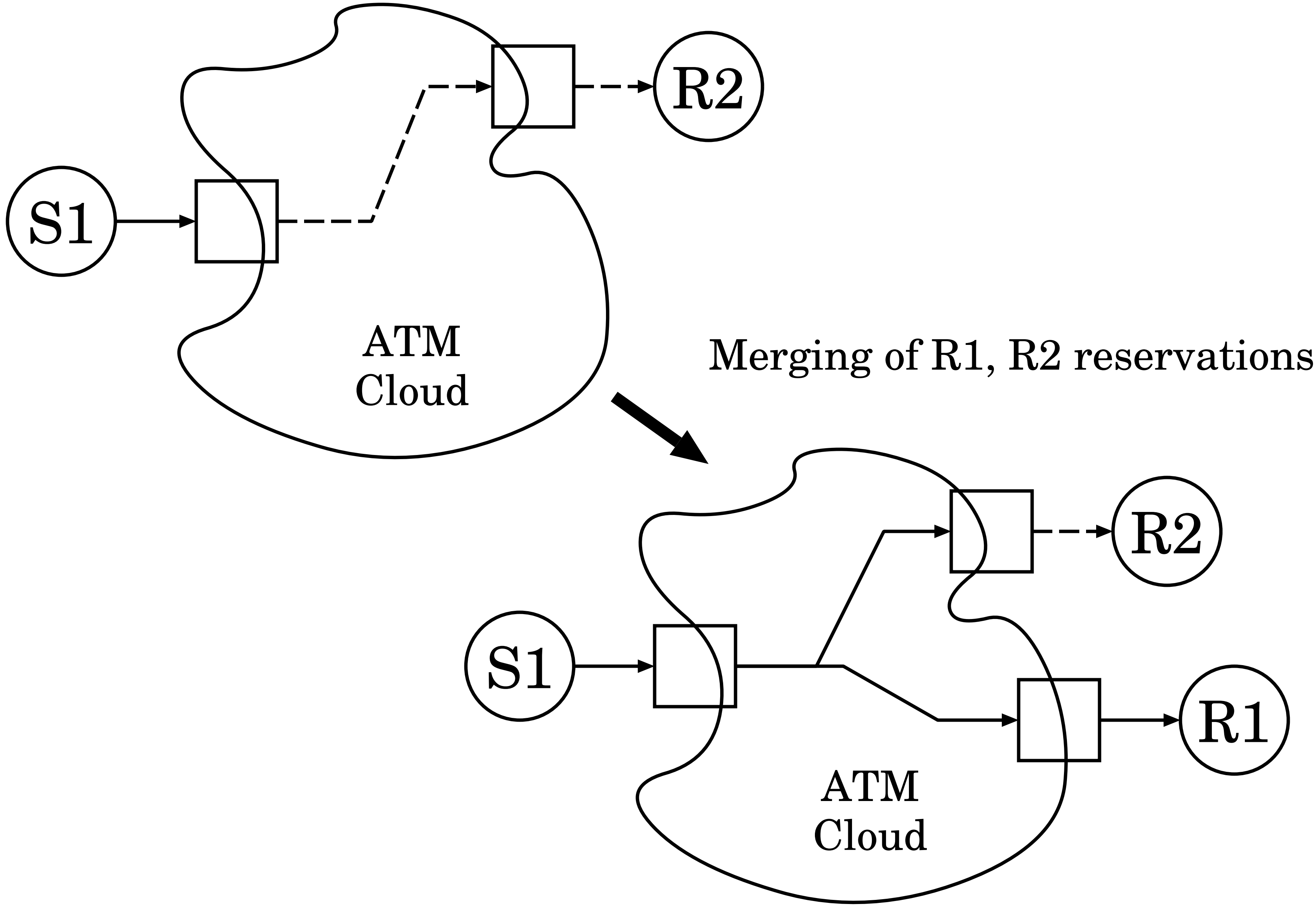
Homogeneous	1 QoS VC (or BE)	1 VC
Limited heterogeneity	BE + 1 QoS VC	2 VCs
Full heterogeneity	BE + multiple QoS VCs	many VCs
Aggregation	Multiplexing	< 1 (or 2) VCs
Other point-to-point hybrids		varies

1. Single VC per flow



R1 and R2 ask for the same QoS

Merging (QoS change)



Limitations of single VC per flow

1. Best effort “free ride”
2. Best effort can’t get QoS to join existing VC
3. Nonconforming traffic and BE VC

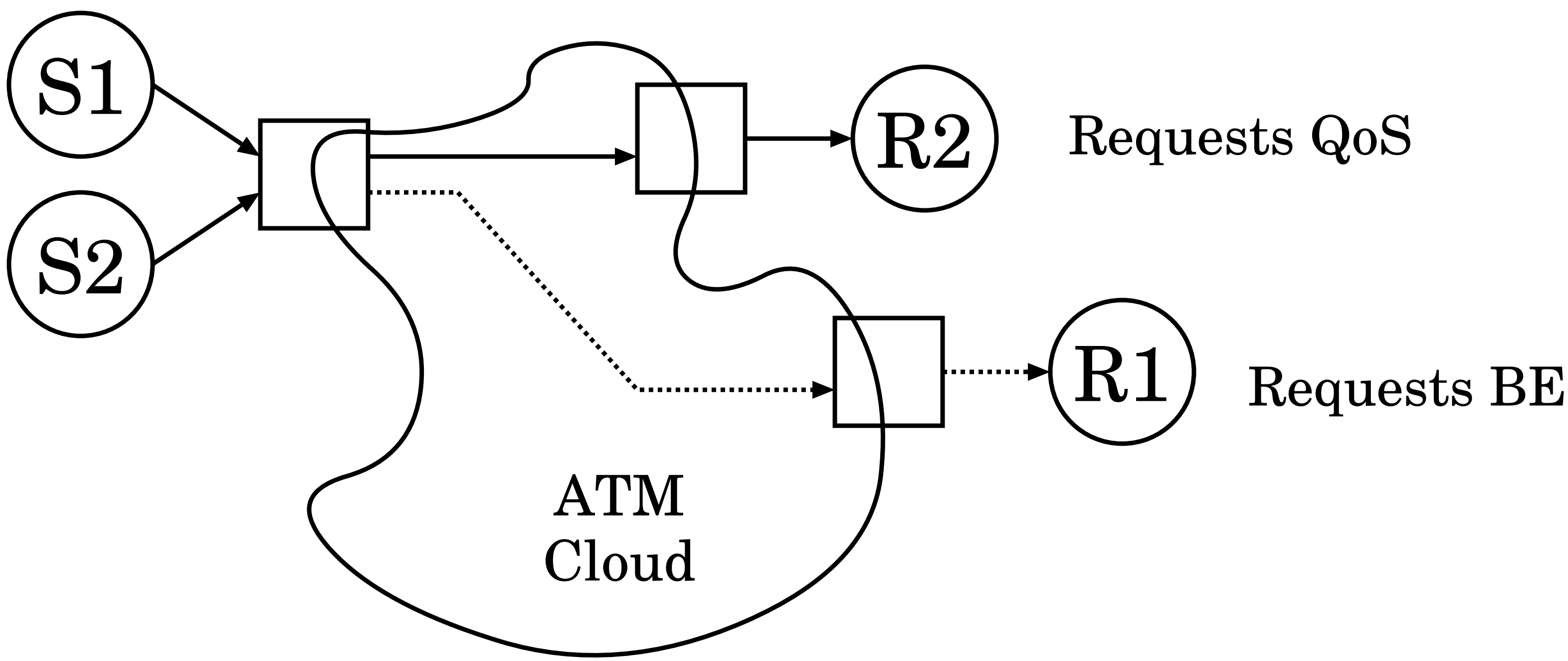
But, ...

1. Simple
2. Zero bandwidth wasted model
3. Reasonable for hierarchical encoded multicast groups

Multiple VCs per flow

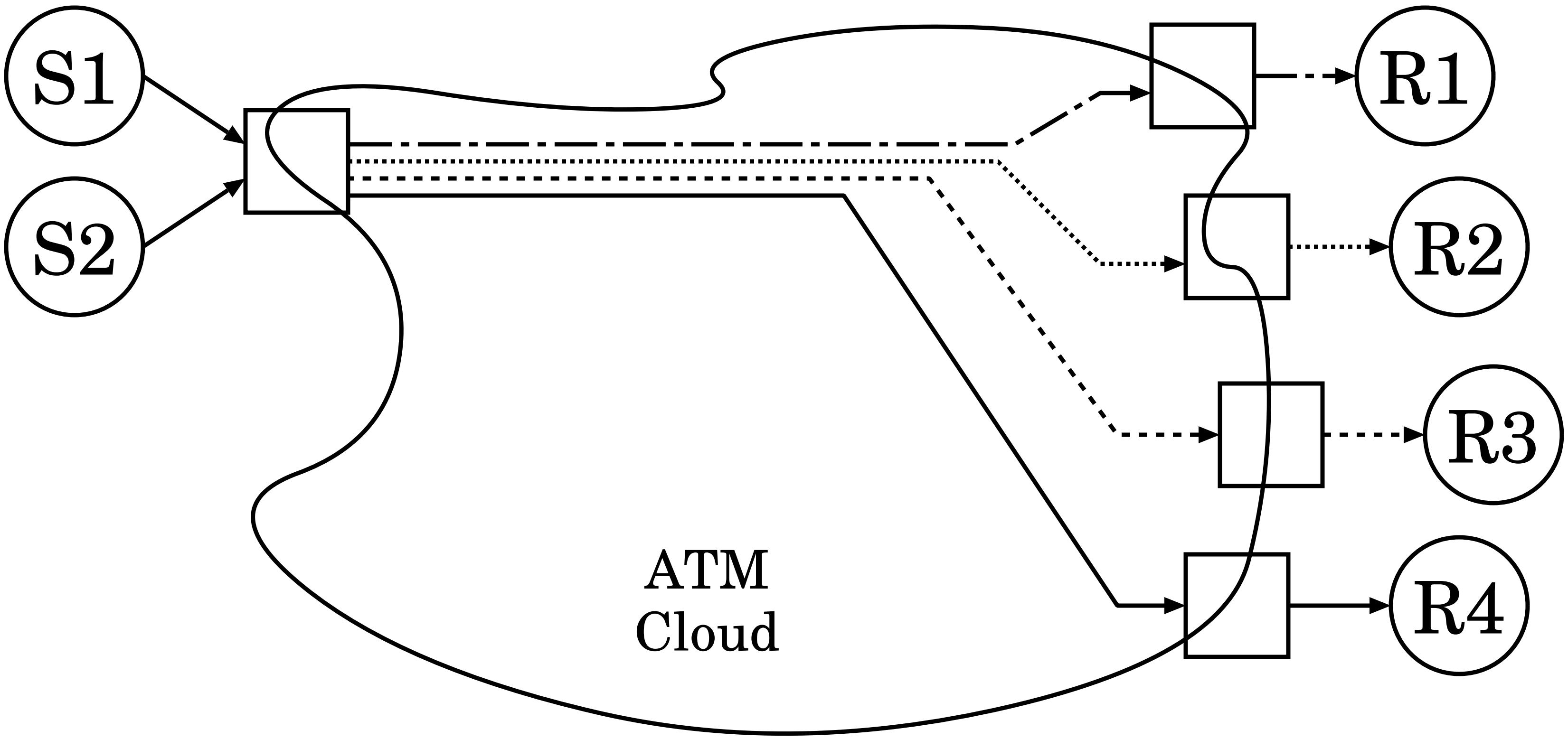
Heterogeneity caused by:

1. Default service vs. reserved service
2. Receivers making different reservations for the same session



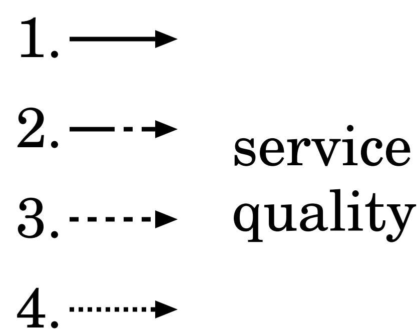
Result - 2 VCs, each with the same data

3. Supporting full heterogeneity



R1, R2, R2, and R4 all request different QoS

implies 4 different VCs



Limitations of full heterogeneity

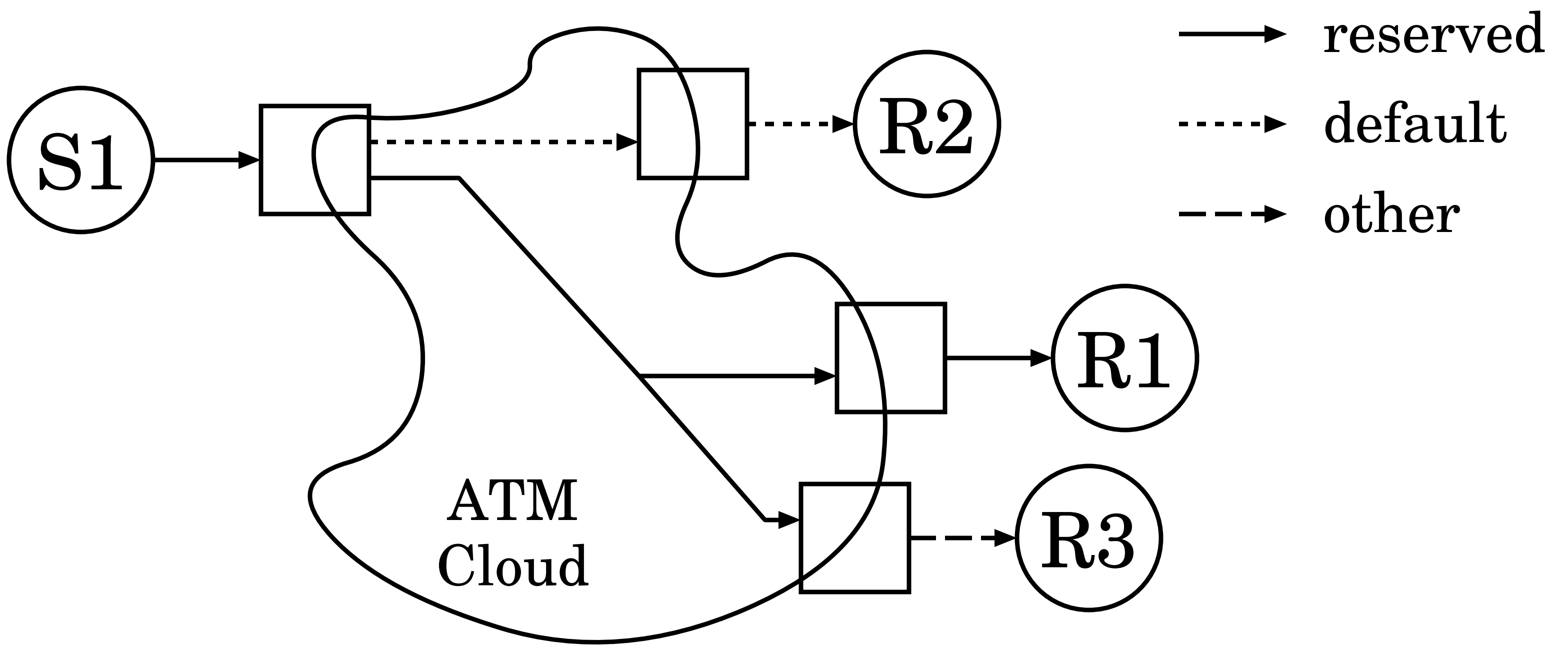
1. Uses potentially a large number of VCs

2. Wastes potentially a large amount of bandwidth

but, ...

1. Customers can get exactly what they want

2. Limited heterogeneity - no more than 2 VCs

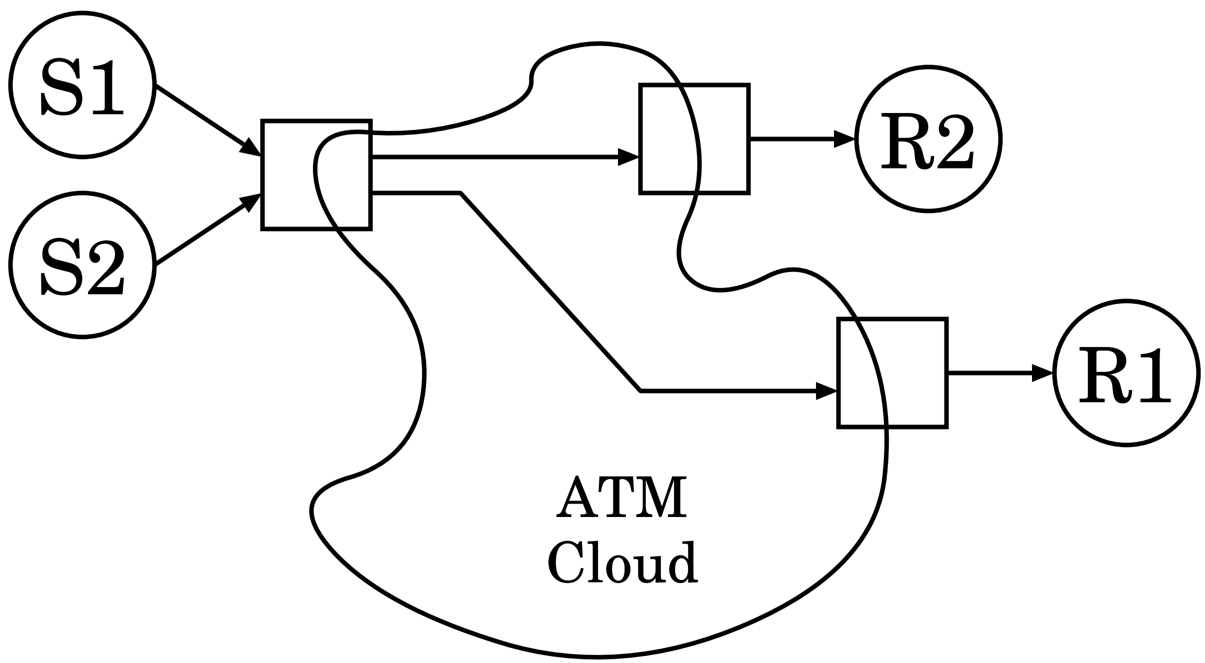


2 copies of each packet are sent

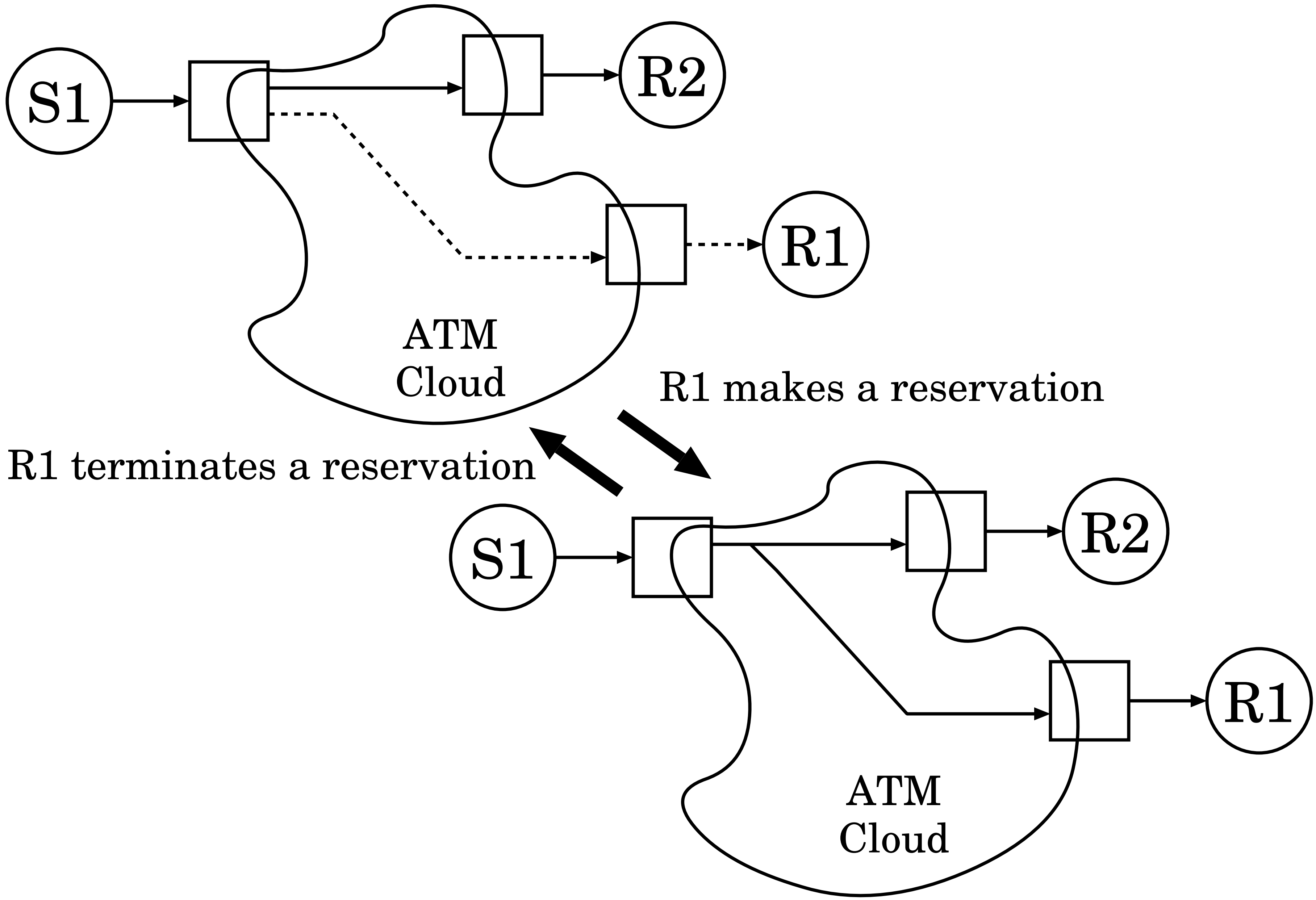
R2 can share VC with other sessions traffic

Dynamic heterogeneous QoS

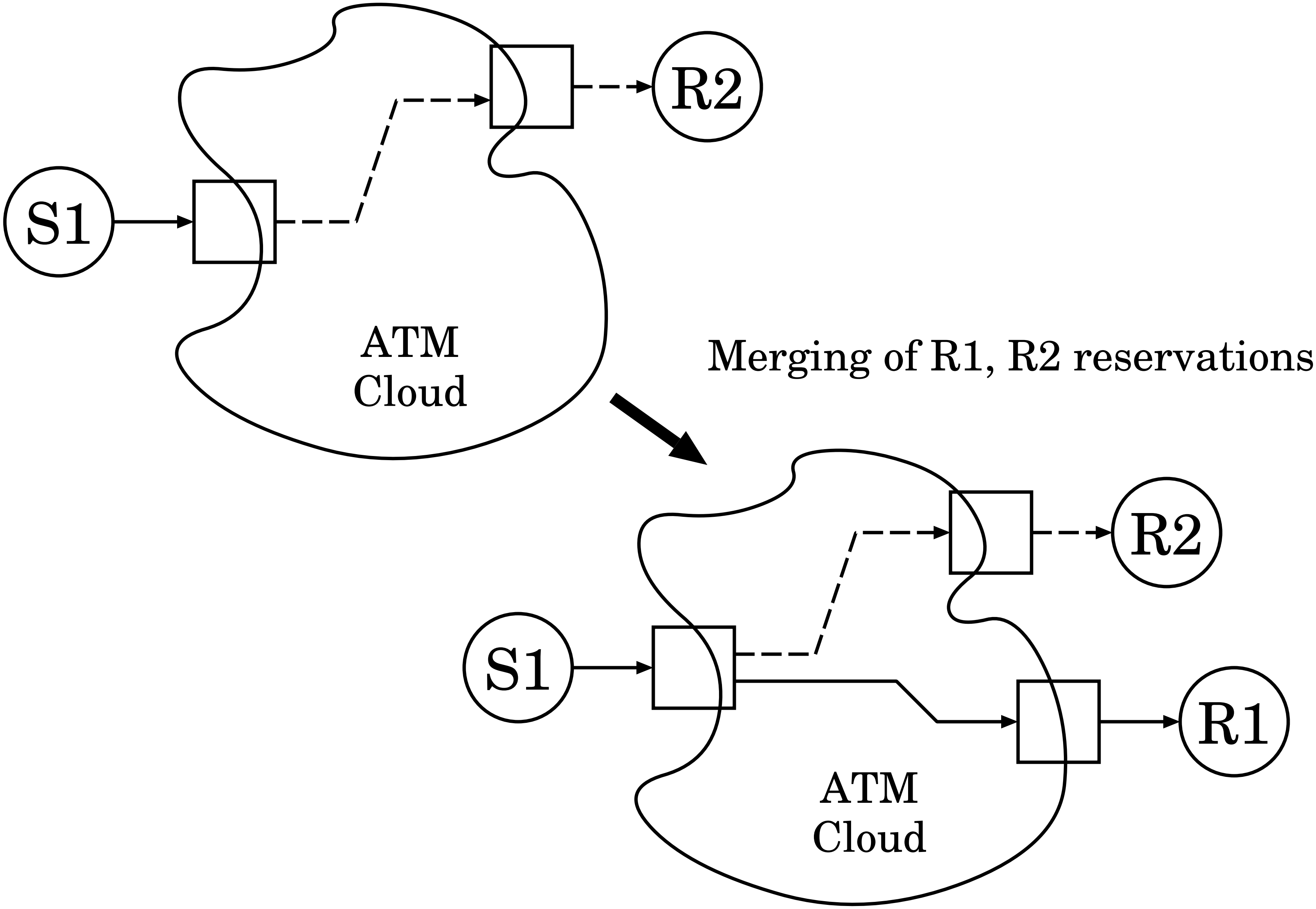
1. Receiver makes a reservation (e.g. default to QoS)
2. Receiver changes reservation (e.g. QoS to default)
3. Merging



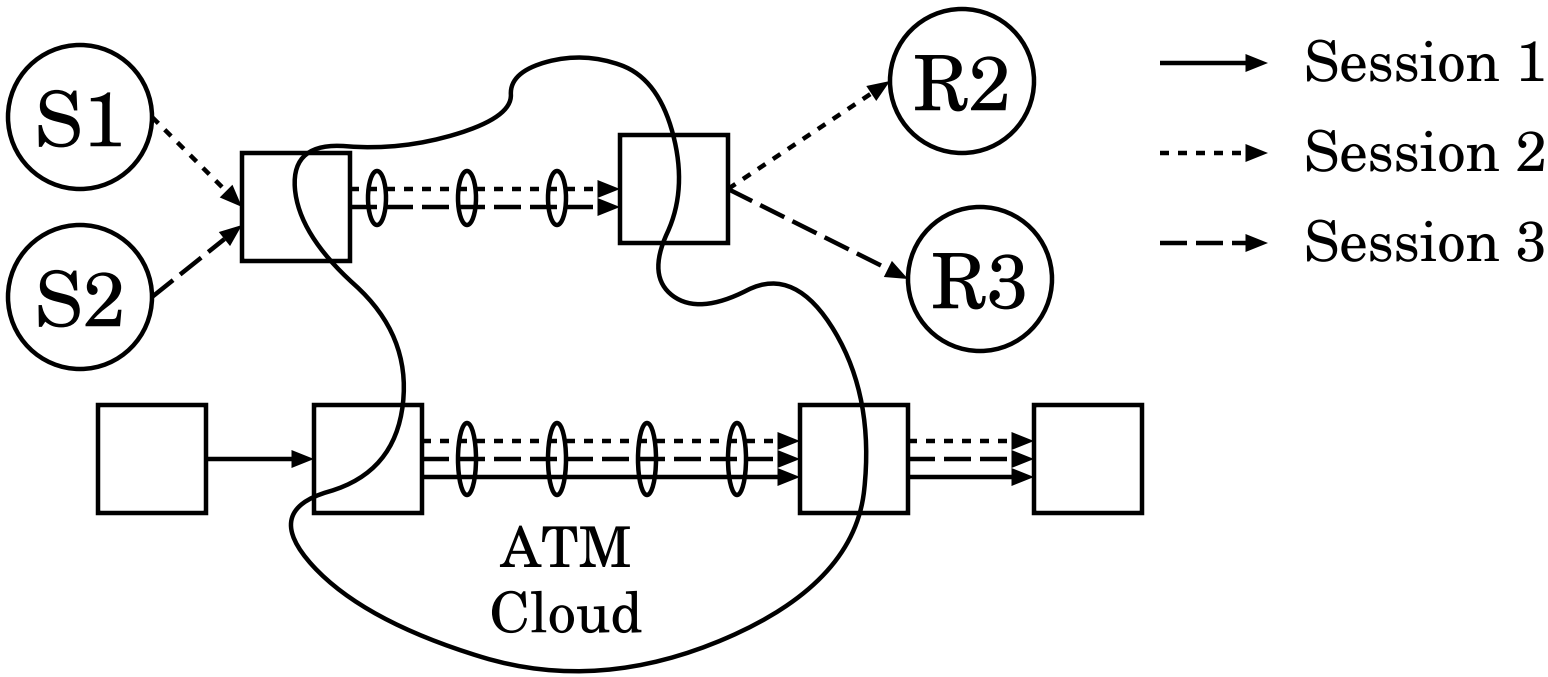
Dynamic QoS - reservation change



Dynamic QoS - Merging



4. Aggregation - small number of VCs



VC can be shared with other sessions traffic

Limitations of aggregation

1. point-to-point vs. point-to-multipoint
   a. point-to-point wastes bandwidth
   b. point-to-multipoint makes multiplexing difficult
2. choosing VC QoS and placement hard

but, ...

1. No VC latency
2. Heterogeneity problem is reduced to a solved problem
3. Dynamic QoS is solved

Outline

- Motivation
- Data VCs
- Control VCs
- Miscellaneous issues

Control VCs

- Share with data VC
- Single VC per IS flow
- Multiplexed point-to-multipoint VCs
- Multiplexed point-to-point VCs

Control and data share VC

Advantages

- No wasted bandwidth
- No extra VCs
- No signalling latency

Disadvantages

- Non-conforming traffic may cause control packet drops

Possible to give priority within a VC for control packets? CLP?

Separate Control VC per IS flow

Advantages

- Protected control VC
- Simplicity - always create an extra VC

Disadvantages

- Call setup failure - fall back on other schemes
- Signalling latency

Good for hosts

Multiplexed point-to-multipoint control VCs

Advantages

- Multiplexing saves VCs
- Often no signalling latency

Disadvantages

- Changes in egresses can cause problems
- Additional complexity can deal with above problem

Possible for network core

Multiplexed point-to-point control VCs

Advantages

- Multiplexing saves VCs
- Rarely signalling latency
- Possible to use reverse VC

Disadvantages

- Wastes bandwidth
- Choosing QoS (small) issue

Good for network core

Outline

- Motivation
- Data VCs
- Control VCs
- → Miscellaneous issues

RFC 1755 timeouts

RFC 1755 currently recommends that idle VCs be torn down after a default 20 minutes and should be configurable to “infinite”

QoS VCs should be set up with “infinite” inactivity timers

Timers for changing QoS

Currently RSVP attempts to change QoS as fast as requests are made

For subnets with a large signalling latency such as ATM, there should be a minimum time to wait before the QoS can change

QoS for control VCs

QoS for multiplexed sessions depends on (unknown) level of multiplexing

Could create additional control VCs as needed

Best Effort VC for nonconforming traffic

Mostly a QoS translation issue, but BE VC can be used for control messages

"Permanent" control VCs or on-demand

"Permanent" VCs (PVC or SVC) tie up resources, but resources are there when needed

Reverse channel issue

- point-to-point data VCs
  - reverse for control
  - reverse for data
- point-to-point control VCs
  - reverse for control
- slight interoperability issue
  - worst case - reverse VC wasted

Future

A prototype will be ready soon

ATM UNI 4.x

QoS translation

Aggregation