

# Results of The SURFnet4 Management Project 1997

B.J. van Beijnum, A.Pras, R.A.M. Sprenkels, B.D. van der Waaij

Telematics Systems and Services Management Group

Centre for Telematics and Information Technology

University of Twente

web: <http://wwwsnmp.cs.utwente.nl/nm/>

<http://wwwctit.cs.utwente.nl/>

P.O. Box 217, 7500 AE Enschede, The Netherlands

phone: +31 53 4893779

fax: +31 53 4894524

**Abstract:** This technical report bundles the five reports that were written for two SURFnet4 projects in 1997. Each report discusses a different research aspect of the SURFnet4 ATM network. The topics are 1) the Architecture of a ATM PVC Management Tool, 2) Reporting on the Use of the ATM Network, 3) SVCs in a Production Environment, 4) Quality of Service Measures, 5) Operation and Maintenance (OAM).

# Table of contents

<b>Introduction .....</b>	<b>2</b>
<b>1 ATM Management Tool Architecture .....</b>	<b>3</b>
1.1 Evaluation of the 1996 Prototype 3	
1.2 Tool Design 3	
1.3 Implementation Work plan 7	
<b>2 Report On Network Use .....</b>	<b>9</b>
2.1 Reports 9	
2.2 Conclusions 15	
<b>3 SVCs In A Production Environment .....</b>	<b>16</b>
3.1 Solutions to Policy Control 19	
3.2 State of the Art in Policy Control 23	
3.3 References 25	
<b>4 QoS Measures .....</b>	<b>26</b>
4.1 Introduction 26	
4.2 Definition of ATM Layer QoS Measures 27	
4.3 QoS Measurement Methods 36	
4.4 Examples of QoS Measurement Scenarios 41	
4.5 Conclusions 43	
4.6 References 44	
<b>5 Operation and Maintenance .....</b>	<b>45</b>
5.1 F4 and F5 OAM Flows 45	
5.2 OAM functions 45	
5.3 Support for OAM functions in standards and ATM equipment 47	
5.4 References 47	

This technical report bundles the five reports that were written for two SURFnet4 projects in 1997. The binding factor for the reports is that each reports treats a different research aspect of the SURFnet4 ATM network. Each of the five reports is covered in a separate section of this technical report:

### **Section 1: ATM Management Tool Architecture**

This section presents the architecture of the SURFNet ATM PVC management tool. First the tool that was written for the 1996 SURFnet ATM management project is analyzed. The results of this analysis then serve as the input to the design of a new prototype ATM management tool. Finally the work plan for the implementation of this new tool is presented.

### **Section 2: Report On Network Use**

This section specifies the information needs of SURFnet bv to manage PVCs in the SURFnet ATM network. Because the actual control of the PVCs is done by PTT Telecom, all the information that is needed comes from reports provided by PTT Telecom to SURFnet bv. Areas of interest for PVC management include the current PVC configuration, throughput, availability, reported network problems and the connector panel configuration.

### **Section 3: SVCs In A Production Environment**

This section examines the consequences of the introduction of an ATM SVC service in a production ATM network environment, in particular the consequences of the ability of the user to claim network resources (by setting up SVCs with a guaranteed QoS). A study of the current State of the Art in this area is included.

### **Section 4: QoS Measures**

ATM networks are designed to support Quality of Service (QoS). The agreed QoS contracts between service providers and their customers must be verified; QoS measurements are needed for that. Examples of situations where this is the case include the contract between PTT Telecom and SURFnet bv, and contracts between SURFnet bv and its connected institutions.

For such measurements to be useful it is required to have precise definitions of each QoS measure, and to have measurement methods to determine the values of these measures for a given connection. In this deliverable, we consider the definition of QoS measures and measurements methods to determine their values.

### **Section 5: Operation And Maintenance**

This section describes the functions of Operation And Maintenance (OAM) cells in an ATM network. The two types of OAM flows (The F4 and F5 flows) are examined, as are the functions that can be performed using those OAM flows. Furthermore, current support for OAM in standards and in available ATM equipment is examined.

### 1 Introduction

This section presents the architecture of the SURFNet ATM PVC management tool. First the tool that was written for the 1996 SURFnet ATM management project is analysed. The results of this analysis then serve as the input to the design of a new prototype ATM management tool. Finally the work plan for the implementation of this new tool is presented.

---

### 2 Evaluation of the 1996 Prototype

This section evaluates the tool that was written for the 1996 SURFnet ATM Management project [TOOL96].

- Web Based Management paradigm

The Web Based Management (WBM) paradigm that was used for the 1996 prototype was well received by various parties interested in ATM management (amongst others at the 1997 JENC8 conference). The use of a web browser for Customer Network Management purposes provides the user of the management tool with a familiar user interface to network management information. Also web browsers are already available to most potential users of the tool.

- User Interface Design

The design of the user interface needs to be improved on some points. Currently the user frequently needs to retype the same information, e.g. to specify a time interval of interest for statistical information. Also the order in which different pieces of information need to be supplied is not always the same.

- Data Base Implementation

Currently the database of statistical information is implemented as a set of formatted text files. The software to access and maintain these files is currently proprietary, written special for this tool. This adds to the overall complexity of the software: code to manage the proprietary database must be designed, implemented and debugged. It is better to use standard solutions for standard functionality like a Database Management System (DBMS).

- Choice of Implementation Language

For the 1996 prototype Tool Command Language (TCL) was chosen for the implementation language [TCL]. It turned out that Tcl has some properties that make it less suited for building larger applications. These limitations include limited support for arranging source code into different module files, not being object oriented, and there is no notion of a compile time syntax check. The latter makes it difficult to produce syntax error free code, and results in frequent crashes of the prototype software.

- Generation of HTML code

The generation of HTML code has proven not to be very straightforward, and should therefore be avoided if possible.

---

### 3 Tool Design

This section presents the design of the new version of the ATM management tool. First the most important design decisions are listed and motivated, then the actual design is presented.

## 3.1 Design Decisions

- Web Based Management approach

The fact that a standard web browser is sufficient to access ATM management information proved to be one of the strong points of the 1996 prototype tool. Also for the redesign of the tool this approach will be taken. This means that the overall system will use a web browser and the HTTP protocol to access a HTTP server.

- Client server architecture

The management tool architecture will be a client server architecture. this allows the client to do some additional processing on the management information it sends and receives. The user interface will be implemented as a part of the client application, and the server is relieved of the task of formatting the information to be able to present it to the user.

- Java as implementation language for the client

At this time java is the only language available to be used to build applications that can run inside a web browser.

- Java as the implementation language for the server

There are several reasons to choose java as the implementation language for the server application:

- Minimize complexity

For the client application java must already be used, so choosing java for the server also decreases the complexity of the software development for the full project.

- Multi threading

Another is that java is multi threaded, allowing a separate process to be instantiated for every client server connection.

- Platform independence

Java code can be executed on many different platforms without any changes. This allows the server software to run on different computer platforms without any changes.

- Object Oriented

Java is an Object Oriented language. OO is a modern paradigm in software development.

- Device Driver for the ATM Switch

The core functionality of the management tool interacts with the managed ATM switch via a device driver, using an internal API that is independent of any particular method to access the actual ATM switches. This way all dependencies of a particular switch are centralized in the device driver. This approach enables the tool to manage switches of different vendors, that use either standard or non-standard mechanisms to exchange management information.

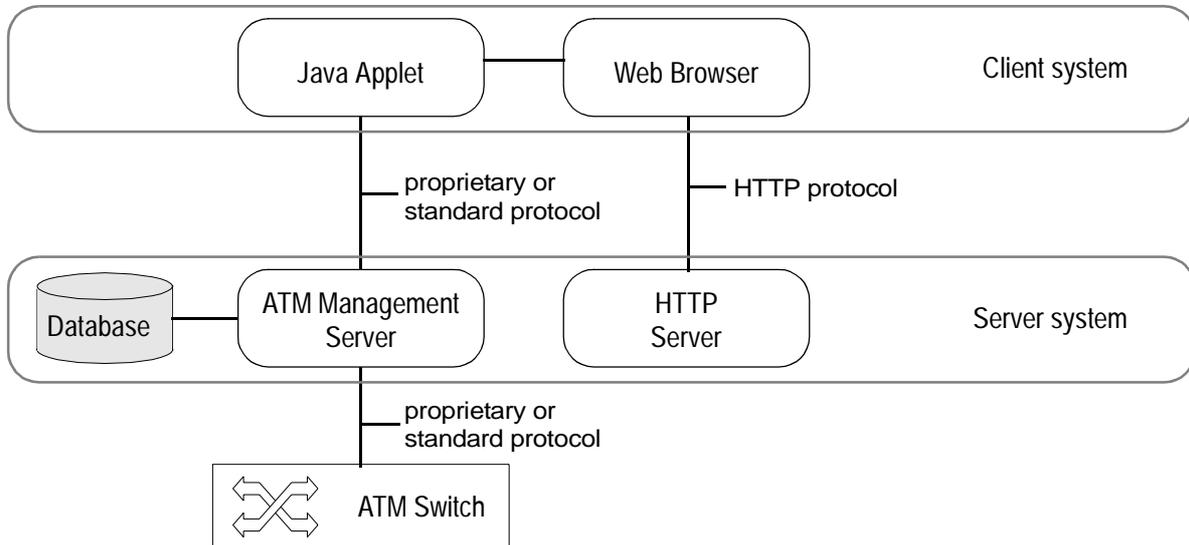
- Third party database Management system

For the database portion of the management tool an of-the-shelf database management system will be used, instead of writing proprietary database management software for this particular project. This reduces the amount of software to be developed and debugged.

## 3.2 The Design

An overview of the complete system for the ATM PVC management tool is given Figure 1.1 below:

Figure 1.1: System overview of the ATM PVC Management tool



The figure shows the components of complete system and their relationships. The client system runs a web browser, that initially contacts the HTTP server on the server system. The HTTP server then sends a java applet to the client system. This java applet finally establishes a connection with the ATM Management Server. This management server in its turn has access to the actual managed ATM switch, and to a database of configuration information and statistical information of the ATM switch.

### 3.2.1 Client to Server Communication Options

The connection between the web browser and the HTTP server uses the HTTP protocol. HTTP is the standard protocol for the exchange of HTML information between a web server and a web browser.

For the connection between the java applet and the ATM Management Server there are different options:

- Remote Method Invocation (RMI)

RMI is part of the java 1.1 specification, and allows java objects present on one system to invoke methods of objects that reside on another system. This can be used to exchange information between a java application running in a web browser and the management server running on the server machine.

- Proprietary protocol over a TCP/IP socket

The java security system allows java applets to open TCP/IP sockets to the server machine the applet originates from. The client applet could open such a socket, and use a proprietary protocol to exchange information across that socket connection with the management server.

For the initial implementation the approach of a TCP/IP socket and proprietary protocol will be used. The reason for this is that currently java 1.0 is the most widespread java implementation among web browsers. Using RMI would restrict the usability of the tool to java 1.1 capable browsers. This is expected to change over time, so a transition to java 1.1 and RMI for the tool implementation is expected in the future. For the client server communication this transition should have only minor implications.

### **3.2.2 Server to ATM Switch Communication Options**

For the connection between the ATM Management Server and the actual managed ATM switch there are also different options, depending on the implementation of the switch driver (see figure below).

For the prototype tool two standard protocols for information exchange will be implemented:

#### *SNMP and MIBs*

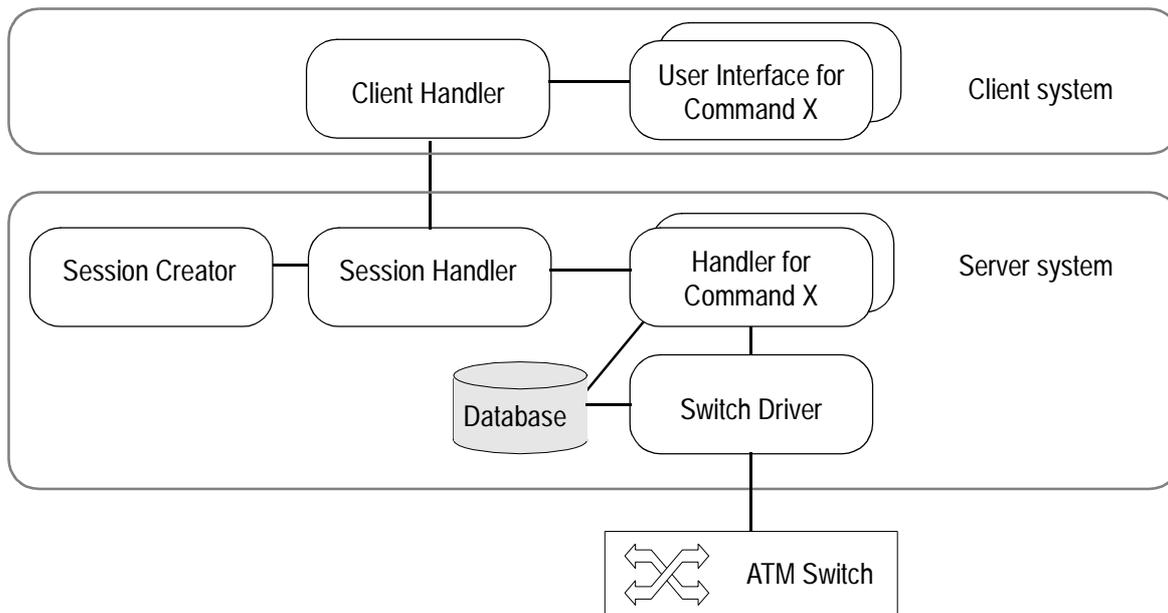
Most current ATM switches support some form of SNMP access to exchange management information, so a driver that uses SNMP will be implemented. Even though the protocol for the exchange of the information is standard, the information itself is not. There are MIB specifications by standardization organizations like the IETF and the ATM Forum; these include MIB-II (RFC1213) and the AToM MIB (RFC1695). Support for these standard MIBs will be included in the tool. Most switch vendors also have (or in some cases, only have) proprietary MIBs. Support for these MIBs will not be included in the tool initially.

#### *Email exchange of Information*

Another standard way of exchanging information is email. A driver that uses email to exchange information with an ATM switch will be implemented. The switch itself will not respond directly to the email messages, but instead the email messages will be sent to a human operator responsible for configuring the switch. This way of exchanging information with a switch is useful in situations where information needs to be exchanged across management domain borders. In such cases SNMP access across the management domain border is generally not allowed.

The structure of the java client and server application in more detail is given below:

Figure 1.2: Structure of the java client and of the server application



The way in which the client server system in the figure above comes into existence is described next.

The user establishes a connection with the server by loading the opening page for the management tool into the web browser (The user has the URL for this page, this can be a hyperlink from another page). This opening page contains a java applet.

In its initialization phase the applet opens a socket to the Java server application running on the web server. The server in its turn creates a new process to service the new client, and sends some configuration information back, including a list of commands that exists and a list of accessible ATM switches. This results in the runtime structure of the client server system given in the figure above.

---

## 4 Implementation Work plan

The design will be implemented in multiple phases. This approach enables us to gain experience with the design and to validate it, and to allow early feedback on the prototype software.

### 4.1 Phase 1: Framework and limited functionality

In this phase the framework of the tool is implemented, together with a simple management command. For this command the retrieval of some elementary system information is chosen, like the time since the system was last rebooted.

### 4.2 Phase 2: PVC Management

Of the all the possible uses of an ATM network management tool, SURFnet has assigned the highest priority to the management of PVCs. Therefore management of PVCs is the portion of

the tool that is developed in this phase. Full management of PVCs is further divided into the following sub-functions:

#### 1. PVC creation and PVC deletion

This is considered to be the basic PVC management function, and is the first function to be implemented in the tool.

#### 2. PVC overview

This function gives the user an overview of all configured PVCs in the managed ATM switch. This overview will be retrieved directly from the switch through the switch driver

#### 3. Administrative database of PVCs.

When the database function of the server system is implemented in this sub-phase the system will maintain a list of PVCs that should exist in the switch. This allows the system to compare the PVCs that actually exist in the switch against the administrated list of PVCs that should exist, and respond to any differences between the two.

### 4.3 Phase 3

In this phase remaining functions of the tool will be implemented. These functions include the gathering of statistical information from the network like the availability, transmission errors, and throughput.

## References

- [ATMF] The ATM Forum, <http://www.atmforum.com>
- [ATOM] RFC1695 The AToM MIB (for ATM PVC management)
- [IETF] Internet Engineering Task Force, <http://www.ietf.org>
- [MIB-II] RFC1213 "Management Information Base for Network Management of TCP/IP-based internets: MIB-II"
- [SNMP] RFC1215 "A Convention for Defining Traps for use with the SNMP"  
RFC1213 "Management Information Base for Network Management of TCP/IP-based internets: MIB-II"  
RFC1157 "A Simple Network Management Protocol (SNMP)"  
RFC1155 "Structure and Identification of Management Information for TCP/IP-based Internets"
- [JAVA] The Java homepage, <http://java.sun.com/>
- [TCL] Tool Command Language, <http://www.sco.com/Technology/tcl/Tcl.html>
- [TOOL96] 1996 prototype Web Based Management Tool, <http://atlas.cs.utwente.nl:8890>, user is "user", password is "password".

### 1 Introduction

The purpose of this section is to specify the information needs of SURFnet bv to manage PVCs in the SURFnet ATM network. Because the actual control of the PVCs is done by PTT Telecom, all the information that is needed comes from reports provided by PTT Telecom to SURFnet bv.

The following areas are of interest to manage PVCs:

- The current PVC configuration
- The throughput
- The availability
- The reported network problems
- The connector panel configuration

With information on these areas SURFnet bv can manage the PVCs in the SURFnet ATM network. Besides the management issue it is also useful for tactical and strategic management purposes. Each area is described in one or more reports, which must be generated at a monthly interval by PTT Telecom.

---

### 2 Reports

In this chapter each report that is needed from PTT Telecom is described in more detail. The reports must be provided in a monthly interval, and in a human readable form, preferable HTML or postscript.

Before the reports are discussed, some naming definitions are presented.

#### 2.1 Naming conventions

To avoid misunderstandings, we specify the following naming conventions that must be used in the reports coming from PTT Telecom:

- *SURFnet client*  
A SURFnet client is identified with its domain name: <domain\_name>. For example: UT-wente, CWI, TUE, etc.
- *SURFnet switch*  
A SURFnet switch is identified with its domain name. If there are several switches at the institute, it can be followed by an extension: <domain\_name> or <domain\_name>-<ext>. For example SARA, UTwente-test.  
Currently, all the SURFnet switches are GDC switches. For the purpose of this report, the JAMES network is considered to be a SURFnet client, and therefore has also SURFnet switches.
- *PTT Switch*  
A PTT switch is a switch that is located within the public PTT ATM network and has no connection with SURFnet clients. These switches can not be controlled by SURFnet bv.
- *SURFnet interface*  
A SURFnet interface is identified with the name of the ATM-switch it belongs to. Next an iden-

tification of the interface type (e.g. ST). Finally two numbers identifying the two switch-connectors, one for each direction of this interface. The numbers are paired: an odd and an even number (e.g. 13/14): <SURFNet\_switch\_name>/<interface-type>-<odd-number>/<even-number>. For example SARA/ST-1/2, UTwente-test/ST-13/14.

- *PVC*

A PVC is identified with the two locations of the SURFnet clients it is connected to followed by a number: <location>-<location>/ATM<number>. For example asd-es/atm1. Notice that the PVC is numbered on a client to client basis, and not on a switch to switch basis. If there are two PVC between Amsterdam an Enschede, and the PVC is connected to two ATM switches at Enschede and one at Amsterdam, they would be called: asd-es/atm1 and asd-es/atm2.

Due to these conventions, a SURFnet client, a switch, an interface and a PVC have a unique SURFnet-wide name.

## 2.2 PVC Configuration Overview

The PVC Configuration Overview report contains an overview of all currently configured PVC connections<sup>1</sup> in the SURFnet ATM network. The report is created as a table, each row in this table contains the following items:

- The name of the PVC (see paragraph 2.1)
- The date and time the PVC has been set up.
- The name of the SURFnet A-interface, this is (arbitrarily chosen) one of the two interfaces of the PVC.
- The VPI and VCI (if applicable) of the A-interface.
- The name of the SURFnet B-interface, this is the opposite interface of the SURFnet A-interface.
- The VPI and VCI (if applicable) of the B-interface.
- For the direction A to B:
  - Peak Cell Rate (PCR),
  - Sustainable Cell Rate (SCR),
  - Cell Delay Variation Tolerance (CDVT),
  - Burst Tolerance (BT),
  - Service category for the PVC (UBR, CBR, VBR-real-time, VBR-non-real-time, ABR),
  - Type of policing action, if any (none, tag violating cells, drop violating cells),
- For the direction B to A:
  - Peak Cell Rate (PCR),
  - Sustainable Cell Rate (SCR),
  - Cell Delay Variation Tolerance (CDVT),
  - Burst Tolerance (BT),
  - Service category for the PVC (UBR, CBR, VBR-real-time, VBR-non-real-time, ABR),
  - Type of policing action, if any (none, tag violating cells, drop violating cells),

---

1. In this deliverable only point to point PVCs are considered

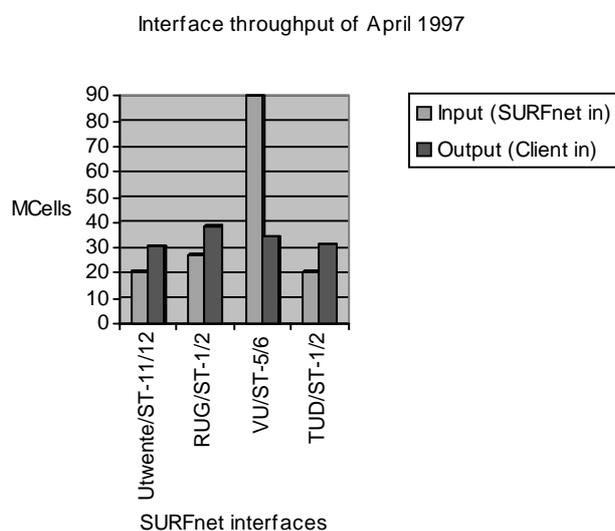
## 2.3 Throughput overviews

The throughput report consists of four different overviews. One describes the throughput per SURFnet interface. One for the PVC throughput and one describing the throughput of the entire SURFnet ATM network. The fourth overview describes the number of dropped cells for each PVC.

### 2.3.1 Interface throughput

For each SURFnet interface the throughput must be measured. This is performed for both directions. The input direction describes the total amount of traffic from the SURFnet client into SURFnet ATM network. The output direction describes the traffic the SURFnet client is receiving from SURFnet ATM network. The throughput is measured in megacells per month, but this can be extended into gigacells if applicable.

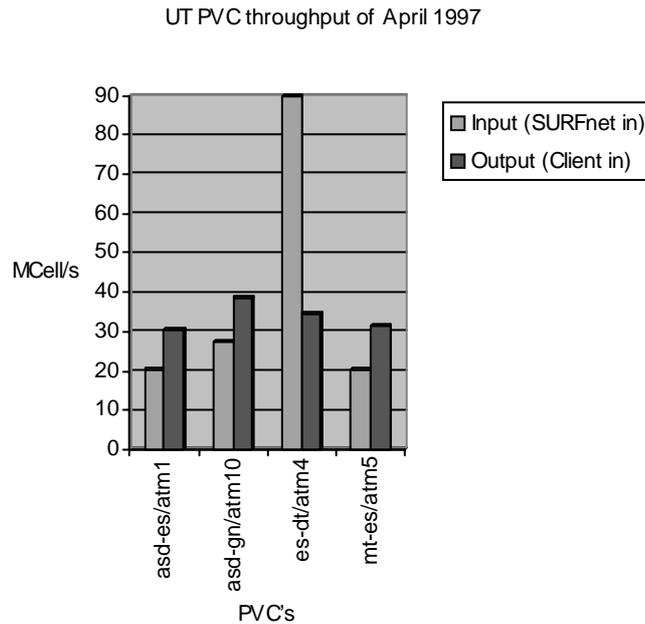
Figure 2.1: Example of Interface throughput



### 2.3.2 PVC throughput

For each SURFnet client of SURFnet by a PVC throughput overview is generated. This overview contains for each PVC of the SURFnet client, referenced by its name as described in paragraph 2.2, its throughput. The throughput is specified for both directions. The input direction describes the traffic from the SURFnet client into SURFnet ATM network. The output direction describes the traffic the SURFnet client is receiving from SURFnet ATM network. The throughput is measured in megacells, but this can be extended into gigacells if applicable.

Figure 2.2: Example of PVC throughput

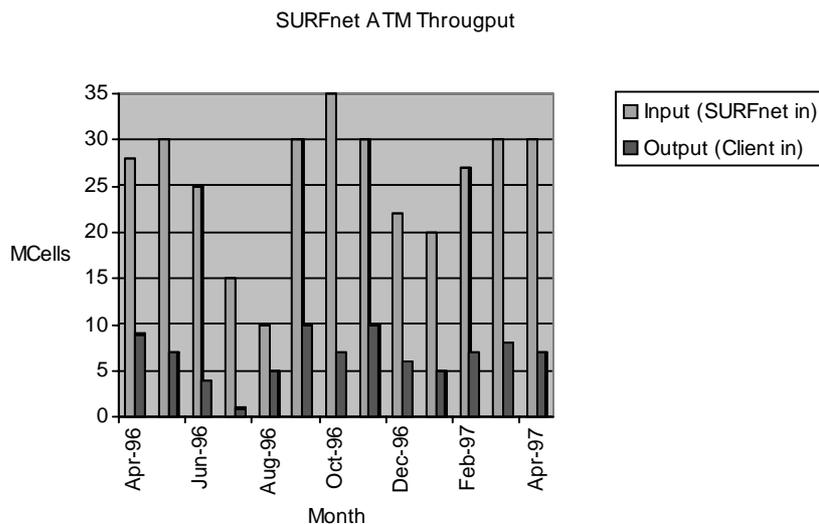


### 2.3.3 SURFnet ATM network throughput

This overview contains information about the total number of transmitted cells in the SURFnet ATM network. For each month all incoming cells from the SURFnet clients into the SURFnet ATM network are counted, also the outgoing cells are counted. The number of cells are specified in megacells, this can be extended if applicable.

The overview will always contain the data of 13 months, using a sliding window mechanism over the past 13 months. If the data of the previous months are not available, a value of 0 will be reported.

Figure 2.3: Example of the ATM network throughput



### 2.3.4 PVC dropped cells

For each SURFnet client of SURFnet by a "PVC dropped cells" overview is generated. This overview specifies for each PVC of the SURFnet client, referenced by its name as described in paragraph 2.2, four different counters:

- The number of cells that have violated the traffic contract of this PVC.
- The number of cells that are dropped due to violating the traffic contract of this PVC<sup>1</sup>.
- The number of cells that are dropped due to congestion in the network.
- The number of cells that are dropped due to line errors, etc.

The four counters are specified for each direction of the entire PVC. This means that the counters contain the sum of the counters in the network. For example: there are cells dropped due to congestion at several SURFnet and PTT ATM switches in the SURFnet ATM network. In this situation the counter specifying the number of dropped cells due to congestion will contain the sum of all the dropped cells in the network.

For each counter also the percentage relative to the total number of received cells is generated. The percentage must have a decimal fraction of 3 digits.

Figure 2.4: *Example of PVC dropped cells*

		violated		dropped					
				contract		congestion		line errors	
PVC	direction	#	%	#	%	#	%	#	%
asd-es/atm1	asd-es	150	0.015	145	0.014	5	0.001	0	0.000
	es-asd	100	0.010	80	0.008	0	0.000	20	0.002
ut-nm/atm25	ut-nm	2365	0.230	2000	0.200	350	0.028	15	0.002
	nm-ut	0	0.000	0	0.000	0	0.000	0	0.000

## 2.4 Availability

The availability report contains two overviews. The first overview describes the availability per SURFnet interface. The second overview is a matrix, specifying the availability per (possible) connection between two SURFnet interfaces.

### 2.4.1 Availability per SURFnet interface

The availability overview per SURFnet interface is created as a table, each row in this table contains the following items:

- The name of the SURFnet interface (see paragraph 2.1).
- The number of downs that lasted longer than 1 hour.
- The number of downs that lasted between 10 minutes and 1 hour.
- The number of downs that lasted at maximum 10 minutes.

---

1. Notice that this counter will have the same value as the number of cells that violate the traffic contract if the contract specifies that all cells that violate the traffic contract must be dropped.

- The time this interface was down at working hours (08:00 - 20:00)
- The percentage this interface was down during the working hours (08:00 - 20:00). The percentage has must have a decimal fraction of 2 digits.
- The time this interface was down at all hours (00:00 - 24:00)
- The percentage this interface was down during all hours (00:00 - 24:00). The percentage has must have a decimal fraction of 2 digits.

Figure 2.5: Example of the overview

SURFnet Interface	down >1hr	down 1h -10m	down 0 -10min	down working	% working	down all	% all
UTwente/ST-11/12	0	1	0	00:00:00	100.00%	00:12:34	99.15%
SARA/ST-1/2	0	0	2	00:00:00	100.00%	00:00:02	99.86%

## 2.4.2 Availability per connection

The availability overview per connection is presented as a matrix. The rows and the columns of this matrix contain all the SURFnet interfaces. The crosspoint in the matrix specifies the percentage that this connection was available during the last month. A connection lies between two SURFnet interfaces, similar to a PVC or SVCs. The matrix gives an overview of the availability of all possible connections. The percentage has must have a decimal fraction of 2 digits.

Figure 2.6: Example of the report

to /from	UTwente/ST-11/12	UTwente/ST-13/14	SARA/ST-1/2	TUD/ST-5/6
UTwente/ST-11/12	x	99.65%	100.00%	90.00%
UTwente/ST-13/14	99.65%	x	97.23%	99.11%
SARA/ST-1/2	100.00%	97.23%	x	95.98%
TUD/ST-5/6	90.00%	99.11%	95.98%	x

## 2.5 PTT Telecom Helpdesk Summary

The ATM helpdesk of PTT Telecom will keep a log of all reported problems in the SURFnet ATM network. These problems can be reported by SURFnet bv or directly by a SURFnet client. PTT Telecom will provide SURFnet bv with a report, containing a table, each row in this table contains the following items:

- The institute who reported the problem.
- PTT Telecom call number.
- The date the problem is reported.
- The time the problem is reported.
- The time elapsed before the problem was fixed.
- A description of the problem.
- Found solution.

Figure 2.7: *Example of the report*

Reported by	call #	date	time	fix to time	problem description	solution
UTwente	?	10/05/1997	10:12	00:10	No cells are arriving at Enschede any more	Cable was disconnected

## 2.6 Connector Panel Configuration

Between the SURFnet ATM network and the SURFnet client ATM network a management interface exist. This interface is physically identified with a panel, called the connector panel. Via this panel the connections can be made between the GDC switch (the border switches of the SURFnet ATM network) and the institution ATM switches.

This panel and its connections must be described in detail to avoid misunderstanding between the SURFnet client, SURFnet bv and the PTT Telecom. Therefore a report from the PTT Telecom is necessary that specifies:

- The name of the SURFnet ATM switch (see paragraph 2.1).
- The SURFnet client of this connector panel configuration overview.
- The SURFnet interface name.
- The interface type (e.g. STM1, TAXI).
- The mode of the fiber used for this interface (mono or multi).
- The GDC interface this SURFnet interface is connected to.

Figure 2.8: *Example of the report*

SURFnet Switch	SURFnet client	SURFnet interface	Int. type	fiber mode	connected to GDC interface
UTwente-test	UTwente	UTwente-test/ST-13/14	STM1	multi	?
SARA	SARA	SARA/ST-1/2	TAXI	multi	?

---

## 3 Conclusions

If SURFnet bv receives every month the reports as described in section 2 of this deliverable from PTT Telecom, SURFnet bv has enough information to perform tactical and strategical management on PVCs in the SURFnet ATM network.

### 1 Problem Description

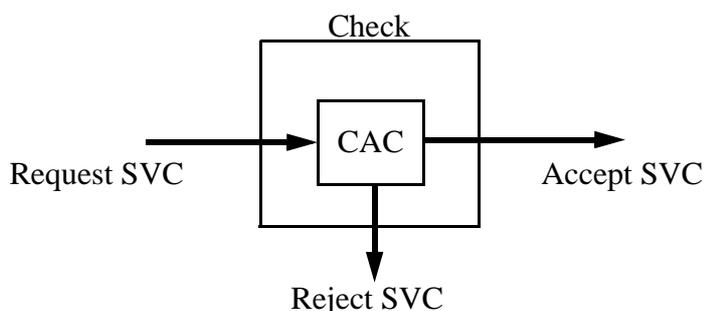
Section 3 examines the consequences of the introduction of an ATM SVC service in a production ATM network environment, in particular the consequences of the ability of the user to claim network resources (by setting up SVCs with a guaranteed QoS). A study of the current State of the Art in this area is included.

#### 1.1 The SVC Service

This section provides a short overview of those aspects of the ATM SVC service that are of importance to be able to discuss ATM SVC Policy Control.

- The SVC service enables the end user to set up ATM connections on demand. At this point a distinction must be made between users and endstations. In this document the term *user* is used to identify a human being. The term *endstation* specifies a computer system that is at the end of an ATM link. An endstation can be referenced by its ATM address. Therefore a SVC setup is always initiated by an end-user, and the resulting SVC connects two or more endstations.
- a SVC claims network resources. These resources include bandwidth on the ATM links in the network, available VPI and VCI values, cell buffer space in the ATM switches etc. These resources are all shared and finite.
- When a user requests a new SVC from an ATM network, the network will check if there are enough network resources available to accommodate that SVC. If this is the case, the SVC setup request is accepted, otherwise it is rejected. This function is known in current ATM standards as Call Admission Control (CAC).

Figure 3.1: Logical view of the Call Admission Control function of an ATM network



#### 1.2 The Need for an SVC Policy

With ATM switches currently on the market it is possible to build an ATM network that provides an SVC service. This SVC service will do Call Admission Control, so a request for a SVC will only succeed if the network has sufficient resources to meet the Quality of Service guarantees for that SVC.

The main topic of this document is that having only a CAC function in the network is not sufficient for a production ATM network. Additional mechanisms are needed to control the accep-

tance of SVCs, and thus how the scarce network resources are divided among the users. We will refer to these additional mechanisms as the SVC acceptance policy or SVC Policy for short.

An ATM SVC network with only CAC (which can be build using ATM equipment available today) can be regarded as having a simple “first come first served” SVC Policy. All users have equal chances of claiming and using network resources, on a first come first served basis. A user can at any point in time claim all remaining network resources and keep them occupied for an arbitrary long period of time. In a production network, where users actually pay for the network service, such a policy will not be acceptable.

Consider as an example a 155 Mbit/sec university ATM network that serves students as well as staff members. Say the network is completely empty on sunday evening. A student requests a 155Mb CBR SVC. The request is accepted by the CAC, because the network can currently support the connection. The student does not release the connection. On monday morning, no student, staff member or professor is able to set up an SVC, because all resources are still in use by the student. This is clearly an unwanted situation. Other mechanisms are needed to support more detailed and refined policies than the simple “first come first served” policy provided by CAC.

### 1.2.1 SVCs compared to PVCs

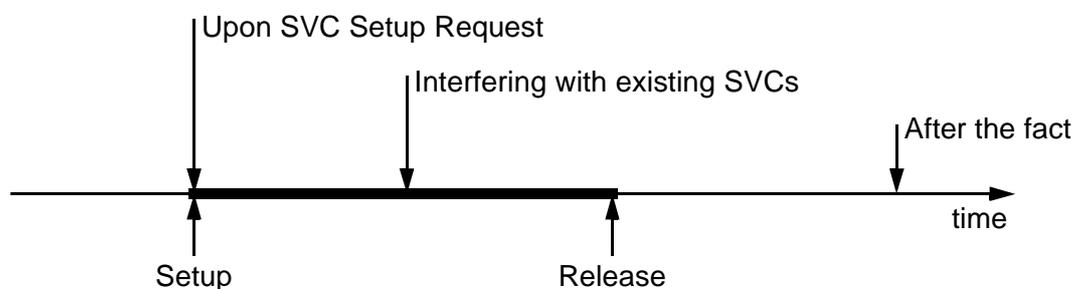
Management (instead of signalling) can also be used to create connections in ATM networks. In ATM terminology such connections are called PVCs. These PVCs claim network resources, just like their SVC counterparts, so in principle the same Policy Control issues as in the SVC case apply.

The key difference between PVCs and SVCs is that PVCs are not actually created by the end-user; they are created by a manager instead. Thus, the decision weather to accept a PVC is made by that manager also. The manager must check if the required resources are available, and if the PVC complies with the PVC acceptance policy. As a result, for PVCs Policy Control is automatically present in the system; the PVC Policy is the set of rules used by the manager to accept or deny a user request for a PVC.

## 1.3 The Location of Policy Control in the SVC Lifecycle

An SVC is setup at some point in time, then exists for a period of time, and finally the SVC is released. We will call this the Lifecycle of the SVC. In this lifecycle we identify three different sections where Policy Control can be applied.

Figure 3.2: Lifecycle of an ATM SVC



Policy Control can be applied in the SVC lifecycle upon the SVC Setup Request, while the SVC is active, or after the SVC has been released. Each of these will be discussed in a separate section below.

### **1.3.1 Checking Upon SVC Setup Request**

When Policy Control is to be performed the moment the network receives a SVC setup request, two things have to be checked. These are the normal CAC check and the Policy Control check. Because the ATM network sets up a SVC on demand with a sub-second setup time, both checks must be performed in that same time-frame, and therefore in an automated manner.

#### *1.3.1.1 Connection Admission Control*

Connection Admission Control (CAC) ensures that a new connection is only accepted if the network has sufficient resource to accommodate that particular new connection, without degrading the service level for the already existing connections. Resource of interest in this respect is e.g. bandwidth, cell buffer space in switches, etc. Standards for ATM CAC already exist [UNI31] and current ATM switches support them.

#### *1.3.1.2 Policy Control*

The task of Policy Control is to divide the available network resource in a defined manner over the set of network users competing for that resource. Policy Control enforces a connection admission policy. A connection admission policy is a set of rules defining under what circumstances (other than the mere availability of the requested resource) the request for a SVC is to be accepted.

### **1.3.2 Interfering with existing SVCs**

Another way to enforce a Policy in a more ad-hoc manner is to allow a manager to interfere with existing SVCs. The manager in this case could be either a human manager or a software management application. The manager could free up resources by deleting SVCs, thereby making the free resources available for other SVCs. To do this, the manager should be able to

- get an overview of existing SVCs;
- delete a particular (unwanted) SVC;
- prevent a deleted SVC from being re-setup immediately.

For this form of Policy Control the manager would need a set of software tools that provide the functionality listed above.

### **1.3.3 Checking after the fact**

The checking after the fact method is an additional solution to the problem of controlling SVCs. It inherently cannot prevent users from claiming all of the available remaining resource; since it works only after a SVC has been setup and destroyed again. Instead, this mechanism ensures that the network users know that their usage of the network is somehow measured. These measurements can then e.g. be used to send bills to users or groups of users.

---

## 2 Solutions to Policy Control

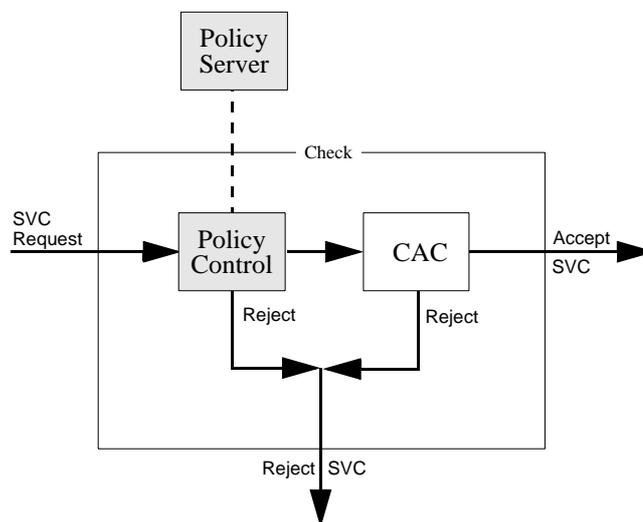
To be able to enforce a Policy, the ATM network must be extended with a Policy Control function. This chapter provides an architectural framework to define and implement such policies.

### 2.1 An Implementation Architecture for Policy Control

There are three options for the location of the Policy Control check relative in time to the CAC check; either before CAC, after CAC or in parallel with CAC.

In general a combination of information on individual users, accounting information and global network status will be needed to be able to make a Policy decision. To avoid storing copies of all this information at each point in the network where Policy decisions must be taken, a network wide Policy Server is introduced. This Policy Server will hold all the information needed to enforce the network SVC acceptance Policy, and switches will contact this server for their information. This is shown in the figure below.

Figure 3.3: A switch with policy control



## 2.2 The policy areas

Policy decisions are taken based on a set of criteria. In this section four areas of such criteria are identified. Also the architecture of a possible implementation for Policies using those criteria is presented.

### 2.2.1 Specific Users or Endstations

The decision to accept a request for a new SVC can be based on which particular human user or which particular ATM endsystem is requesting the connection. Ideally, one would like to base policies on human users rather than on ATM endsystems. It is to be expected that implementing Policies based on specific users is more difficult than implementing Policies based on specific ATM endstations. Identifying a specific human user would have to be part of the ATM signalling system.

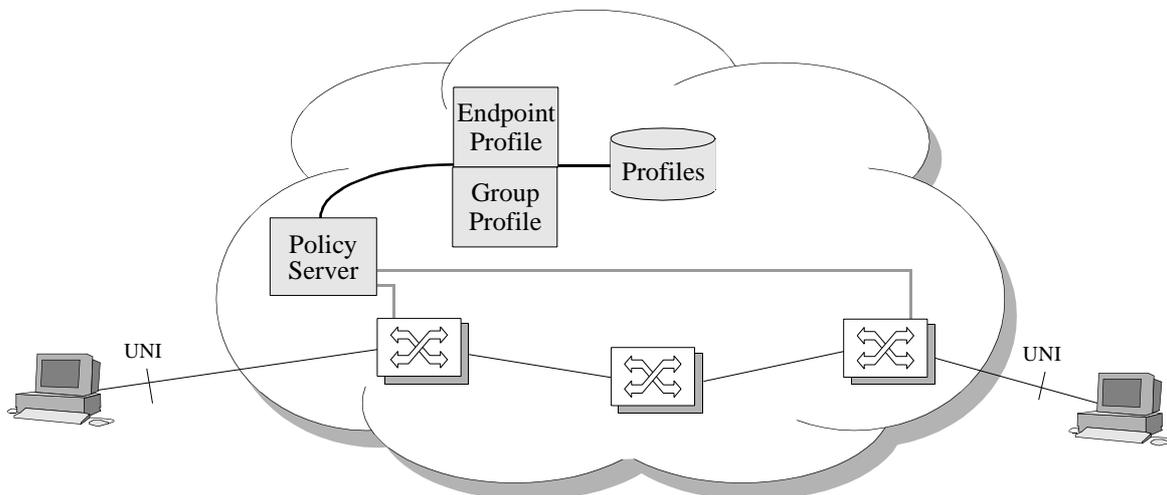
An example of a Policy based on specific users is a policy where users in the group 'students' are allowed connections up to 1 Mbit/sec, and users in the group 'staff members' are allowed connections up to 10 Mbit/sec.

To be able to make a Policy decision based on which particular endpoint or which particular user is requesting the new SVC, the Policy server needs information on the endpoints and users that exist. Each endpoint (or each user) has a certain 'profile' attached to it. This profile contains parameters for the Policy decision, e.g. maximum allowed bandwidth per connection. The ITU-T has defined an extension to the Q.2931 [Q2931] signalling system that enables the generation and transport of end station identifiers. This extension is defined in the Q.2941.1 and Q.2941.2 documents [Q2941.1] [Q2941.2].

A notion of groups is also modeled in the database of endpoints and users. This allow Policies to take into consideration that groups of similar users or endpoints exist, and to check criteria that apply for the whole group. An example of a policy that uses this when all members of the group 'students' are allowed to use only a total bandwidth of 50 Mbit/sec.

The policy server has access to a database of endpoints and their associated profiles. This is shown in figure 3 below.

Figure 3.4: Individual users policy



## 2.2.2 Remaining credits

Policy decisions can also be based on the amount of resource the requesting endpoint or the destination endpoint has used in the past. As such this criterion for Policy decisions is an extension to the Policies based on specific users or endstations of section 2.2.1.

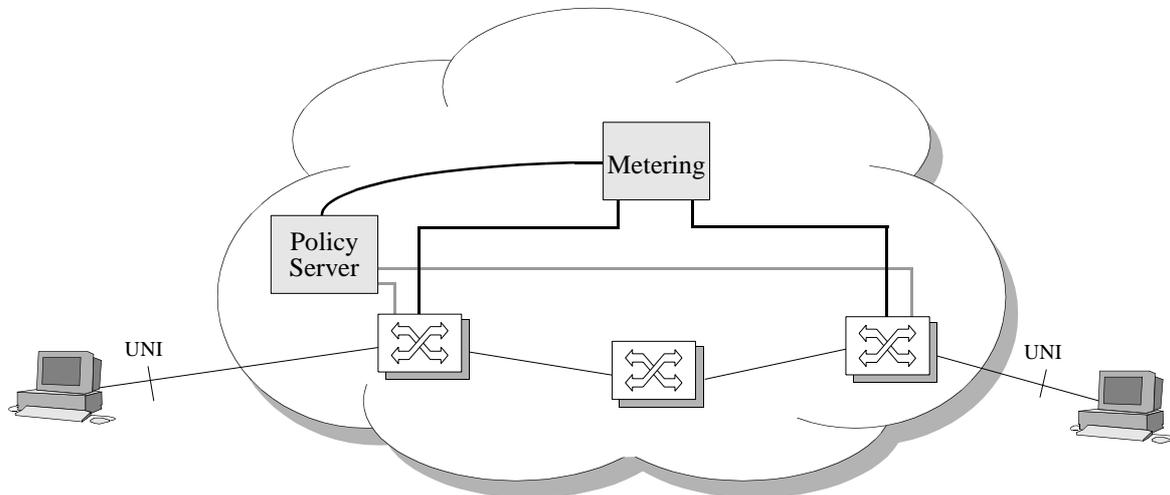
A new connection can be accepted or rejected based on how much resource this endpoint (or the group the endpoint belongs to) has already used over the last measurement period. As an example an individual endpoint could be allowed to own connections for a total of 100 Mbps-hour per month. This is e.g. a 1 Mbps connection for a period of 100 hours, or a 10 Mbps connection for 10 hours. At the start of a new period the endpoint gets additional credit for setting up SVCs. When the endpoint actually sets up SVCs that credit is debited. For this to work, the usage of network resource by each (group of) endpoint(s) needs to be administrated over time.

Note that credits are not exactly the same as accounting plus billing. With accounting plus billing the user, belonging to the endpoint, must pay for the used resources some time after he has used them. There is however no limit to how much resource a user can use in a single billing

period. The combination of accounting and billing is a psychologically restricting mechanism, where a credit based policy actually prevents additional connections to be set up once the user has used up all his credit.

To implement Policies based on usage credit, the network needs to gather usage information from the network for each endpoint or group of endpoints, as shown in the figure below:

Figure 3.5: Remaining credits policy



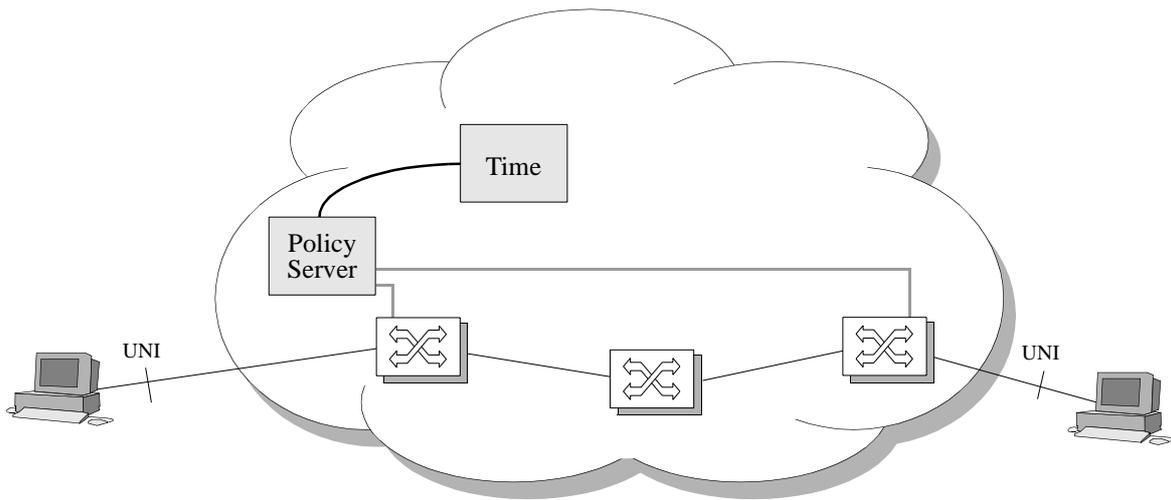
The box with caption 'Metering' in the figure gathers information about each SVC that is created in the network. This information should as a minimum include the amount of allocated resource for each SVC (e.g. bandwidth) and the timespan it existed. The (future) standards for accounting in ATM networks cover the retrieval of this information.

### 2.2.3 Current time

The current time can be used as one of the criteria for the decision to accept or reject the request for a new SVC. An example of a time dependent policy is when student group endpoints are allowed connections of 1 Mbps during business hours, and connections of 2 Mbps outside business hours.

The network Policy server needs access to the current time for time dependent policies, as shown in the figure below.

Figure 3.6: Current time policy



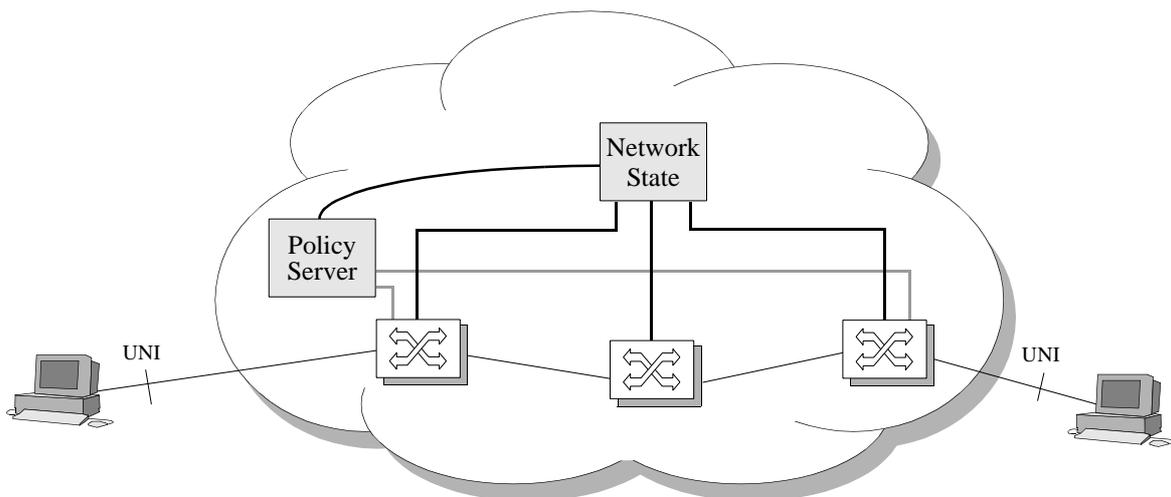
Time dependencies of the policy should be implemented in the policy server. A time dependency will always be used in conjunction with some other criterion, like the maximum allowed bandwidth per user in the example.

## 2.2.4 Current network status

The current status of the network can be taken into account when taking Policy decisions. An example of such Policy is when there are two user groups with different priority; low priority endpoints are not allowed to set up SVCs when the network is over 60% loaded, where high priority endpoints are always allowed to set up new SVCs.

To be able to take the current network status into account when taking Policy decisions, the network status must constantly be measured, and the Policy control function must have access to the network status information, as shown in the figure below.

Figure 3.7: Current network status policy



---

## 3 State of the Art in Policy Control

In this chapter an overview is given of the current state of the art in policy control technology. The first paragraph describes the RSVP world approach.

### 3.1 RSVP

In the internet world RSVP is used to control the access of endpoints to the network. When the endpoint makes a request for a connection with a certain QoS (in RSVP this is called resource reservation), RSVP carries the request through the network, using the same routing mechanism as the connection will use when it is created. At each visiting node that supports RSVP an attempt is made to allocate the requested resources and to check if the endpoint has enough administrative permissions to make this reservation. If they both succeed, the connection is made, otherwise the endpoint gets a notification that there are not enough resources for his request.

The two modules in the resource allocation mechanism are: admission control and policy control. The admission control module's responsibility is to check if this node has sufficient available resources to supply the requested QoS. This module is similar to the Call Admission Control module in an ATM network. The policy control module checks if the user has administrative permission to make this reservation. In RSVP policy control is described as:

Reservation protocols, by definition, discriminate between users, by providing some users with better service at the expense of others. Therefore, it is reasonable to expect that these protocols be accompanied by mechanisms for controlling and enforcing access policies. The IETF has already started work for the RSVP protocol in this direction [RAP].

For this paper the admission control module is of less interest, only the policy control module will be discussed.

#### 3.1.1 Policy control

The policy control mechanism of RSVP is designed to offer a very flexible mechanism to experiment and develop all kinds of policies. In order to achieve this, the policy space is divided into several (small) policies that can be combined together to offer the desired behavior. Each policy is responsible for a small and clear policy that handles one aspect of the desired policy. For instance, one of the proposed standard policies is a *string credentials* check [LPM]. It checks if a certain string, given by the user, is correct/known.

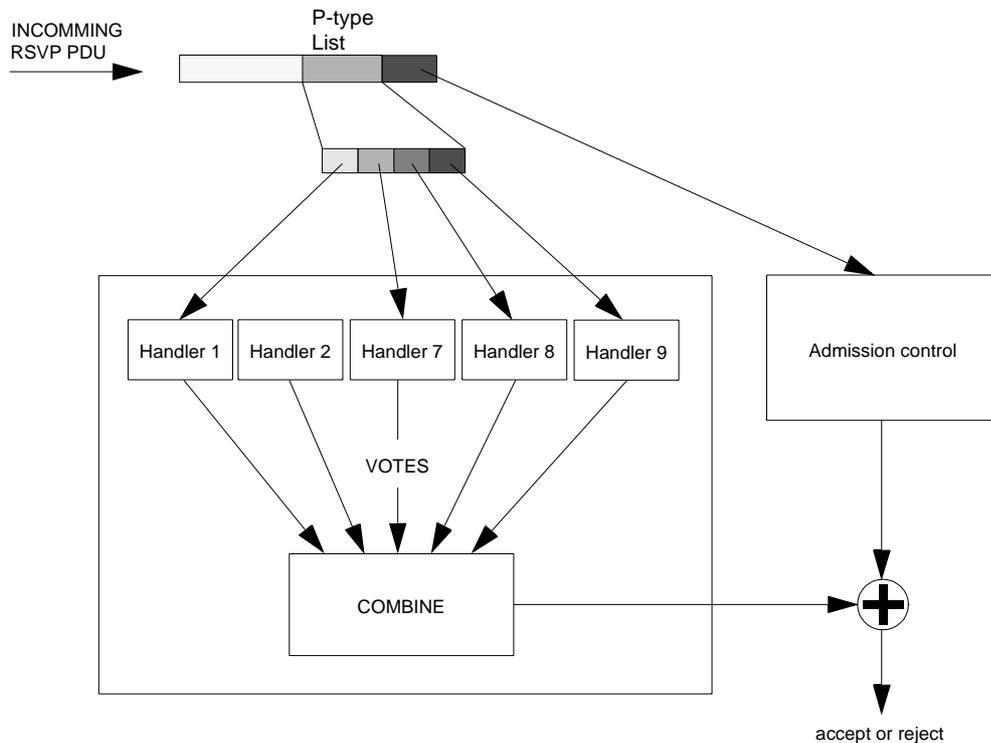
Each policy is identified by its description type, its P-type [LPM]. In the RSVP RESV PDU, that specifies the requested connection, a list of the desired P-types and their attributes is located. There are a number of standard defined policies, with defined P-types. Beside these standard policies it is possible for manufactures to define their own policies. The total policy space is 65535 policies wide.

A RSVP node will implement all, or a subset of all the available policies. Each implemented policy has its own handler. When a RSVP-PDU arrives, it will be given to the admission control module to check if it is technically possible to grant this connection request. If so, it will be given to the policy control module. In here the list of P-types is extracted from the PDU and each P-type is given to its corresponding handler. Each handler will examine the attributes belonging by this policy and will place its vote. The vote can be: accept (this policy is passed), snub (this policy is not passed, but someone else may accept) or veto (this policy is not passed and that is unacceptable). All these votes are combined into a single decision that indicates if this reser-

vation is granted or rejected [OOPS]. The rule used to combine the votes is: *A reservation is approved if there is at least one handler that accepts it, and no other that veto-ed it [LPM].*

If a node has no handler for a certain policy, the complete request will be forwarded to the next RSVP node on the path. This is no problem because RSVP is by design able to work with a mix of RSVP and non-RSVP nodes. Therefore if a node has a certain handler not implemented it looks like if that was a node with no RSVP support

Figure 3.8: RSVP Policy Control



### 3.2 ATM Accounting

A detailed overview of ATM accounting including background information and an overview of the current state of the art by A. van Dijk [VDIJK] can be found on the web.

Various solutions for ATM accounting are currently under development, by the IETF, the ATM Forum and some ATM Switch vendors.

The IETF [IETF] has two internet drafts available on accounting: "Accounting Information for ATM Networks, November 1996" and "Managed Objects for Controlling the Collection and Storage of Accounting Information for Connection-Oriented Networks, November 1996".

The ATM Forum [ATMF] includes in its B-ICI specification detailed information on the usage of the ATM network, that can be used for accounting purposes.

Some vendors of ATM equipment have accounting built into their ATM products. These vendors include Fore [FORE], Cisco [CISCO], General DataComm [GDC] and Newbridge [NBRIDGE].

---

## 4 References

- [ATMF] The ATM Forum, <http://www.atmforum.com>
- [CISCO] Cisco, <http://www.cisco.com>
- [FORE] Fore Systems, <http://www.fore.com>
- [GDC] General Data Comm, <http://www.gdc.com>
- [IETF] The Internet Engineering Task Force, <http://www.ietf.org>
- [LPM] Local Policy Modules (LPM): Policy enforcement for Resource Reservation Protocols. Internet-Draft, draft-ietf-rsvp-policy-lpm-01.[ps,txt], Nov. 1996.
- [NBRIDGE] Newbridge, <http://www.newbridge.com>
- [OOPS] Open Outsourcing Policy Service (OOPS) for RSVP. Internet-Draft, draft-ietf-rsvp-policy-oops-00.[ps,txt], Sept. 1997.
- [RAP] R. Yavatkar, D. Pendarakis et al., *A Framework for Policy-based Admission Control*, IETF Internet-Draft, draft-ietf-rap-framework-00.txt, November 20, 1997.
- [UNI31] ATM Forum, User Network Interface Specification version 3.1, july 1996.
- [VDIJK] ATM Accounting Management, Thesis of A. van Dijk, <http://www.snp.cs.utwente.nl/~nm/projects/ut-atm/accounting/>
- [Q2931] ITU-T Recommendation Q.2931, "B-ISDN DSS 2 UNI Layer 3 specification for basic call/connection control"
- [Q2941.1] Draft new ITU-T Recommendation Q.2941.1, "DSS2 Generic Identifier Transport"
- [Q2941.2] Draft new ITU-T Recommendation Q.2941.2, "DSS2 Generic Identifier Transport"

c

---

## 1 Scope of this section

ATM networks are designed to support Quality of Service (QoS). The agreed QoS contracts between service providers and their customers must be verified; QoS measurements are needed for that. Examples of situations where this is the case include the contract between PTT Telecom and SURFnet bv, and contracts between SURFnet bv and its connected institutions.

For such measurements to be useful it is required to have precise definitions of each QoS measure, and to have measurement methods to determine the values of these measures for a given connection. In this deliverable, we consider the definition of QoS measures and measurements methods to determine their values.

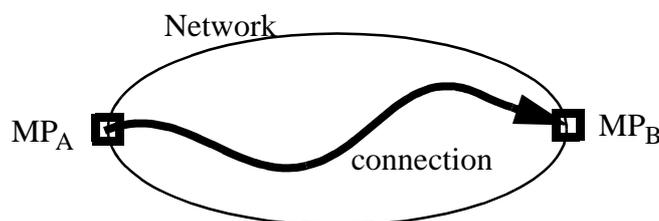
---

## 2 Introduction

By definition QoS measures pertain to a network connection. In this deliverable we consider point-to-point connections only. We can use a very simple, though general, model to clarify to what the measures pertain, this is illustrated in Figure 1.

The general setup for QoS measurements is given in the figure below

Figure 4.1: QoS Policy measures pertaining to a point-to-point connection.



The figure shows a connection through a network from Measurement Point A ( $MP_A$ ) to Measurement Point B ( $MP_B$ ). QoS measures will be determined for this single connection in a single direction. The network boundaries are not specified here and thus the precise location of the measurement point are left unspecified. These measurement points can either be some well chose Service Access Points (SAPs) at the ATM layer, or some well chosen physical interfaces.

This deliverable is organized as follows. In Section 3 a set of QoS measures is defined. This section is based on the definitions found in standards of the ATM Forum [AF-UNI3.1, AF-TMS4.0] and the ITU-T [I.356]. In Section 4 methods are examined and developed for QoS measurements needed to derive values of these measures for a given connection. Section 5 briefly discusses some bottom up approaches for QoS measurements. The conclusions of this work are presented in section 6. References to sources of information are listed in section 7.

---

## 3 Definition of ATM Layer QoS Measures

In this chapter we consider the QoS parameters as defined by ITU and ATM Forum. The basic material presented here originates from the following standards:

- ATM Forum User-Network Interface Specification 3.1
- ATM Forum Traffic Management Specification version 4.0
- ITU-T Recommendation I.356 B-ISDN ATM Layer Cell Transfer Performance.

The objective of this chapter is to give further clarification on the QoS Measures as defined by the standardization bodies. Such clarification is considered necessary, because the measures defined are rather complex and, moreover, they are mutually dependent in their definition.

The basic line of reasoning in the construction of the QoS measures is as follows:

- definition of “events” and “corresponding events”
- definition of “basic outcomes” (defined in terms of events and corresponding events)
- definition of derived outcomes (defined in terms of basic outcomes and a threshold)
- definition of “Severely Errored Cell Block Ratio”
- definition of “Cell Error Ratio”, “Cell Loss Ratio”, “Cell Misinsertion Rate”
- definition of speed related measures, like: Cell Transfer Delay (CTD), Mean CTD, Peak-to-peak CDV (ptpCDV), maximum CDV (maxCDV), Cell Delay Variation (CDV).

The three standards listed above, give definition of QoS measures that can in principle be determined (i.e. measured) in an out-of-service fashion. The standards also contain ‘in-service performance measurement methods’ to obtain estimations of the QoS measures. However, there can be large deviations between these estimated values and the outcomes that are obtained through out-of-service performance measurements. The approach we take deviates from the standards in that we also introduce definitions of QoS measure estimations. So, for a single measure we have two definitions: one is the true definition; the other is the estimated. These two definitions form the basis for an analysis of the deviation between in-service QoS measurements (the estimated case) and out-of-service QoS measurements (the ideal case). However, this analysis is beyond the scope of this document.

### 3.1 Definition of Events and Outcomes

#### 3.1.1 Events and Corresponding Events

There are two events that play a central role in the definition of outcomes: cell exit event and cell entry event. These events are defined as (we cite [AF-TMS4.0]):

*Definition of Cell Exit Event:* A “cell exit event” occurs when the first bit of an ATM cell has completed transmission out of an end-system to a Private ATM network element across the “Private UNI” Measurement Point, or out of a Private ATM network element to a Public ATM network element across the “Public UNI” Measurement Point, or out of an ATM end-system to a Public ATM network across the “Public UNI” measurement Point.

*Definition of Cell Entry Event:* A “cell entry event” occurs when the last bit of an ATM cell has completed transmission into an end-system from a Private ATM network element across the “Private UNI” Measurement Point, or into a Private ATM network element from the Public ATM network element across the “Public UNI” Measurement Point or

*into an ATM end-system from the Public ATM network element across the "Public UNI" Measurement Point.*

In the standards [AF-UNI3.1] exactly the same definitions of cell exit event and cell entry event are used. These ATM Forum definitions are based on, but not identical to, ITU definitions (see [I.356]).

When a cell exit event occurs at the transmitting end of an ATM connection we ideally expect that, eventually, a 'related' or 'corresponding' cell entry event occurs at the receiving end of the ATM connection. No formal definition of the notion 'corresponding' events is given in the standards. Hence, we have to use our intuitive understanding of this notion.

### **3.1.2 Basic Cell Transfer Outcomes**

In the three standards, four basic cell transfer outcomes are defined. These outcomes are called basic (in this report), because each outcome is based on the notion of corresponding events. There is also a type of outcome based on a consecutive sequence of basic cell transfer outcomes, this outcome is defined in the next section.

The four basic cell transfer outcomes are defined as follows (we adopt the definitions given in [AF-TMS4.0]):

*Definition of Successful Cell Transfer Outcome (SCTO): A successful cell transfer outcome occurs when a cell entry event corresponding to a cell exit event occurs within a specified time  $T_{\max}$  of the cell exit event, and: the binary content of the received cell information field conforms exactly with that of the corresponding transmitted cell; and the cell is received with a valid header field.*

*Definition of Error Cell Outcome (ECO): An errored cell outcome occurs when a cell entry event corresponding to a cell exit event occurs within a specified time  $T_{\max}$  of the cell exit event, and: the binary content of the received cell information field differs from that of the corresponding transmitted cell; or the cell is received with an invalid header field after header error control (HEC) procedures are completed.*

*Definition of Lost Cell Outcome (LCO): A lost cell outcome occurs when a cell entry event corresponding to a cell exit events fails to happen within the specified time  $T_{\max}$ .*

*Definition of Misinserted Cell Outcome (MCO): A misinserted cell outcome occurs when a cell entry event occurs without a corresponding cell exit event.*

The definitions presented in the next sections are based on these four basic outcomes and in particular on the number of outcome occurrences over specific sequences of cells. We denote the number of *SCTO*, *ECO*, *LCO* and *MCO* over a sequence as *nrSCTO*, *nrECO*, *nrLCO* and *nrMCO* respectively.

### **3.1.3 Cell Blocks and Cell Block Outcomes**

In [AF-TMS4.0] a fifth possible outcome, called Severely-Errored Cell Block Outcome (SECBO), is defined. It is a derived outcome, because it is based on the outcomes defined in the previous section. The definition can be constructed as follows:

*Definition of Transmitted Cell Block (TCB): A transmitted cell block is a sequence of  $Nt$  cells transmitted consecutively on a given connection.*

ITU-T Recommendation I.356 states that for practical measurements purposes, a cell block will normally correspond to the number of user information cells transmitted between successive OAM cells. In ITU-T Recommendation I.610 the following block sizes are specified:  $N_t=128, 256, 512$  or  $1024$ .

When transmitting a cell block at  $MP_A$ , we receive a cell block at  $MP_B$ . We define:.

*Definition of Received Cell Block (RCB):* A received cell block is a cell block of length  $N_r$  received at  $MP_r$  as a result of a transmitted cell block  $TCB$ .

Note that the length of a RCB, denoted  $N_r$ , may vary as a result of cell loss or misinsertions. Therefore  $N_r$  is not, unlike  $N_t$ , a constant.

For practical measurement purposes, transmitted cell blocks and received cell blocks can be identified using e.g. OAM cells as delineators. Assuming that all transmitted OAM cells have a corresponding received OAM cell, RCBs corresponding to TCBs can be identified.

*Definition of Severely Errored Cell Block Outcome (SECBO):* Given a transmitted cell block  $TCB$  and the resulting received cell block  $RCB$ , a severely errored cell block outcome occurs iff, relative to the corresponding cell blocks  $TCB$  and  $RCB$ :

$$nrECO + nrLCO + nrMCO \geq M, \text{ with } M \text{ some predefined constant.}$$

Practical values for the threshold value  $M$  is for further study.

For future use (in Section 3.3) we introduce Boolean  $ISSECBO_i$  relative to corresponding cell blocks  $TCB_i$  and  $RCB_i$  such that:

- $ISSECBO_i = true$  if  $TCB_i$  and  $RCB_i$  is a severely errored cell block outcome and
- $ISSECBO_i = false$  if  $TCB_i$  and  $RCB_i$  is not a severely errored cell block outcome.

Also, we will use simple arithmetic operations like summation, subtraction, multiplication and division to define the measures. There is one more construct that will be used to allow conditional computation: the IF construct. It is defined as follows:

$$IF(a, b, c) = \begin{cases} b & \text{if } a = true \\ c & \text{if } a = false \end{cases}$$

In fact this is a kind of 'it-then-else'-function because returns a value. This function will be used in combination with  $ISSECBO$ .

## 3.2 Identification of QoS Measures

Both standardization bodies, ITU and ATM Forum, have identified QoS measures. In Table 4.2 we have listed the QoS measures identified in [AF-UNI3.1] and [AF-TMS4.0] and grouped the measures according to the performance criteria 'Speed', 'Accuracy' and 'Dependability' (see also I.350)

Figure 4.2: Measures grouped by performance criterion

Criterion	ATM Forum (UNI 3.1)	ATM Forum TMS 4.0
Speed	Cell Transfer Delay (CTD) Mean Cell Transfer Delay (MCTD)  1-point Cell Delay Variation (1pCDV) 2-point Cell Delay Variation (2pCDV)	Cell Transfer Delay (CTD) Maximum Cell Transfer Delay (max-CTD) 1-point Cell Delay Variation (1pCDV) 2-point Cell Delay Variation (2pCDV) Peak-to-peak Cell Delay Variation (ptpCDV)
Accuracy	Cell Error Ratio (CER) Cell Misinsertion Rate (CMR) Severely Errored Cell Block Ratio (SECBR)	Cell Error Ratio (CER) Cell Misinsertion Rate (CMR) Severely Errored Cell Block Ratio (SECBR)
Dependability	Cell Loss Ratio (CLR)	Cell Loss Ratio (CLR)

### 3.3 Definition and Measurements of QoS Measures

The definitions of the QoS measures identified in Section 3.2 are based on the ‘outcomes’ defined in Section 3.1. In the next subsections we present these definitions, they are conform the definitions presented in the standards. However, there are two notes to be made:

**Note 1:** The QoS measures as defined in the standards are rather complicated, many QoS measures (for instance all accuracy and dependability related QoS measures) are based on severely errored cell clock outcomes and (i.e. derived outcome) and basic outcomes. As concrete example: Cell Loss Ratio is defined in terms of cell loss outcomes that occur in cell blocks that are *not* severely errored. Detailed explanation of the construction method used to define the measures is outside the scope of this document, but aspects about the rationale will be addressed briefly (Section 3.4)

**Note 2:** In the standards QoS measures are defined and also measurement methods for the QoS measures are defined. The in-service measurements methods defined, however, produce estimations of QoS measures. In fact, this implies that what is measured may seriously deviate from what we would actually like to measure. The estimations derived from the in-service measurements can be captured by formulas that are based on the same outcomes that have been defined previously. The approach taken in this document is that we present both the ‘formal’ definition of the QoS measures and the ‘estimated’ QoS measures. These two distinct definitions of the same QoS notions forms that basis for an investigation in which the two are compared, such investigation is beyond the scope of this document.

To assist in understanding how the QoS measures are defined, we have included an overview of the relationship between the QoS measures and outcome. This overview is given in Table 4.1.

Table 4.1: Relation between Qos measures and possible cell transfer outcomes

QoS Measure	SCTO	ECO	LCO	MCO	SECBO
CTD	X				
MCTD	X				X

QoS Measure	SCTO	ECO	LCO	MCO	SECBO
maxCTD					
1pCTD	X	X	X	X	
2pCTD	X				X
ptpCTD					
CER	X	X			X
CMR				X	X
SECBR		X	X	X	X
CLR			X		X

### 3.3.1 Severely Errored Cell Block Ratio (SECBR)

Given a sequence of transmitted cell blocks  $TCB_1, TCB_2, \dots, TCB_K$  and the sequence of corresponding received cell block  $RCB_1, RCB_2, \dots, RCB_K$ . The QoS measure “Severely Errored Cell Block Ratio” ( $SECBR_{def}$ ) is:

$$SECBR_{def} = \frac{\sum_{i \in [1, K]} IF(ISSECBO_i, 1, 0)}{K}$$

ATM Forum proposes a QoS measurement method to estimate the  $SECBR_{def}$ . We denote this estimation as  $SECBR_{est}$  and it is defined as follows:

$$SECBR_{est} = \frac{\sum_{i \in [1, K]} IF(ISSECBO_{est, i}, 1, 0)}{K}$$

with:

$$ISSECBO_{est, i} = \begin{cases} true & \text{if } Nt_i - Nr_i \geq M \\ false & \text{if } Nt_i - Nr_i < M \end{cases}$$

Note that:  $Nt_i = nrSTCO_i + nrECO_i + nrLCO_i$  and  $Nr_i = nrSTCO_i + nrECO_i + nrMCO_i$ .

### 3.3.2 Cell Error Ratio (CER)

Given a sequence of transmitted cell blocks  $TCB_1, TCB_2, \dots, TCB_K$  and the sequence of corresponding received cell block  $RCB_1, RCB_2, \dots, RCB_K$ . Let the number of errored cell outcomes in corresponding cell blocks  $TCB_i$  and  $RCB_i$  be  $nrECO_i$ . The Cell Error Ratio ( $CER_{def}$ ) is:

$$CER_{def} = \frac{\sum_{i \in [1, K]} IF(SECBO_i, 0, nrECO_i)}{\sum_{i \in [1, K]} IF(SECBO_i, 0, nrSTCO_i + nrECO_i)}$$

An estimation of the cell loss ratio,  $CER_{est}$ , based on an in-service QoS measurement method has not been defined in one of the ATM Forum standards nor in one of the ITU standards referred to in this document. However, they do suggest to define it on the basis of the BIP16 indicator. In-service estimation of CER is for further study.

### 3.3.3 Cell Loss Ratio (CLR)

The CLR is defined as:

$$CLR_{def} = \frac{\sum_{i \in [1, K]} IF(SECBO_i, 0, nrLCO_i)}{\sum_{i \in [1, K]} IF(SECBO_i, 0, N_{TCB})}$$

The in-service estimation of the Cell Loss Ratio is defined as:

$$CLR_{est} = \frac{\sum_{i \in [1, K]} IF(ISSECBO_{est, i}, 0, \max[0, Nt_i - Nr_i])}{\sum_{i \in [1, K]} IF(ISSECBO_{est, i}, 0, Nt_i)}$$

The definition of  $CLR_{est}$  deviates from  $CLR_{def}$  in basically two ways: the estimation of whether a cell block is severely errored is an under-estimation (if Errored Cell Outcomes and/or Misinserted Cell Outcomes occurs in a cell block it can occur that  $ISSECBO_{def} = true$  and  $ISSECBO_{est} = false$ ); also the estimated number of cell loss outcomes in a cell block is an under-estimation (if cell misinsertions occur in the cell block).

### 3.3.4 Cell Misinsertion Rate (CMR)

The QoS measure related to cell misinsertions is defined as a rate rather than a ratio. The rationale of this is that cell misinsertions are (assumed) independent of the cell transfer of the connection under consideration. The definition of the Cell Misinsertion Rate ( $CMR_{def}$ ) is:

$$CMR_{def} = \frac{\sum_{i \in [1, K]} IF(SECBO_i, 0, nrMCO_i)}{\sum_{i \in [1, K]} IF(SECBO_i, 0, T_i)}$$

With  $T_i$  the duration of the time interval in which  $TCB_i$  occurs.

In-service estimation of the Cell Misinsertion Rate,  $CMR_{est}$ , is based on the same measurement as to be used for  $CLR_{est}$ . It is defined as:

$$CMR_{est} = \frac{\sum_{i \in [1, K]} IF(SECBO_{est, i}, 0, \max[0, Nr_i - Nt_i])}{\sum_{i \in [1, K]} IF(SECBO_{est, i}, 0, T_i)}$$

The reasoning on the deviations between  $CMR_{def}$  and  $CMR_{est}$  is similar to the one given for  $CLR_{def}$  and  $CLR_{est}$  in the previous section.

### 3.3.5 Cell Transfer Delay (CTD)

All previously defined QoS measures are either accuracy or dependability related QoS measures and they are all based on counting specific cell transfer outcomes. All speed related QoS measures are based on time intervals between occurring cell events. Or, to formulate it more precisely, they are based on the elapsed time between a cell exit event (as defined in section 3.1.1) and the corresponding entry event (as defined in section 3.1.1).

To define the Cell Transfer Delay, we first have to consider, to what kind of outcome the measure relates.

In ITU-T Recommendation I.356 is defined for 'two corresponding successful cell transfer events'. Let the cell exit event  $e_{exit}$  and the cell entry event  $e_{entry}$  be corresponding successful cell transfer events that occur at time  $t_{exit}$  and  $t_{entry}$  respectively, the Cell Transfer Delay of this 'transfer event' is:  $CTD = t_{entry} - t_{exit}$ .

ATM Forum is less precise in its definition of the cell transfer delay. In [AF-UNI3.1] and [AF-TMS4.0] it is described as:

*The measured Cell Transfer Delay (CTD) is defined as the elapsed time between a cell exit event at measurement point A ... and the corresponding cell entry event at measurement point B ... for a particular connection.*

Hence, ATM Forum defines  $CTD$  for the following outcomes: successful cell transfer outcomes, errored cell transfer outcomes, "late cell transfer outcomes" (which are part of lost cell outcomes). The definition of the value of the  $CTD$  is the same for ATM Forum and ITU (hence, only the domain to which the definition applies differs).

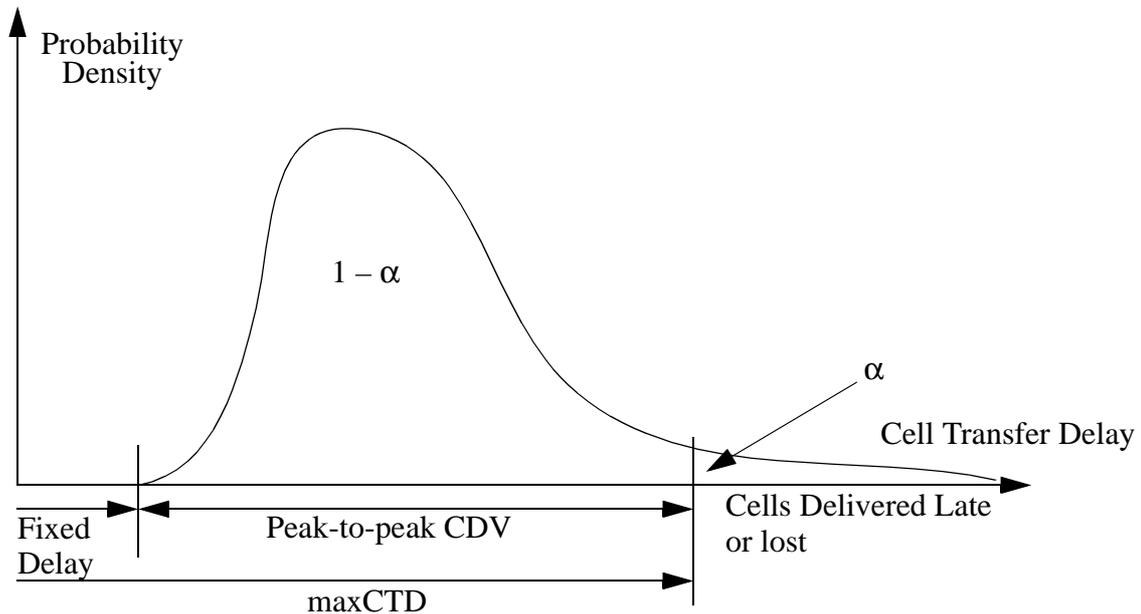
### 3.3.6 Probability Density Function of Cell Transfer Delay

In [AF-TSM4.0] an illustration of the probability density function of cell transfer delay, denoted  $pdf_{CTD}$  is given (see Figure 2). To begin with, it should be noted that this illustration is incorrect in the following way: For cells that are 'really' lost (e.g. due to buffer overflow) the cell transfer delay is not defined. Consequently, the region of the  $pdf_{CTD}$  associated with cells that are 'delivered late or lost' can, in reality, only relate to late cells.

From this observation we have to conclude also that the relationship between CLR and a (which has been left for further study in [AF-TMS4.0]) is probably even more complicated than initially anticipated.

Given a sample of Cell Transfer Delays we can construct a probability density function as illustrated in Figure 4.3. Furthermore, this figure shows the relationship between several metrics that are of interest.

Figure 4.3: Probability density function of the cell transfer delay



### 3.3.7 Mean Cell Transfer Delay (MCTD)

The Mean Cell Transfer Delay is defined as follows.

*Definition of Mean Cell Transfer Delay (MCTD):* Given a sample of Cell Transfer Delays,  $S_{CTD}$ , the Mean Cell Transfer Delay, denoted  $MCTD$ , over the sample is defined as:

$$MCTD = \frac{\sum_{d \in S_{CTD}} d}{|S_{CTD}|}$$

### 3.3.8 Maximum Cell Transfer Delay (maxCTD)

Considering the  $maxCTD$  as brought forward in [AF-TMS4.0] we can conclude that it can not be measured directly. There are two different ways to consider  $maxCTD$ :

- Either we need to consider  $maxCTD$  as some predefined constant (and consider it as the value to be taken for  $T_{max}$  as used in the definitions of ‘outcomes’ in Section 3.1.2, and given  $pdf_{CTD}$  we can determine  $\alpha$  (i.e. the portion of Late Cell Transfer Outcomes).
- Or, given  $pdf_{CTD}$  and some predefined value for alpha (i.e. the portion of Late Cell Transfer Outcomes), we can determine  $maxCTD$ .

In either case,  $maxCTD$  is a constant and its value and that of the Cell Loss Ratio are related.

### 3.3.9 1-Point Cell Delay Variation (1pCDV)

The 1-point Cell Delay Variation ( $1pCDV$ ) is a measure to characterize the variability in the pattern of cell events at some point of the connection (for instance the receiving end) with respect to the 'cell reference arrival time'. The  $1pCDV$  is defined as follows.

Let the actual arrival time of cell  $i$  at the measurement point be  $at_i$ .

Let the reference arrival time of cell  $i$ , denoted  $rt_i$  be defined as:

$$rt_0 = at_0 \text{ and } rt_{i+1} = \text{Max}[rt_i, at_i] + T$$

with  $T$  the reciprocal of the (negotiated) peak cell rate. The 1-point CDV of cell  $i$ , denoted  $1pCDV_i$  is defined as:

$$1pCDV_i = rt_i - at_i$$

In this definition we can have positive  $1pCDV$  values, corresponding to "early" cell arrivals and negative  $1pCDV$  values which corresponds to "late" cell arrivals. For further details on the  $1pCDV$  we refer to [AF-TMS4.0] and [ITU-I.356].

Given a stream of arriving cells at the measurement point, we can derive the  $1pCDV$  probability functions which gives information on the probabilistic nature of the cell delay variation. Observe that, because the  $1pCDV$  is only with reference to one end of the connection, the cell outcome is not taken into account. Hence, when measuring  $1pCDV$  we consider all cells arriving at the measurement point. In case of the receiving end of the connection this implies: successful cell transfer outcome, errored cell outcome, certain lost cell outcomes (namely the "late" ones) and misinserted cell outcome.

### 3.3.10 2-Point Cell Delay Variation (2pCDV)

The 2-point Cell Delay Variation, denoted  $2pCDV$ , describes the variability in the pattern of cell events observed at the receiving end of the connection with reference to the pattern of the cell arrivals at the transmitting end of the connection. The definition presented in [AF-TMS4.0] and [ITU-I.356] is complicated as it appears. However, in fact it is very simple. First of all, the outcomes to which the definition applies are: successful cell transfer outcome, errored cell outcome and "late cell outcome". The  $2pCDV$  is defined as follows.

Let a cell exit event  $i$  occur at  $t_{exit,i}$  and the corresponding cell entry event at  $t_{entry,i}$ , the  $2pCDV_i$  is define as:

$$2pCDV_i = CTD_i - CTD_0$$

Hence the 2-point CDV is similar to the CTD except for a shift by  $CTD_0$ , hence the CTD of cell 0 (this is proposed in [ITU-I.356]) or the 'Fixed Delay' which is the minimum cell transfer delay experienced by all cells (this is proposed in [AF-TMS4.0]). Also, the shapes of the probability density functions for CTD and  $2pCDV$  are identical.

Details on the relationship between  $1pCDV$  and  $2pCDV$  can be found in [AF-TMS4.0] and [ITU-I.356].

### 3.3.11 Peak-to-Peak Cell Delay Variation (ptpCDV)

The Peak-to-Peak Cell Delay Variation, denoted  $ptpCDV$ , is a constant which cannot be measured directly. Instead it is determined by  $pdfCDV$ , the fixed delay (let it be denoted  $fixedCTD$ ) and the percentage of cell outcomes that are considered as “late”. There is another simple relationship which relates  $ptpCDV$  to  $maxCTD$  and  $fixedCTD$ :

$$ptpCDV = maxCTD - fixedCTD$$

## 3.4 About the Rationale of the QoS Measures

The QoS measures defined by ATM Forum and ITU are cell block based: Cell Error Ratio, Cell Loss Ratio and Cell Misinsertion Rate are based on specific cell transfer outcomes that occur in cell blocks that are not severely errored. Also, speed related QoS measures are based on successful cell transfer delays that occur in cell blocks that are not severely errored.

As a result the definitions and estimations of the QoS measures become rather complex. The main questions that pops up is: What is the advantage of these definitions over simple straight forward definitions and estimations of QoS measures?

Simple, straightforward QoS measures are, for instance, averages over the connection duration. Such measures are inappropriate for real-time applications as voice and video in which correlation between outcomes play an important role in ‘user perceived quality of service’. Cell block based QoS measures do a better job here. The foundation of this statement can be found in the area of model-based performance studies, as illustrative example we refer to [NaKu92].

---

## 4 QoS Measurement Methods

In this section we discuss the basic QoS measurement methods that can be deployed to determine values for the QoS measures that have been identified in Section 3. We do not consider the instrumentation of the measurements, i.e. the monitoring equipment and the traffic generators. In general, this instrumentation may be dedicated hardware (e.g. ATM analyzer) or it may be specialized software. Among others, the instrumentation depends on the exact location of the measurement points (e.g. physical interfaces or ATM-SAPs).

The measurement methods can be organized according to the following ‘variables’: *connection usage* and *captured cells*.

By *connection usage* we indicate whether true user information is transferred along the connection, this is called *in-service*, or if the connection is entirely dedicated and used exclusively for QoS measurements, this is called *out-of-service*.

To determine values for the QoS measures, we need to gather a sample on which the required statistical inferences can be based. This sample (i.e. the *captured cells*) can be obtained in several distinct ways. The first way is to capture the entire sequence of cells that passes a measurement point over some period of time. On basis of these sequences, the true Basic Cell Transfer Outcomes can be determined, a detailed discussion of the analysis method is given in Section 4.1. The second way is based on capturing specific, identifiable cells that pass the measurement points, in particular OAM cells are of interest here. The analysis method based on OAM cells is discussed in detail in Section 4.2.

Given these options (for connection usage and captured cells), four measurement methods are identified as shown in Table 4.2. The method selected affects the functional requirements and capabilities needed to set-up the measurement configuration.

Table 4.2: QoS measurement methods

	In-service	Out-of-service
<b>sample of OAM Cells</b>	Interleave a dedicated OAM cell stream and the user data cell stream cells at the transmitting end, and capture the OAM cell stream at receiving end.	Inject a dedicated test stream (including OAM cells) at the transmitting end, and capture the OAM cells at the receiving end.
<b>sample of User Data Cells</b>	Capture the user data stream at the transmitting and receiving end of the connection of interest.	Capture the test stream at the transmitting end and receiving end of the connection of interest.

## 4.1 QoS Measurements based on User Data Cells

The four possible basic CTOs are (for their definition we refer to section 3.1.2): Successful Cell Transfer Outcome (SCTO); Errored Cell Outcome (ECO); Lost Cell Outcome (LCO); Misinserted Cell Outcome (MCO).

How these basic CTOs can be ‘detected’ is illustrated by the following example, it is based on a construction method. Assume we have set-up a connection and that we want to determine the QoS of the connection between  $MP_A$  and  $MP_B$  (see also Figure 1). Let the sequence of ATM cells sent during the lifetime of the connection at  $MP_A$  be as given in Table 3. And let the sequence of ATM cells received at  $MP_B$  be given by Table 4.3.

Table 4.3: Sequence of CTOs

$a_0$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$	$a_7$	$a_8$	$a_9$
-------	-------	-------	-------	-------	-------	-------	-------	-------	-------

$a_0$	$a_1$	$b_2$	$a_3$	$x$	$a_4$	$b_5$	$y$	$a_6$	$a_8$	$z$
-------	-------	-------	-------	-----	-------	-------	-----	-------	-------	-----

Each symbol in the above two tables denotes a time-stamped ATM cell, hence the entire cell-header and payload as well as the time instant at which it occurred at the specific measurement point.

We denote these sequences as A and B respectively, and the  $i^{\text{th}}$  element in the sequence as  $A_i$ . Note that the sequence length at  $MP_A$  and  $MP_B$  are 10 and ‘ respectively. In order to easily detect the CTOs that have taken place, we introduce a special cell; the ‘empty cell’ denoted  $\epsilon$ . The empty cell has no content, its length in terms of bits is 0, and the time-stamp of the cell event is not defined. The role of  $\epsilon$  can, to some extent be compared to the empty string as used in finite automata. We use  $\epsilon$  to extend both sequences as shown in Table 4.4.

Table 4.4: Extended sequence of CTOs

$a_0$	$a_1$	$a_2$	$a_3$	$\epsilon$	$a_4$	$a_5$	$\epsilon$	$a_6$	$a_7$	$a_8$	$a_9$	$\epsilon$
$a_0$	$a_1$	$b_2$	$a_3$	$x$	$a_4$	$b_5$	$y$	$a_6$	$\epsilon$	$a_8$	$\epsilon$	$z$

The first row is the extended version of A, denoted eA, the second row is the extended version of B, denoted eB.

It is the task of the row extension algorithm to pair up corresponding cell events, by comparing the payloads of the relevant cells. The algorithm has a notion of 'identical payloads' (straight-forward), 'nearly identical payloads' (e.g. 5 or less bit errors) and 'different payloads' (e.g. 6 or more bit errors). The algorithm determines when it is required to insert  $\varepsilon$  in either sequence eA or eB, and where. In the above table, we've used  $a_i$  and  $a_i$  to denote identical corresponding payloads,  $a_i$  and  $b_i$  to denote nearly identical corresponding payloads, x, y and z to denote payloads that differ a lot of payloads in their corresponding neighborhood in the source cell sequence, and  $\varepsilon$  for the empty cell event. A formal description of this algorithm is for further study.

On the basis of the extended sequences the CTO's are easily found:

- $(eA_i = eB_i) \Leftrightarrow SCTO$
- $(eA_i \neq eB_i) \wedge (eA_i \neq \varepsilon) \wedge (eB_i \neq \varepsilon) \Leftrightarrow ECO$
- $(eB_i = \varepsilon) \Leftrightarrow LCO$
- $(eA_i = \varepsilon) \Leftrightarrow MCO$

The basic CTO analysis measurement method yields the per block counts of the basic CTOs defined in section 3.1.2. The QoS measures given Section 3 are defined in terms of these exact counts. Hence, this measurement method results in values for the *defined* measures, not the *estimated* measures.

#### 4.1.1 Counting Errored Cell Transfer outcomes

In order to get an idea of the time interval required to observe 'enough' errored cells (so that valid statistical inferences can be made) we include a simple example of calculation. More details about values for measures can, for instance, be found in [ONV94,ONV95].

Assume that the connection in which we are interested consist of a certain number of Hops, say nrHops. Let the Bit Error Ratio of a single hop be BER<sub>one-hop</sub>. The Cell Error Rate of the connection is:

$$CER_{\text{connection}} = 1 - (1 - BER_{\text{one-hop}})^{\text{nrHops} \cdot 48 \cdot 8}$$

The measurement time interval, denoted  $T_m$ , needed to capture 30 cell errors (on basis of this number sound statistical inferences can be made) given a connection of  $S$  bps is:

$$T_m = \frac{53 \cdot 8 \cdot 30}{S \cdot CER_{\text{connection}}} \quad \text{sec.}$$

For a connection with  $nrHops = 10$ ,  $BER_{\text{one-hop}} = 10^{-10}$  and  $S = 10^6$  bps the cell error ratio of the connection and the measurement time interval are:

$$CER_{\text{end-to-end}} = 3.84 \cdot 10^{-7}$$

$$T_m = 33125 \quad \text{sec.}$$

The assumption made in these calculation is that bit errors occur randomly, which, in reality, is usually not the case.

### 4.1.2 Measuring Speed Related QoS Measures

To measure the speed related QoS measures given in section 3.2 using the Basic CTO Analysis Method it is necessary to capture the ATM cells passing at both measurement points ( $MP_A$  and  $MP_B$ ) and time-stamp each ATM cell (according to the 'rules' defined by the exit and entry event). The ATM cells (both header and payload) are used to determine the basic outcome and the time-stamps of the cells are needed in, for instance, the calculations of the cell transfer delays.

When considering time-stamps, we definitely need to consider clock issues. The IETF has developed a "Framework for IP Performance Metrics" [IETF-IPPM] in which they address clock characterization issues. Notions to characterize a single clock are: clock-offset (defined as the difference between the time reported by a clock and the true time, e.g. as defined by UTC); clock accuracy (defined as how close the absolute value of the offset is to zero); clock-skew (defined as the first derivative of the offset with respect to true time); clock drift (defined as the second derivative of the clock offset with respect to true time); clock resolution (defined as the smallest unit by which the clock's time is updated). These notions are important in ATM QoS measurements (e.g. for 1pCDV).

In QoS measurement it is especially interesting to express how two clocks compare. The terminology introduced in [IETF-IPPM], given two clocks C1 and C2, include: relative offset (defined as the instantaneous difference in time reported by C1 and C2); joint resolution (defined as sum of resolutions of C1 and C2, it indicates the accuracy of time intervals computed by subtracting time-stamps generated by the clocks). Two clocks are considered to be accurate with respect to one another if their relative offset is zero, such clocks are said to be synchronized. Not only in Internet measurements, but also in ATM measurements, time differences computed by comparing the output of two clocks are important (e.g. for CTD, 2pCDV).

Given the observation that CTD, 2pCDV and 1pCDV are in fact random variables, they each have a probability density function (as illustrated by Figure 2 for CTD). The distribution of CTD and 2pCDV have the same shape. In order to derive accurate values of the QoS measures the joint resolution of the clocks must be "small enough" (at least a few times) compared to the spread of the distributions (otherwise no accurate value for ptpCDV can be determined, nor the portion of late cell arrivals). In fact, the joint resolution introduces a quantization errors, if the resolution is "small enough" nice statistical properties for the quantization errors hold. The foundation of this theory is found in the area of digital signal processing, we refer to [JaNo84]. To determine the distribution of the 1pCDV a similar reasoning holds for the resolution of the clock related to the measurement point of interest.

## 4.2 QoS Measurements based on OAM Cells

The OAM cell based method to do QoS measurements is described in this section. We consider an ATM connection through an ATM network; the measurement points  $MP_A$  and  $MP_B$  are located on the ends of the connection at the ATM layer (see also Figure?). This method interleaves (at  $MP_A$ ) the cell stream to be measured with OAM cells that contain the following information:

- the number of transmitted user cells since last OAM cell (this is the block size);
- a time-stamp (determined the moment the OAM cell is interleaved into the cell stream);

- the sequence number of this OAM cell (to detect loss, errors or misinsertions of the measurement OAM cells themselves).
- A CRC over the information fields in the OAM cell (to detect transmission bit errors in the payload of the OAM cell)

At the receiving end of the connection (at  $MP_B$ ) the OAM cells are extracted from the cell stream, and the time-stamps of arriving OAM cells are taken. The sequence number is compared to the sequence number of the previous OAM cell to detect lost OAM cells, and the CRC is verified to detect OAM cell payload bit errors. Also the count is kept of user cells that arrived since the last OAM cell.

#### 4.2.1 Detecting Severely Errored Cell Blocks

The OAM cell method measures the *estimate* defined by the ATM Forum (see section 3.1.3) for a SECBO event. Upon reception of a measurement OAM cell, it first must be determined if any previous OAM cells were lost, by checking sequence numbers and OAM CRCs.

In the normal case, where all OAM cells are received without errors, the ATM Forum estimate for SECBO can be directly computed; the estimate basically compares the number of transmitted cells against the number of received cells.

#### 4.2.2 Counting Cell Errors per Cell Block

Only the HEC errors of cells arriving at the destination can be detected and counted using the OAM cell method; this is because the original payloads are not available for comparison against the actual received payloads. However, of cells that have a HEC error it can no longer be determined to which connection they belonged. There is a bit error possibly in the VPI or VCI field. For this reason the OAM method can not count errored cells *on a per connection basis*.

Also, in [I.356] it is noted that a BIP16 indicator could be used to estimate the cell error ratio over a cell block. Further study of this possibility is required.

#### 4.2.3 Counting Cell Losses per Cell Block

The OAM cell method for counting cell loss events per cell block actually counts the ATM Forum estimate for CLR (defined in section 3.3.3). This is done by comparing the actual number of received cells against the original transmitted number of cells. If less cells arrive than were transmitted, the difference is said to be the number of lost cells. This *undercounts* cell loss, because misinsertions again increase the number of arrived cells. What is actually returned as the LCO count per block is  $\max[0, Nt_i - Nr_i]$ , with the  $Nt_i$  and  $Nr_i$  the number of transmitted and received cells in this block of cells respectively (definitions given in section 3.3.1).

#### 4.2.4 Counting Cell Misinsertion per Cell Block

To get an estimate for the number of misinserted cells in a cell block also the number of received cells is compared to the number of transmitted cells, and the ATM Forum estimate for the number of misinsertions is computed (defined in section 3.3.4). The number of misinserted cells is estimated to be  $\max[0, Nr_i - Nt_i]$ .

### 4.2.5 Cell Transfer Delay

The CTD of an OAM cell is determined by subtracting the time of transmission of a OAM cell from its time of arrival. For an accurate calculation of the CTD a small joint resolution (compared to the CTD spread) is required (see also Section 4.1.2).

---

## 5 Examples of QoS Measurement Scenarios

In the previous chapters we have identified and defined QoS measures. Also, some elementary measurement methods have been considered. The theoretical approach taken is needed to get a clear view on what we would like to measure in order to obtain estimates for the QoS measures of interest. In this chapter we make a start with how these measurements can be realized. That is, we consider QoS measurement scenarios that enable the measurements to be performed.

In order to set-up these scenarios, there are a number of selections that have to be made. These are: i) identification of the location of measurement points (i.e. MPA and MPB as shown in Figure 1); ii) connection usage (in-service or out-of-service) and iii) sample (user data cells or OAM cells). In Section 5.1. we consider in more detail a scenario based on the following selection: MPA and MPB are the ATM-SAPs at user ends; the connection usage is out-of-service and the sample consists of captured user cells.

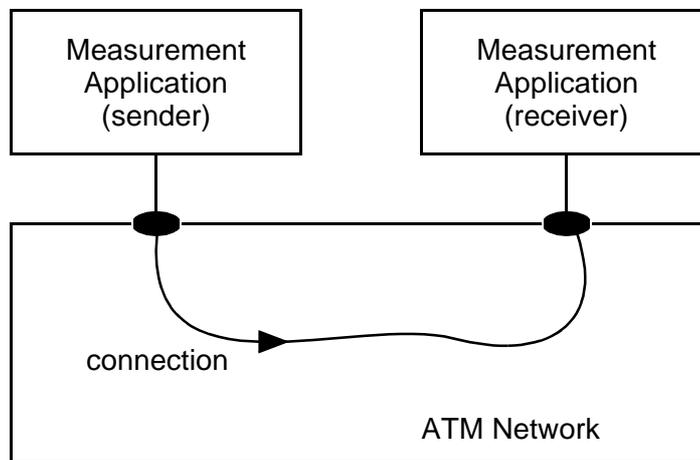
Another approach that can be followed is based on a line of reasoning in which the capabilities of physical network entities are considered. These capabilities can then be used to determine and implement QoS measurement in a bottom-up fashion. This approach is considered in Section 5.2.

This section follows bottom-up approach for QoS Measurements in the SURFnet ATM network. In separate sections it is examined what can be measured via out-of-service measurements at the application layer (section 5.1), what SNMP accessible information is available in the GDC switches used in SURFnet4 (section 5.2) and what support exists for measurement methods using OAM cells (section 5.3).

## 5.1 Application Based Out-of-service QoS Measurements

In Figure 4.4 below a measurement configuration is shown for out-of-service QoS measurements at the application layer. At both ends of a dedicated measurement connection measurement applications are shown.

Figure 4.4: Measurement configuration



The sender application generates a (known) cell stream. This cell stream traverses the network, reaches the receiver application which captures the stream. The original cell stream and the received cell stream can then be analyzed as described in section 4.1.

To determine values for the speed related QoS measures, the cells at the transmitting and receiving end must be captured with a time stamp. In order to obtain valid results the clocks at the sender and receiver must have a known relative offset and the joint resolution of the two clocks must be 'small enough' (see also Section 4.1.2). This induces constraints on the platforms (i.e. terminals) on which the applications run.

Provided that  $\max\text{CTD} (=T_{\max})$  can be chosen freely or is large enough, we can determine the dependability and accuracy related QoS measures without time stamping capabilities.

## 5.2 Bottom-up Approach for QoS Measurements

### 5.2.1 SNMP And MIBs

In the context of this document the GDC switches have not been examined, but it is known that the GDC incorporates an extensive proprietary Management Information Base (MIB). Information can be retrieved from this MIB using the SNMP management protocol. It is to be expected that the GDC switch will at least have some MIB counter objects relevant for QoS measurements.

Subject of further investigations should be if the GDC has these features in its MIB:

- per ATM interface counters for HEC errors, received cells, transmitted cells;
- per ATM connection counters for received cells, transmitted cells, cells dropped due to insufficient buffer space, cells dropped due to traffic contract violations.

## 5.3 OAM Cells

It is currently unknown (to us) if the GDC switch supports in-service QoS measurements using OAM Cells. The use of OAM cells for in-service loss measurements is described in annex B section 1.2 of [AF-TMS4.0] and in section 4.2 of this document. To clarify this issue more technical and implementation details must be submitted and studied (to us).

---

## 6 Conclusions

- ATM Forum and ITU definitions of QoS measures are not straight-forward to understand; in particular the notion of a Severely Errored Cell Block and the dependency of the other QoS measures on this notion is a complicating factor, both in understanding these measures and in measuring them.
- The Basic CTO analysis method results in values for the *exact definitions* of the QoS Measures given in section 3, whereas the OAM Cell based method results in values for the *estimates* for those measures.
- To obtain values for the speed related QoS measures it is required to capture and time stamp cells at both measurement points. With respect to the clocks used for time stamping, the relative offset must be close to zero or be known. Also, the joint resolution of the clocks must be small enough in comparison with the probability density functions that are under consideration. In some cases (e.g. the 2pCDV for a CBR cell stream, or the 1pCDV) time stamps at the receiving end are required only.
- The notion of transmitted user data cells that are non-conforming to the connection traffic contract is not taken into account in this report. This is of importance in cases where non-conforming cells are actually accepted by the ATM network. These non conforming cells are expected to experience a worse QoS than conforming cells. In particular, we may expect that the probability that non-conforming cells are lost is higher than the probability that conforming cells are lost. The impact of non-conforming cells on QoS measurement methods and QoS measures is for further study.
- In this deliverable we have investigated the definitions of QoS measures and possible ways to measure them. Also, we have addressed the possibility of using a 'management' approach for these measurements. However, there is a knowledge gap in the capabilities of the switches that are of interest here (the GDC Switches) which needs to be filled in order to come to a conclusive result.
- In this document, we have considered QoS measures for a connection of interest. That is we have assumed a transmitting and receiving end of a connection. However, we have NOT identified the location of these connection ends in the real network. In principle the choice of connection end-points is free, e.g. we can select physical interfaces or ATM-SAPs to be these end-points. The selection made, however, highly influences the instrumentation of the QoS measurement. If physical interfaces are selected as connection end-points, monitoring equipment (i.e. ATM analyzers) is needed to capture and time-stamp the ATM-cells. If ATM-SAPs are selected as connection end-points, specialized software can be used for the monitoring and time-stamping of ATM-cells.

---

## 7 References

- [AF-UNI3.1] ATM-Forum, *ATM User-Network Interface Specification, version 3.1*, September 1994.
- [AF-TMS4.0] ATM-Forum, *Traffic Management Specification, version 4.0*, April 1996.
- [I.350] ITU-T Recommendation I.350, *General Aspects of Quality of Service and Network Performance in Digital Networks, including ISDNs*, March 1993.
- [I.356] ITU-T Recommendation I.356, *B-ISDN ATM Layer Cell Transfer Performance*, November 1993.
- [I.610] ITU-T Recommendation I.610, *B-ISDN Operation and Maintenance Principles and Functions*, November 1995.
- [IETF-IPPM] *Framework for IP Performance Metrics*, IETF Draft, WG IPPM, <draft-ietf-ippm-framework-02.txt>.
- [JaNo84] N.S. Jayant, P. Noll, *Digital Coding of Waveforms: Principles and Applications to Speech and Video*, Prentice-Hall, 1984.
- [NaKu92] R. Nagarajan, J.F. Kurose, *On Defining, Computing and Quaranteeing Quality-of-Service in High-Speed Networks*, INFOCOM 1992, pp. 2016-2025.
- [ONV94] R.O. Onvural, *Asynchronous Transfer Mode Networks: Performance Issues*, Artech House, 1994.
- [ONV95] R.O. Onvural, *Asynchronous Transfer Mode Networks: Performance Issues*, Second Edition, Artech House, 1995.

---

### 1 Introduction

This section describes the functions of Operation And Maintenance (OAM) cells in an ATM network. The two types of OAM flows (The F4 and F5 flows) are examined, as are the functions that can be performed using those OAM flows. Furthermore, current support for OAM in standards and in available ATM equipment is examined.

---

### 2 F4 and F5 OAM Flows

ATM connections can carry user cells and non-user cells. OAM cells are non-user cells. ITU-T Recommendation I.610 [I.610] defines 5 OAM hierarchical levels. Levels F1 through F3 are in the physical layer, and are outside the scope of this document. Levels F4 and F5 are at the ATM Virtual Path level and the ATM Virtual Channel level respectively. These flows are the scope of this document.

The F4 and F5 flow identify an OAM cell in a different way. For the F4 flow (which is defined for the Virtual Path ATM level) special VCI values within the Virtual Path are reserved for this purpose. For the F5 flow (defined for the Virtual Channel level) the VCI value can not be used. In this case the Payload Type field in the cell header is used to identify the cell as an OAM cell.

Table 5.1: OAM cell identification for F4 and F5 flows

F4 flow		F5 flow	
VCI	Interpretation	PTI code	Interpretation
3	Segment OAM F4 flow cell	100	Segment OAM F5 flow cell
4	End-to-end OAM F4 flow cell	101	End-to-end OAM F5 flow cell
6	Resource Management cell	110	Resource Management cell

---

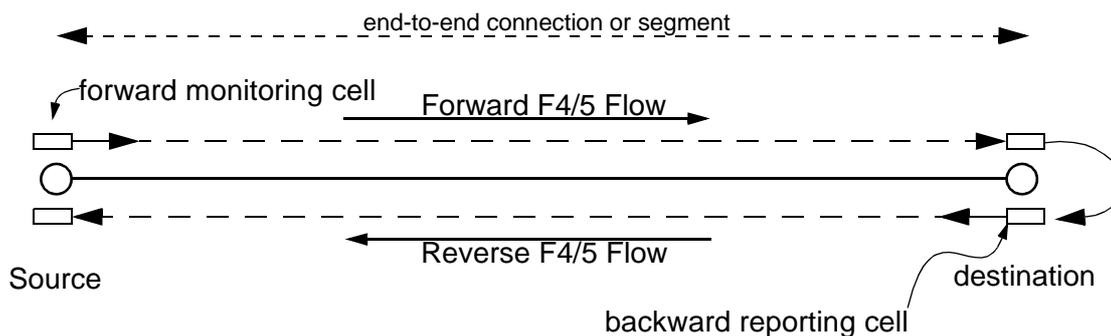
### 3 OAM functions

This section discusses the available OAM functions.

#### 3.1 Performance management

OAM cells are defined that allow performance management of segments or end-to-end connections. The procedure for performance management is outlined in Figure 5.1 below.

Figure 5.1: OAM Cell Flows



At the source of an end-to-end connection or segment forward monitoring cells are inserted. Blocks of user cells are delimited by such forward monitoring cells. These cells contain counter values for the number of user cells in the last block.

At the destination a backward reporting cell is generated for each encountered forward monitoring cell. This is referred to as “paired OAM cells” in section 7.3 of [I.610]. Also at the destination the actual number of user cells in each block is counted. These counter values are inserted into the backward reporting cells also. When the backward reporting cell arrives back at the source entity, the number of transmitted cells is compared against the number of received cells.

From this comparison estimates can be made for:

- errored blocks
- loss/mis-insertions of cells within a monitored block of cells;

Other functions, such as estimates for cell transfer delay, are considered to be for further study by [I.610].

## 3.2 Activation/Deactivation and System management

### *Activation and Deactivation*

The performance monitoring function and the continuity check function of equipment can be activated and deactivated through special OAM cells for this purpose. An OAM cell is sent to the system that has to activate or deactivate a particular function. All the required information for this is encoded in the OAM cell. The destination system then either confirms or denies the request by sending a response OAM cell back. This results in a reliable way to activate or deactivate these functions.

### *System management*

For the F4 and F5 flows system management OAM cells are defined. These cells can be used to control and maintain various functions at the VC or VP layer. They are processed only in end-user equipment. [I.610] explicitly states that “*end-user equipment implementation of the system management OAM cell is optional and is **not recommended** except for specific applications which are not satisfied by other mechanisms*”.

---

## 4 Support for OAM functions in standards and ATM equipment

This section briefly examines support for OAM functions in standardization bodies and in current ATM equipment.

### 4.1 Standardization efforts

The IETF has drafted a document that defines OAM based test procedures for ATM networks. The current status of this work can be found in [draft]. This document represents work in progress, and has no formal status of any kind.

The document defines MIB objects that enable the following functions:

- Assign ID-s to ATM systems for OAM loopback tests;
- Start different types of OAM loopback tests;
- Retrieve the result of those OAM loopback tests.

### 4.2 Support in ATM equipment

#### 4.2.1 Cisco LS1010 ATM switch

The Cisco LS1010 switch has support for two of the OAM functions identified in this section:

The LS1010 supports OAM based loopback tests. These test can be defined and initiated from the switch console (via e.g. a telnet session to the switch), but also via the switch SNMP agent. The loopback test functionality is comparable to the loopback functions defined in the IETF work. Loopback tests can be defined, started, and the results can be retrieved from the agent. The agent can send a SNMP trap to notify a manager upon completion of a loopback test. The MIB objects that are defined to support these functions are proprietary to Cisco, and the term *"oamLoopbackPing"* is used for naming them.

The OAM loopback tests of the LS1010 also gather some performance information: the minimum, average and maximum round trip time for a set of loopback OAM cells is computed. These results can be retrieved via the SNMP agent of the switch.

#### 4.2.2 GDC apex ATM switch

The GDC apex ATM switch MIB definition [GDC-MIB] has some objects for OAM loopback tests. Loopback tests can be defined, started, and the results can be queried. No performance information (like e.g. measured round trip times for a loopback test) is represented in the MIB.

---

## 5 References

- [GDC-MIB] GDC DV2 MIB, version 4.00, September 1995
- [I.610] ITU-T Recommendation I.610, *"B-ISDN Operation And Maintenance Principles And Functions"*, November 1995.