Characterization of SNMP MIB Modules

Jürgen Schönwälder

j.schoenwaelder@iu-bremen.de

International University Bremen

Campus Ring 1

28725 Bremen, Germany

Motivation

- Understanding the contents of MIB modules
- Which SMIv2 features are being used heavily?
- How many MIB modules are produced over time?
- How are MIB modules revised?
- What is a typical size of a MIB module?
- What is the typical size of encoded values?
- . . .

MIB Module Sets

MIB Module Set	Modules	Types	Tables	Columns	Scalars	Notifications
IETF	174	377	875	7479	785	195
ATM Forum	11	63	79	777	39	5
Cisco Systems	482	936	1966	16952	3719	611
Enterasys	58	76	128	825	364	28
Juniper Networks	99	170	434	3606	1051	87
All Modules	824	1622	3482	29639	5958	926

- Quality of MIB module sets made available varies:
  - Lack of separation of vendor specific modules from standard modules
  - Usage of pre-standard modules which differ from the standard modules
  - SMIv2 problems still exist in some vendor's modules

Terms and Metrics

- A *type definition* is either an ASN.1 type definition or an invocation of the SMIv2 `TEXTUALCONVENTION` macro.
- A *variable definition* is the invocation of the SMIv2 `OBJECT-TYPE` macro introducing a scalar object or a columnar object.
- A *notification definition* is the invocation of the SMIv2 `NOTIFICATION-TYPE` macro.
- The MIB *module size* is defined as the number of type, variable and notification definitions contained in a MIB module.
- The *index length* of a conceptual table is given by the number of variables appearing in the SMIv2 `INDEX` clause.

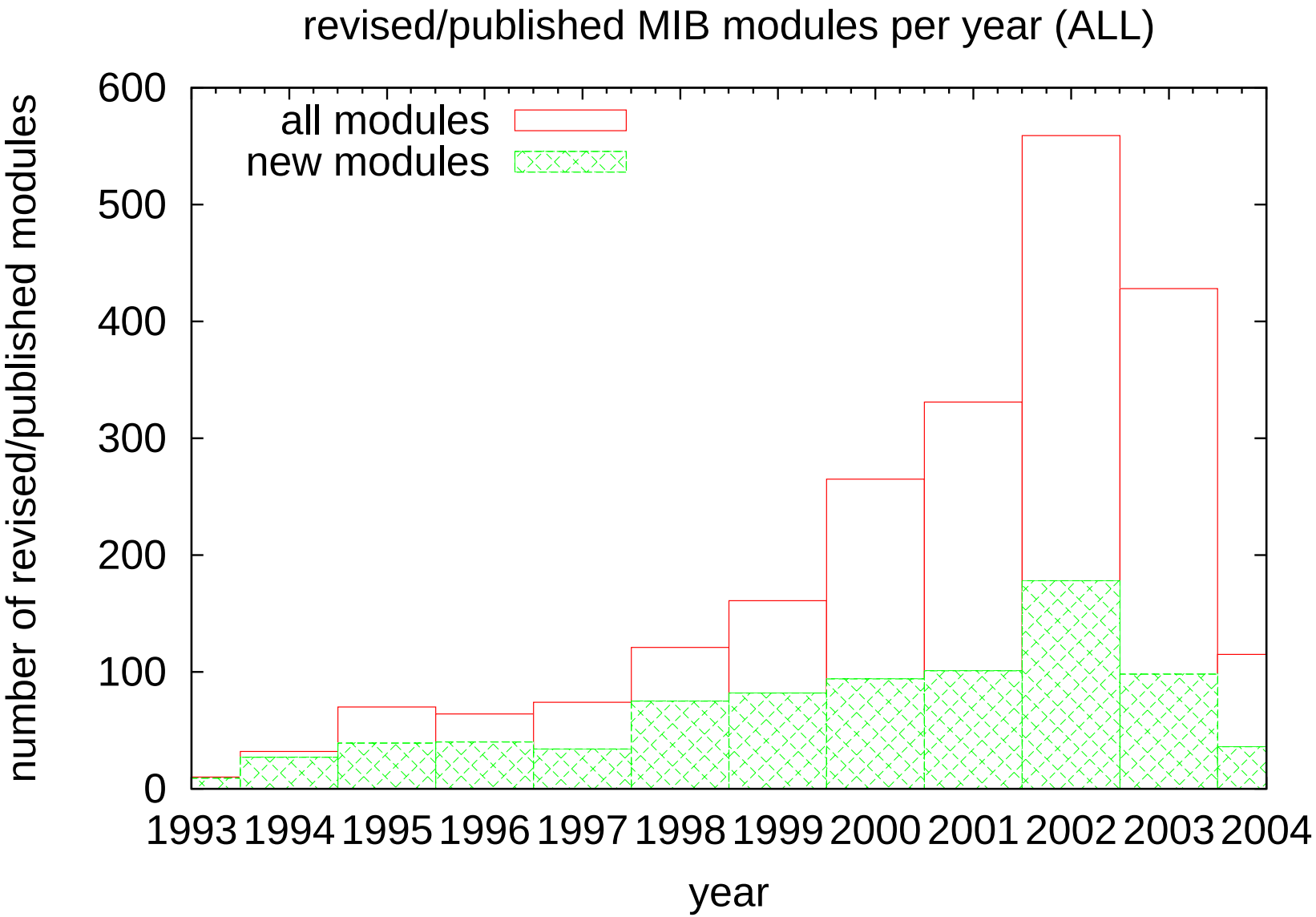
Terms and Metrics (cont.)

- The *row encoding size* of a conceptual row is defined as the number of bytes needed for the BER encoding of a PDU containing the columnar objects of that row (excluding index columns).
- The *notification encoding size* of a given notification is defined as the number of bytes needed for the BER encoding of a notification PDU which includes the mandatory objects of that notification.

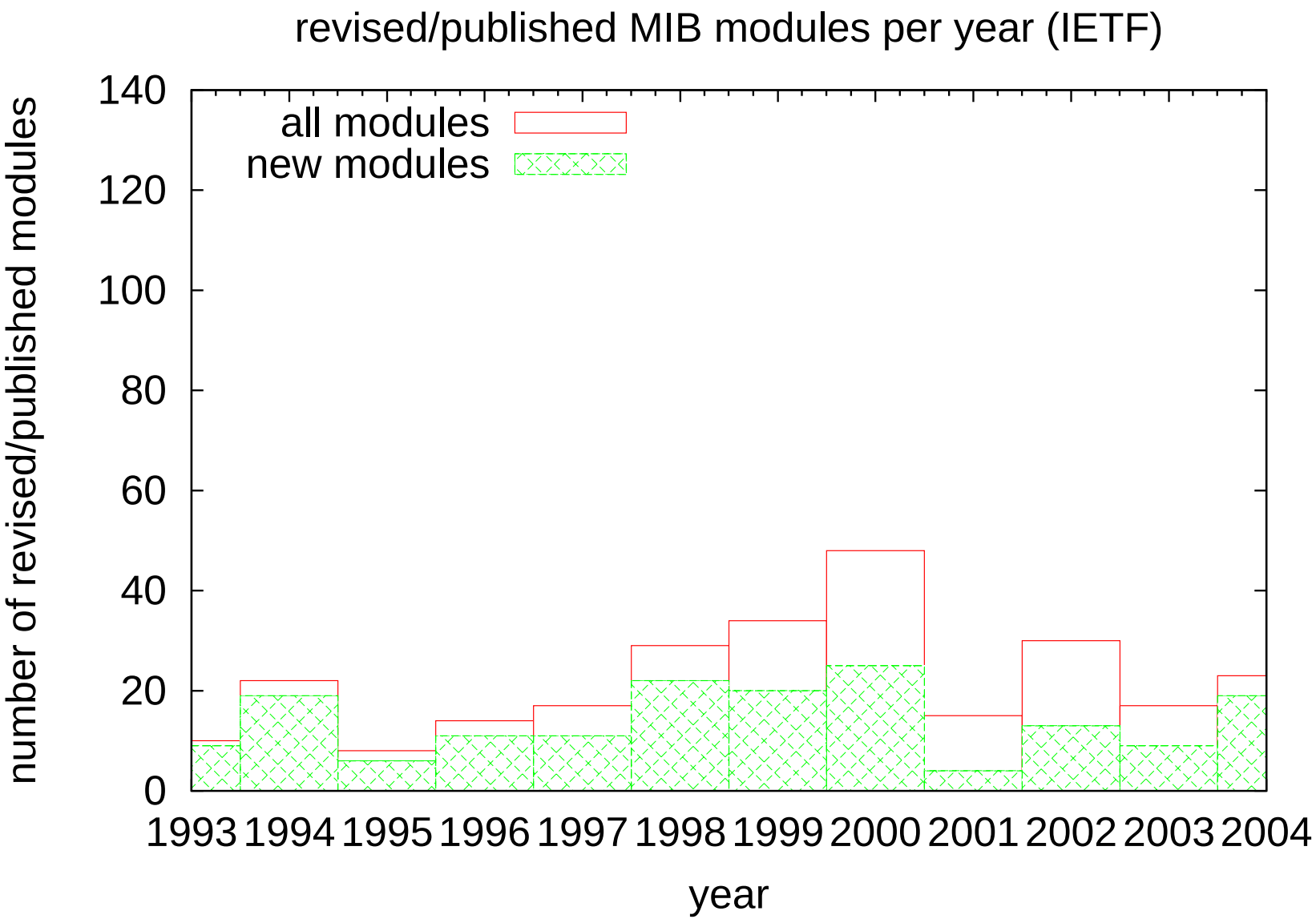
$\Rightarrow$ The encoding sizes are computed by picking a value in the middle of the value space of the underlying type.

$\Rightarrow$ For some well-known types, the length of typical values is assumed (`INET-ADDRESS` typically contains a 4-byte IPv4 address).

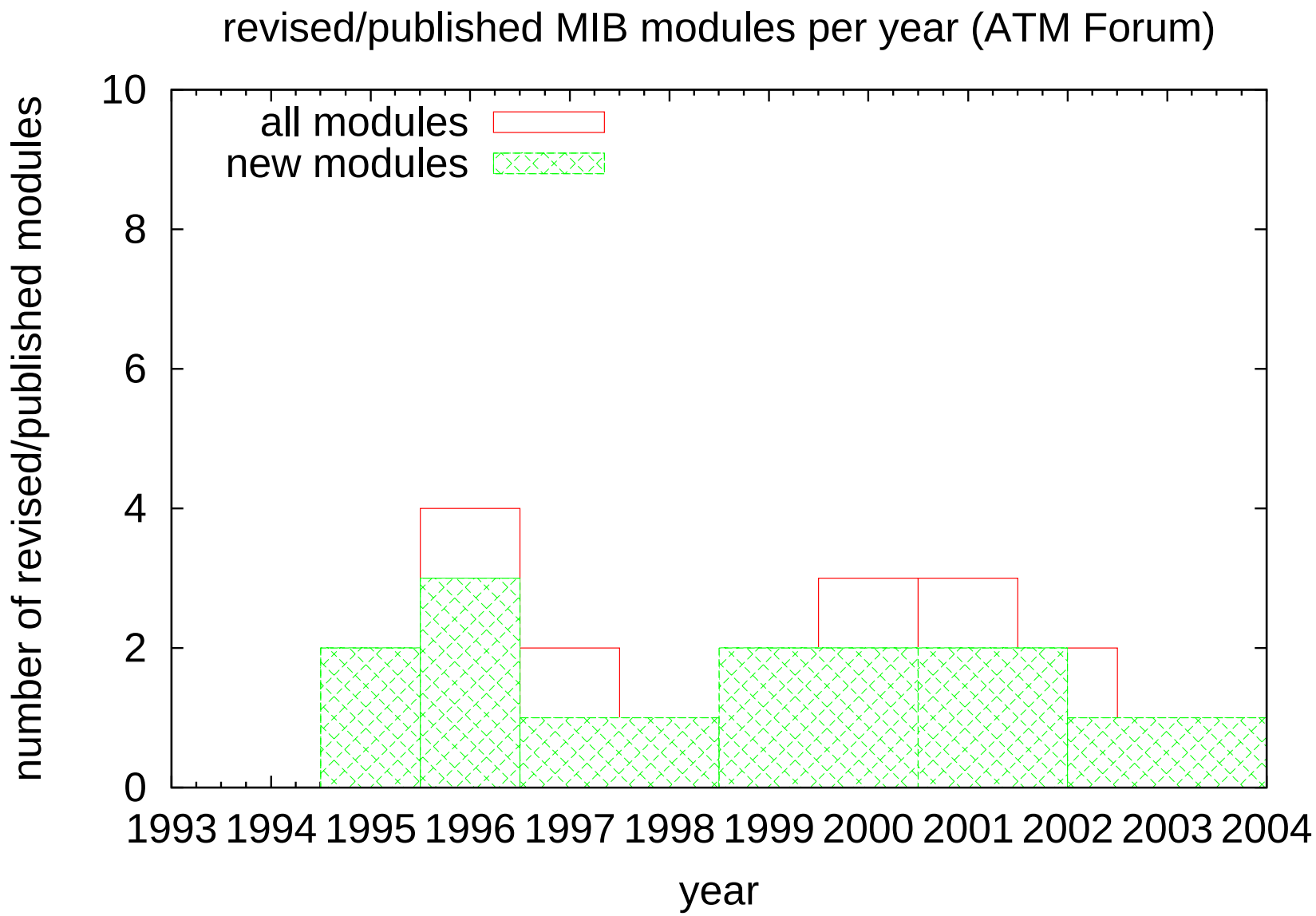
MIB Module Productivity



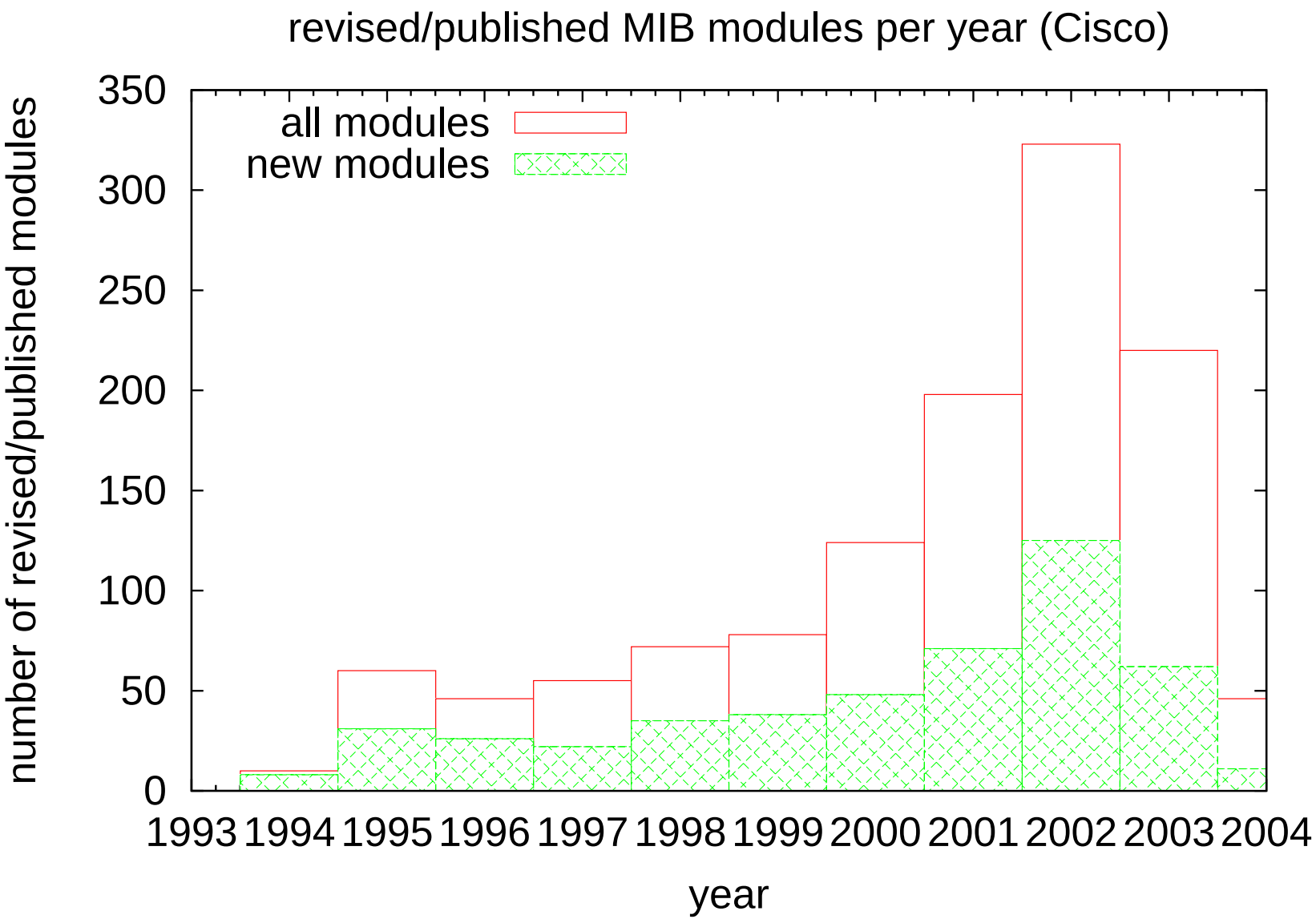
IETF MIB Module Productivity



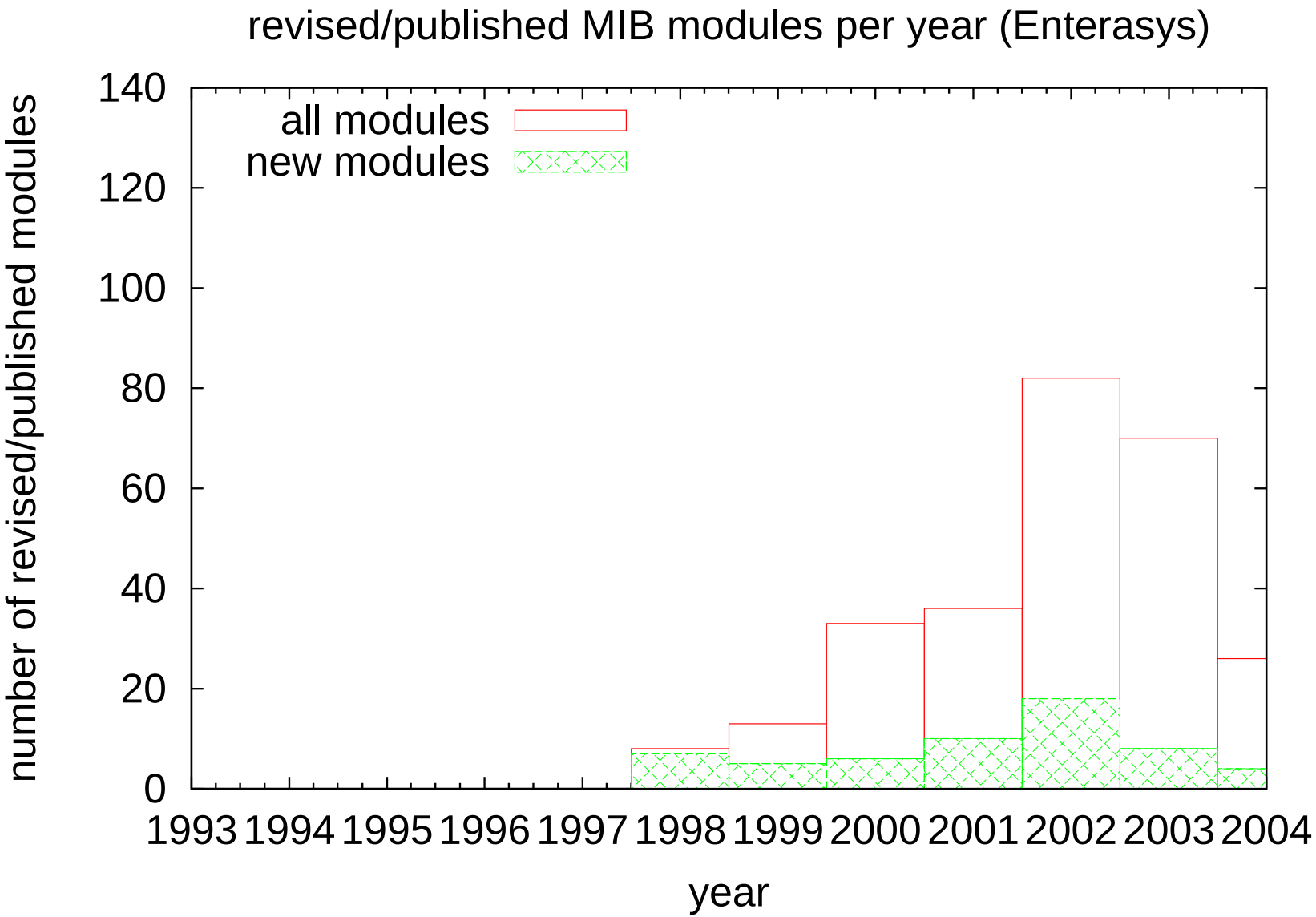
ATMF MIB Module Productivity



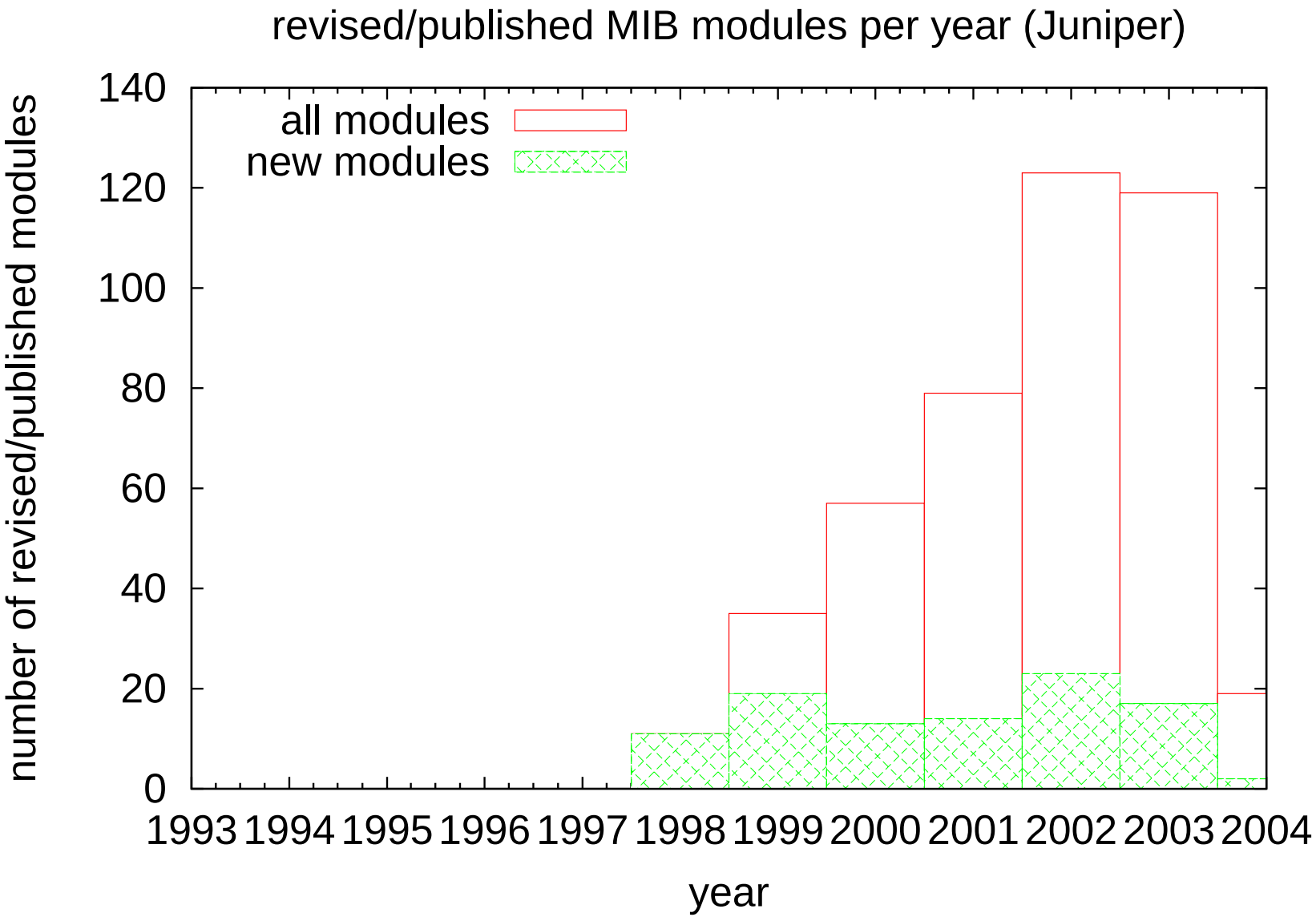
Cisco MIB Module Productivity



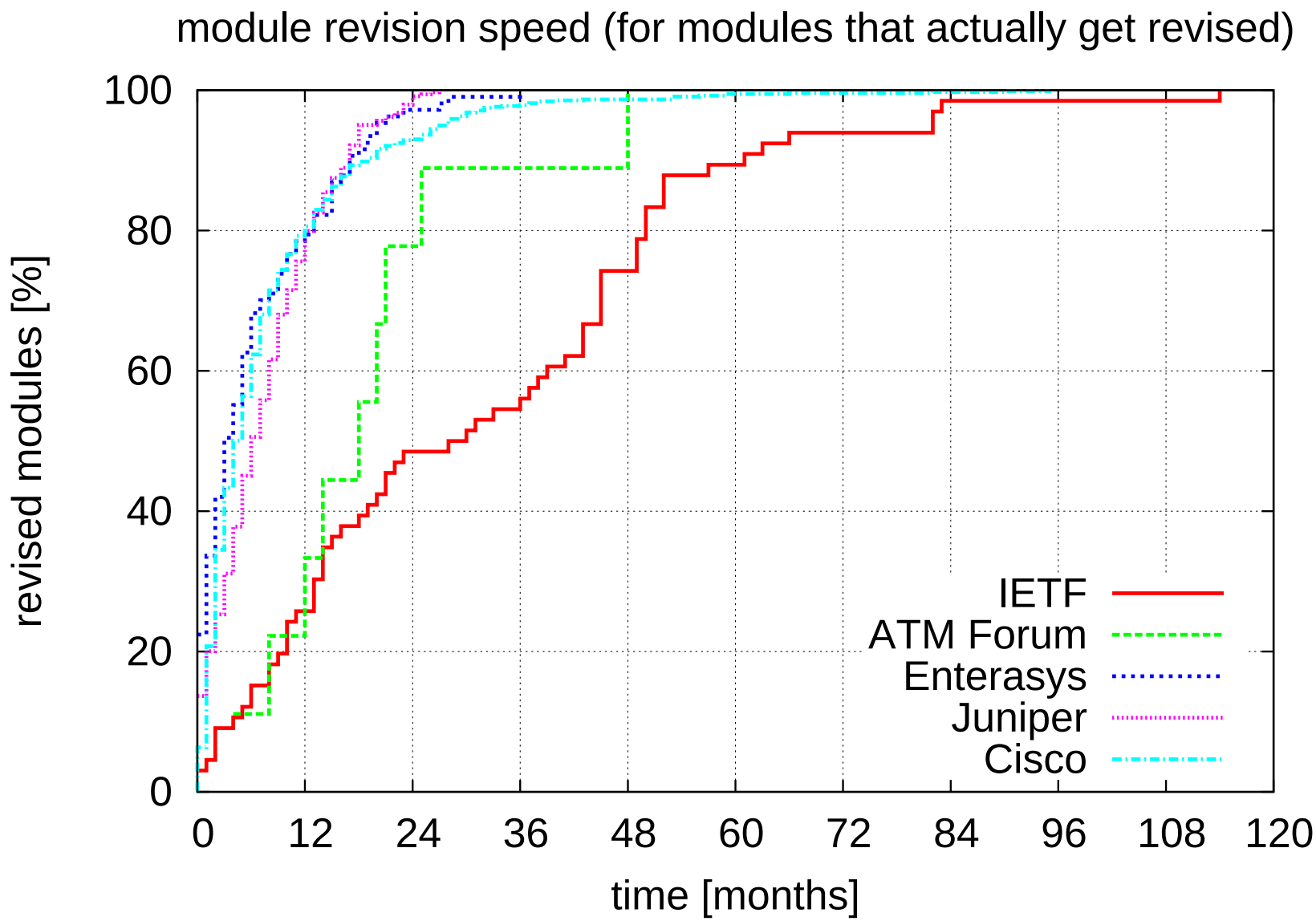
Enterasys MIB Module Productivity



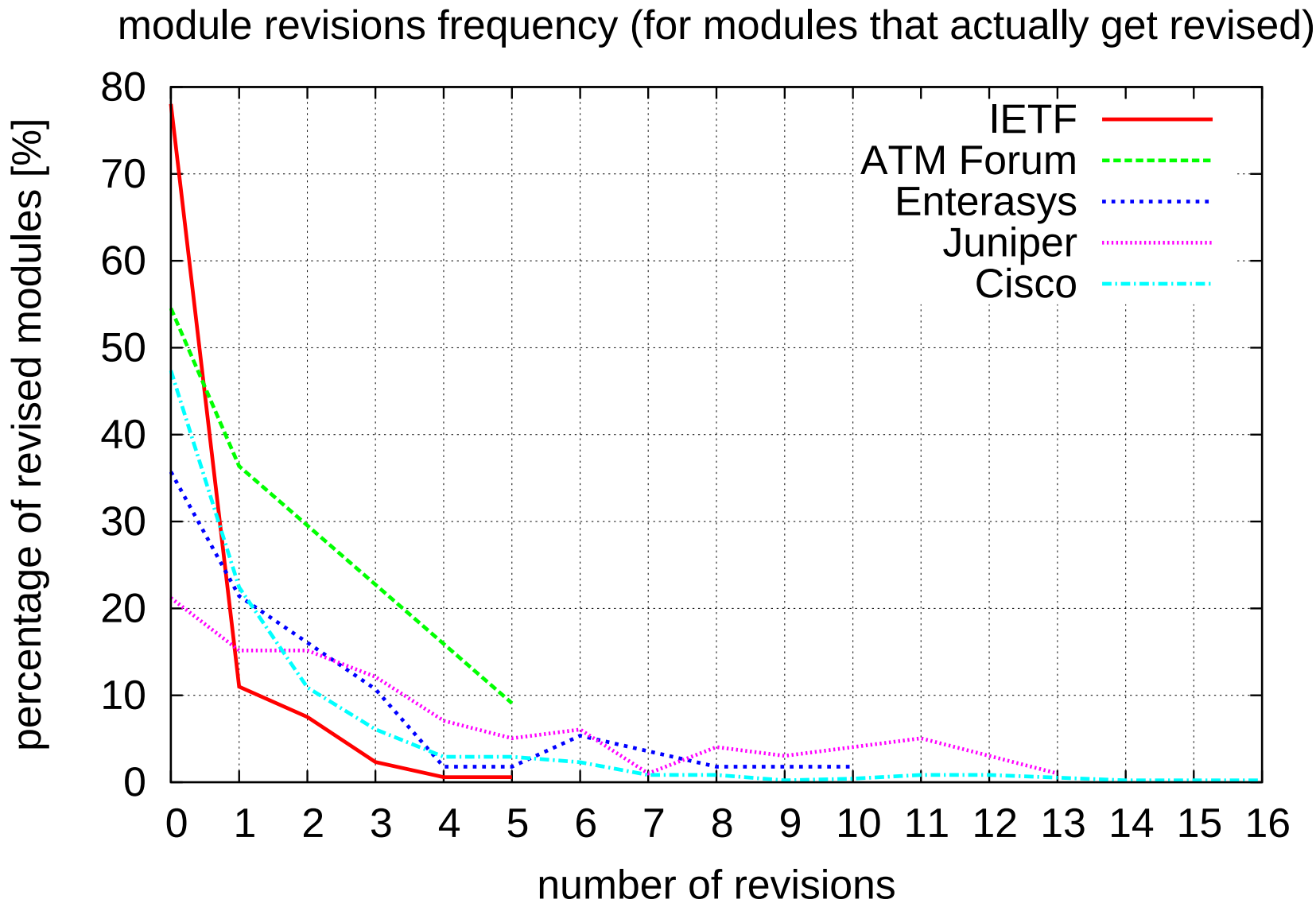
Juniper MIB Module Productivity



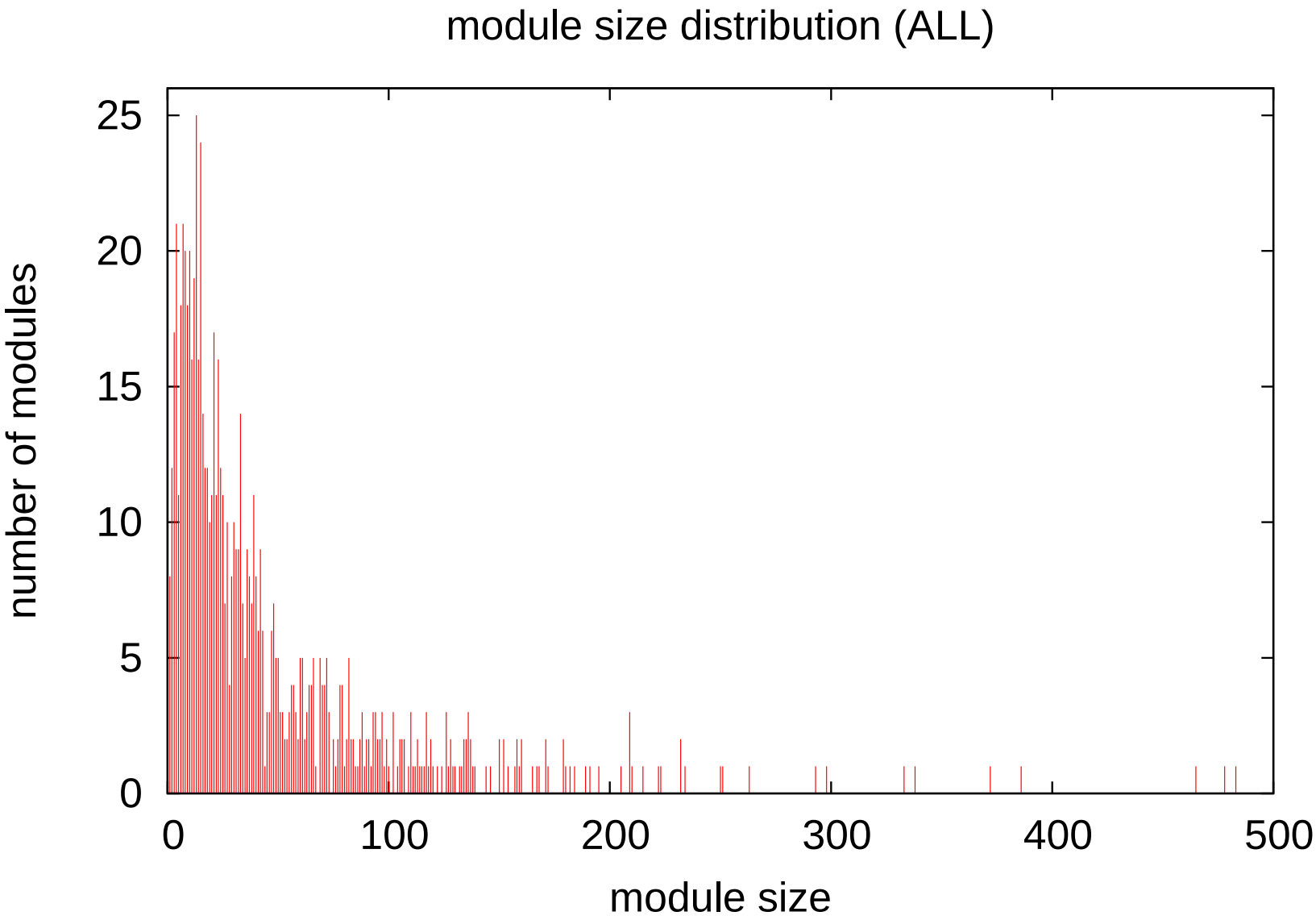
MIB Module Revision Speed



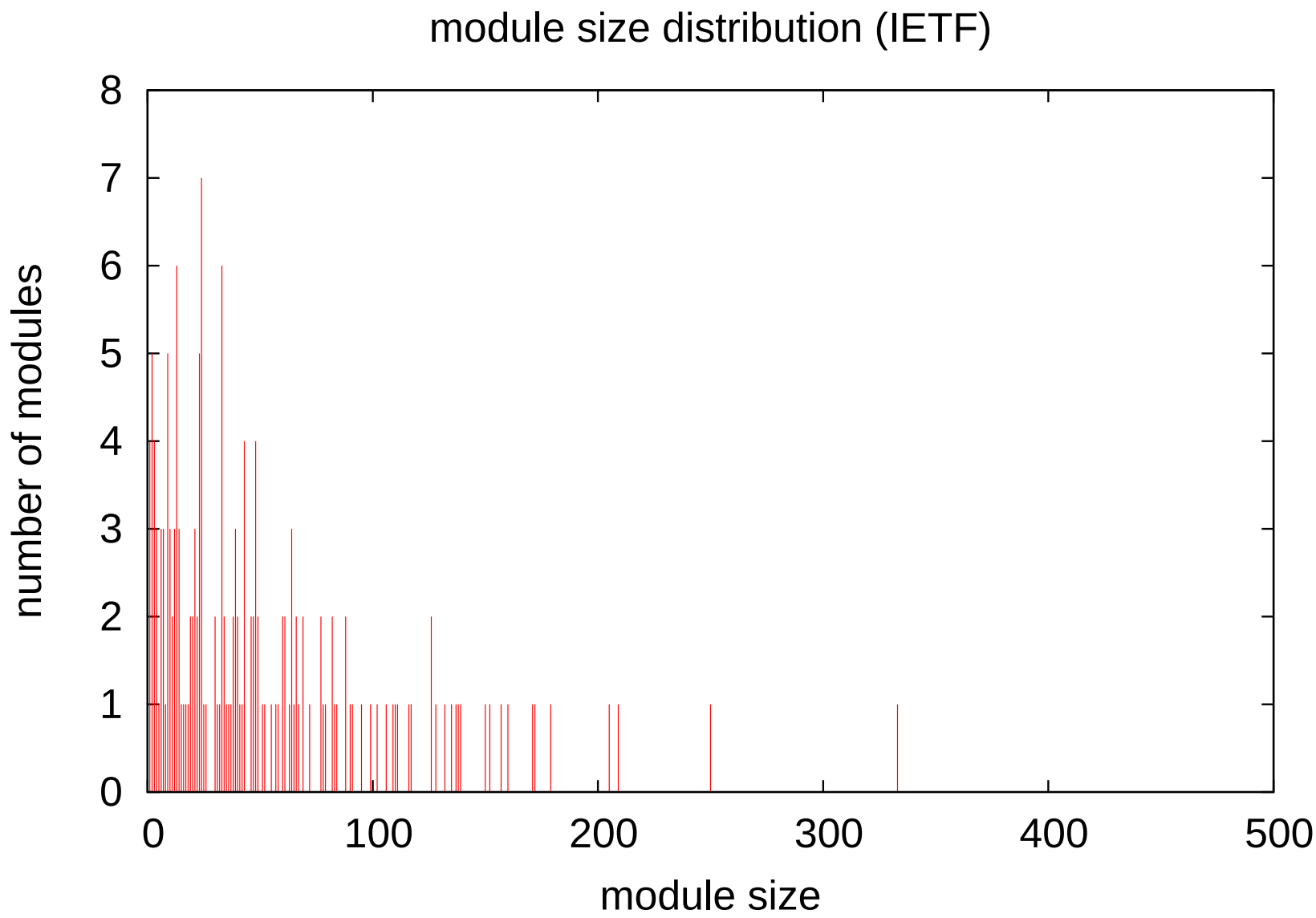
MIB Module Revision Frequency



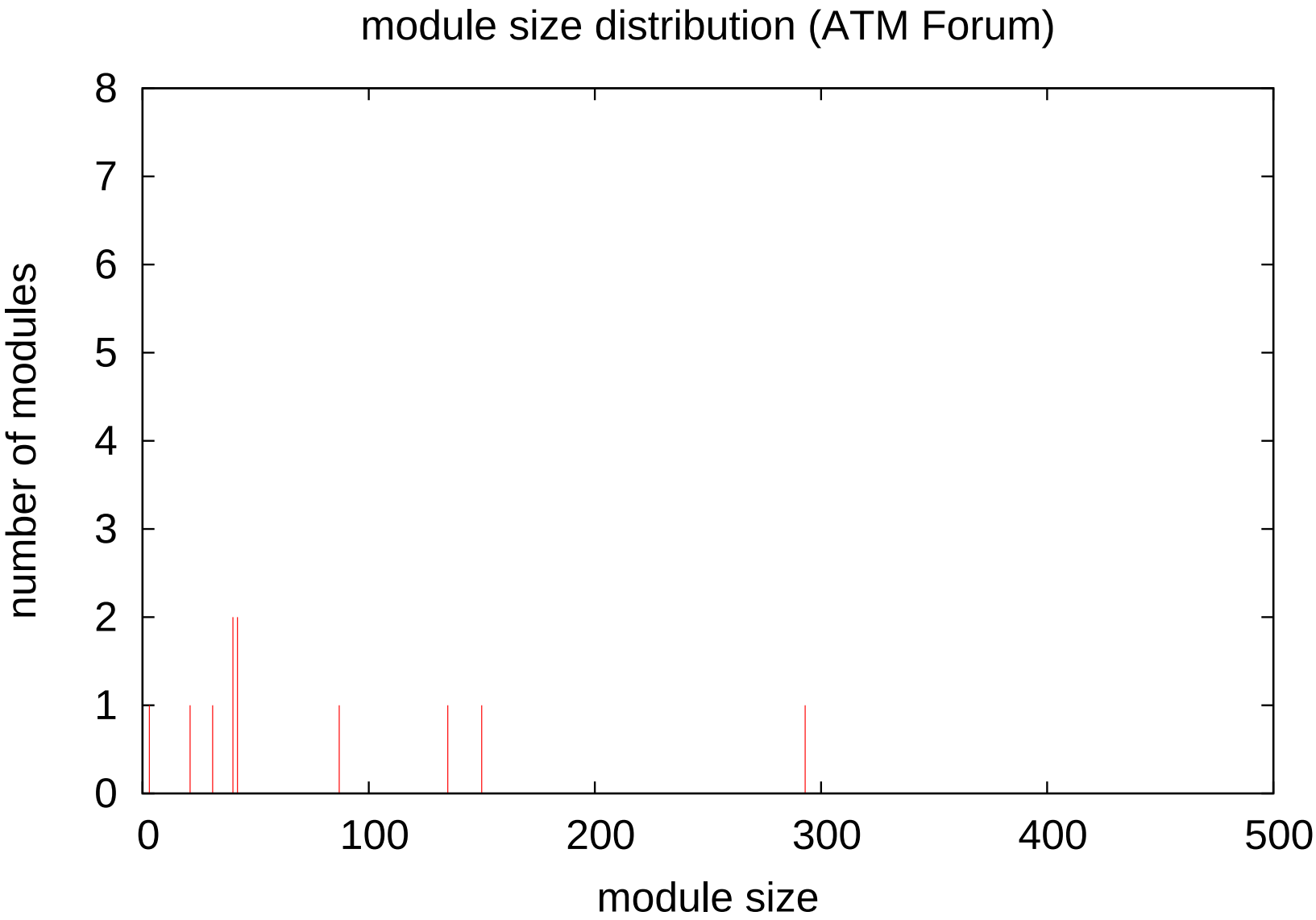
Module Size Distribution



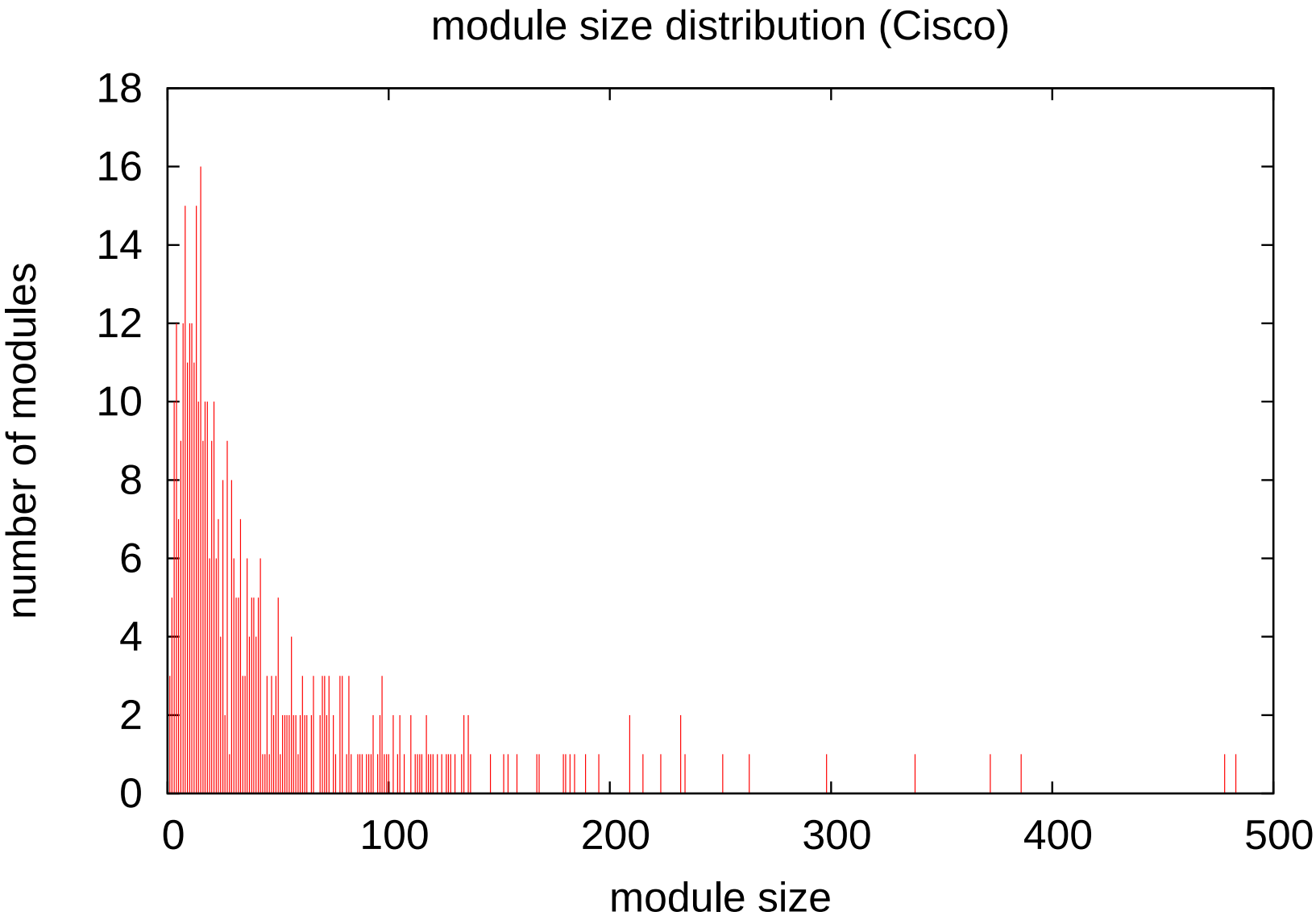
IETF Module Size Distribution



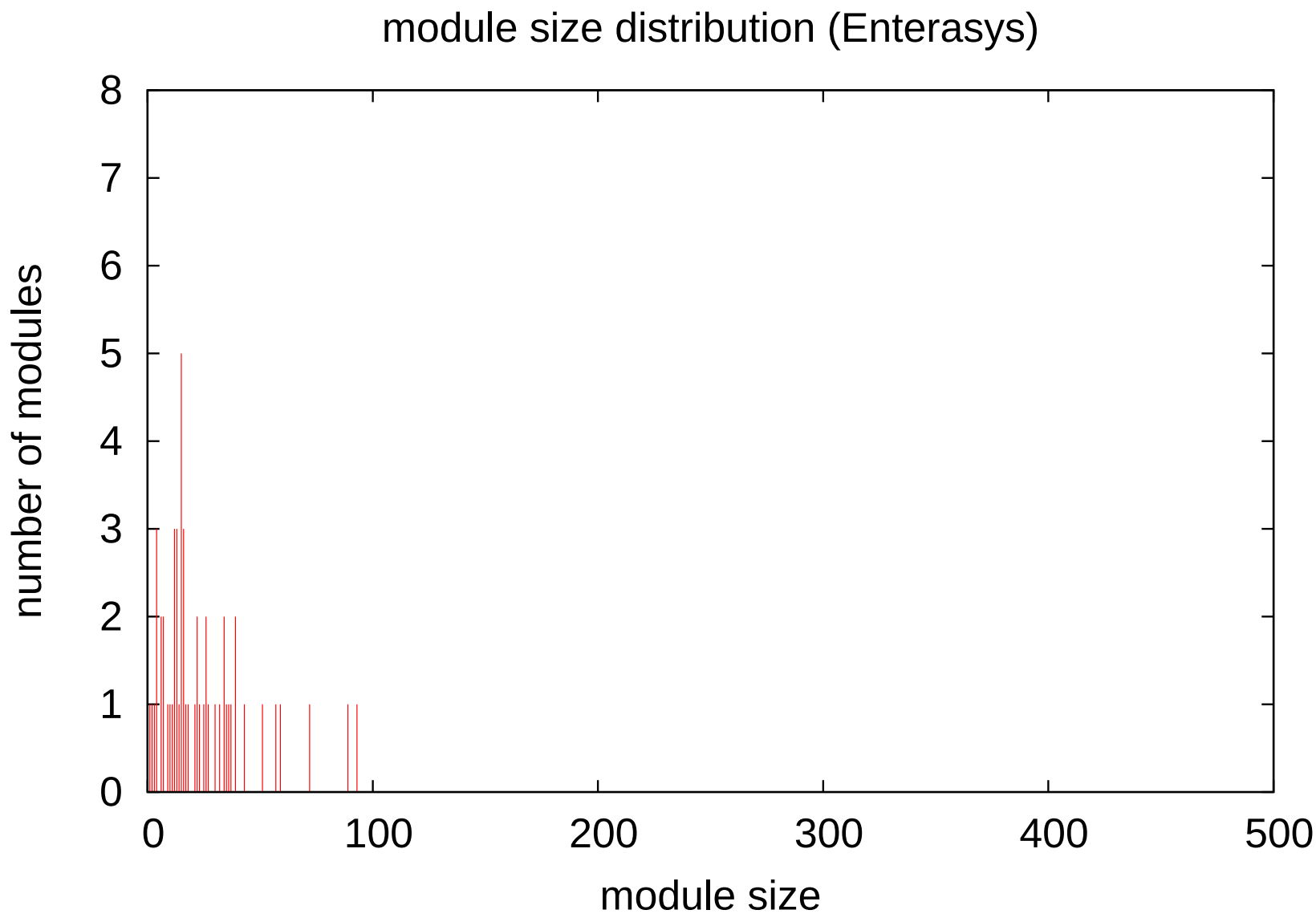
ATMF Module Size Distribution



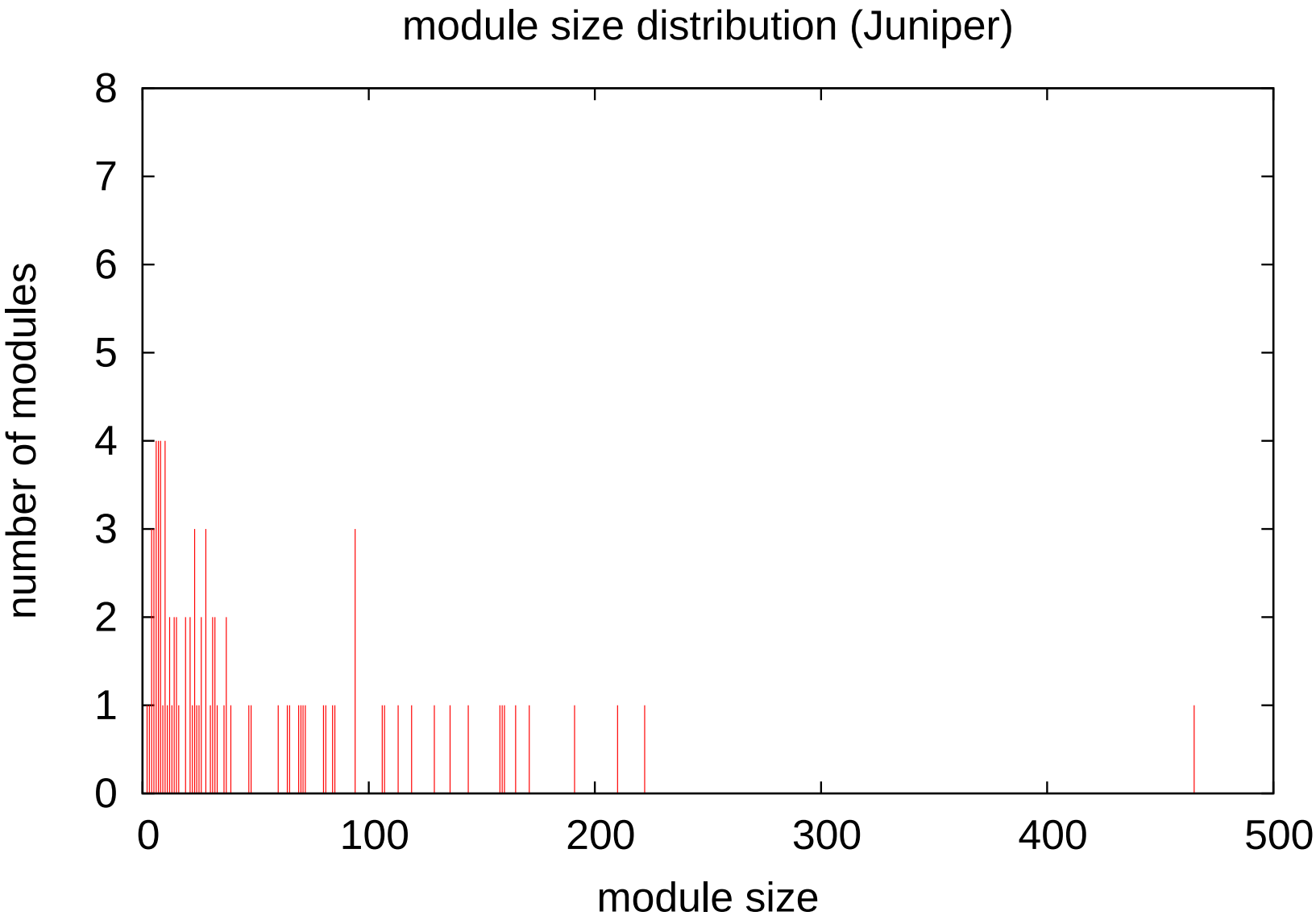
Cisco Module Size Distribution



Enterasys Module Size Distribution



Juniper Module Size Distribution



Data Type Usage

ALL		IETF		ATM Forum	
Integer32	18.2%	Integer32	18.5%	Integer32	20.3%
Counter32	15.6%	Counter32	16.1%	Counter32	7.7%
Enumeration	8.7%	Enumeration	8.4%	Gauge32	7.7%
Unsigned32	8.2%	Unsigned32	6.6%	Enumeration	7.6%
TruthValue	4.7%	SnmpAdminString	3.4%	TruthValue	7.0%
DisplayString	3.1%	OctetString	3.0%	Unsigned32	4.6%
Gauge32	3.1%	RowStatus	2.6%	RowStatus	3.3%
IpAddress	2.5%	DisplayString	2.4%	PnniNodeId	3.2%
RowStatus	2.4%	IpAddress	2.4%	PnniNodeIndex	2.5%
SnmpAdminString	2.2%	Gauge32	2.3%	PnniPortId	2.0%
Counter64	2.2%	TruthValue	2.2%	PnniLevel	1.9%
OctetString	1.9%	InterfaceIndex	1.7%	InterfaceIndex	1.7%
InterfaceIndex	1.4%	TimeStamp	1.5%	AtmLaneAddress	1.7%
TimeStamp	1.2%	Counter64	1.3%	AtmAddr	1.1%

Data Type Usage (cont.)

Cisco		Enterasys		Juniper	
Integer32	18.3%	Integer32	18.2%	Integer32	16.9%
Counter32	17.2%	Counter32	10.4%	Counter32	10.3%
Enumeration	9.3%	Unsigned32	9.3%	Unsigned32	7.8%
Unsigned32	9.1%	Enumeration	8.2%	TruthValue	7.7%
TruthValue	4.9%	TruthValue	4.5%	Enumeration	7.3%
Gauge32	3.3%	OctetString	4.0%	IpAddress	6.3%
DisplayString	2.9%	EnabledStatus	3.9%	DisplayString	5.7%
SnmpAdminString	2.2%	SnmpAdminString	2.8%	Counter64	5.5%
RowStatus	2.0%	InterfaceIndex	2.6%	InterfaceIndex	3.9%
Counter64	2.0%	MacAddress	2.6%	RowStatus	3.8%
IpAddress	1.8%	DisplayString	2.3%	Gauge32	3.0%
TimeStamp	1.4%	TimeTicks	2.2%	OctetString	1.7%
OctetString	1.4%	RowStatus	2.0%	JuniEnable	1.6%
InetAddress	1.2%	IpAddress	1.9%	InterfaceIndexOrZero	0.8%

Base Type Usage

Modules	Integer32	Unsigned32	Unsigned64	OctetString	ObjectId	Enum	Bits
All	21.5	35.5	3.3	15.0	0.6	23.3	0.9
IETF	22.3	36.5	2.7	16.6	1.9	18.9	1.2
ATM	32.5	27.0	0.0	11.2	0.4	28.1	1.0
Cisco	20.8	38.1	2.8	13.7	0.2	23.6	0.7
Enterasys	18.3	26.7	0.8	22.0	0.2	28.6	3.4
Juniper	21.5	25.6	7.3	17.0	0.2	27.8	0.6

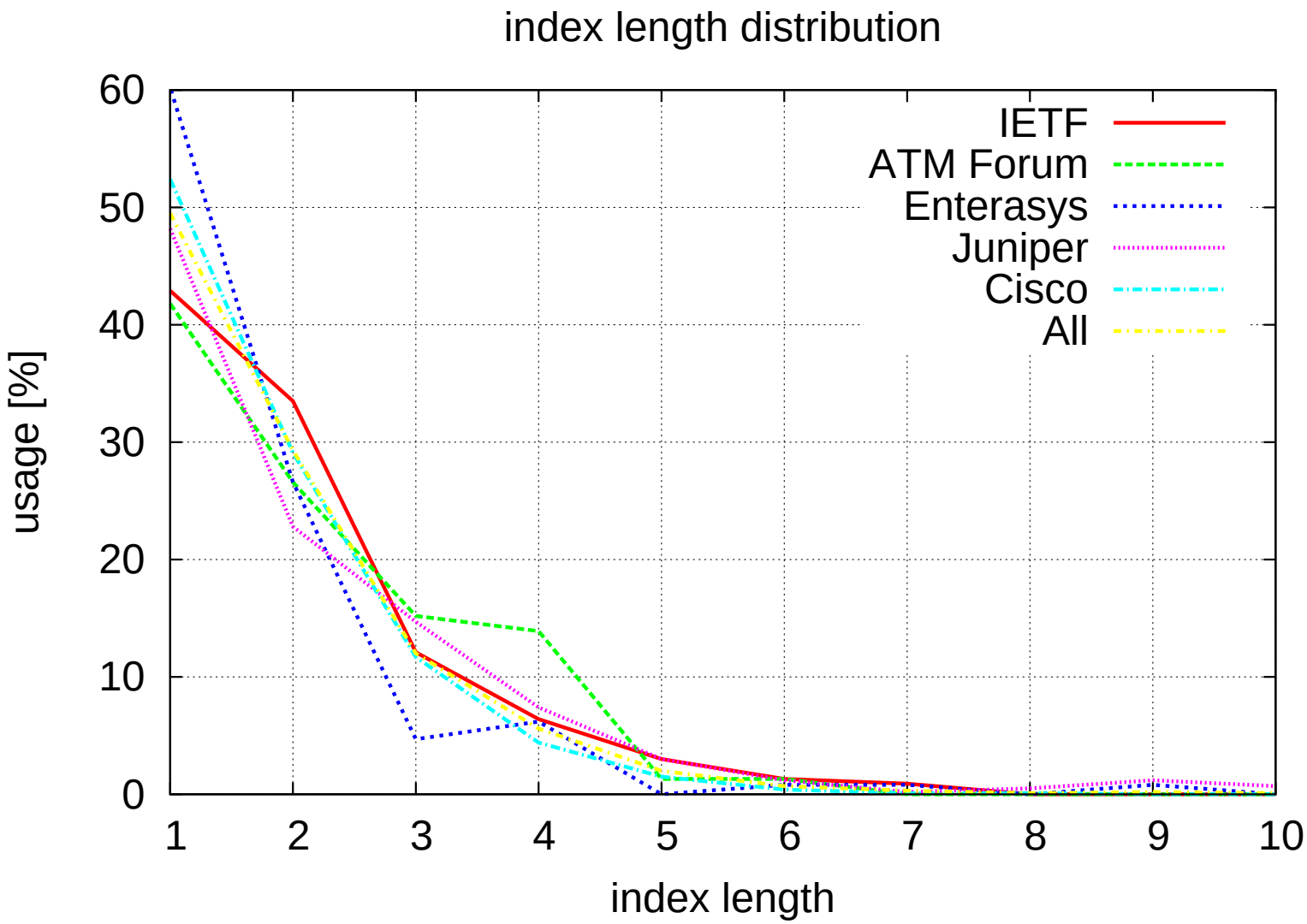
- Looking at all MIB modules, more than 83.6% of all variables are encoded as ASN.1 `INTEGER` values
- Close to 80% are 32-bit integer values that fit into 1-5 bytes
- Note that the actual usage distribution might be different

Maximum Access Distribution

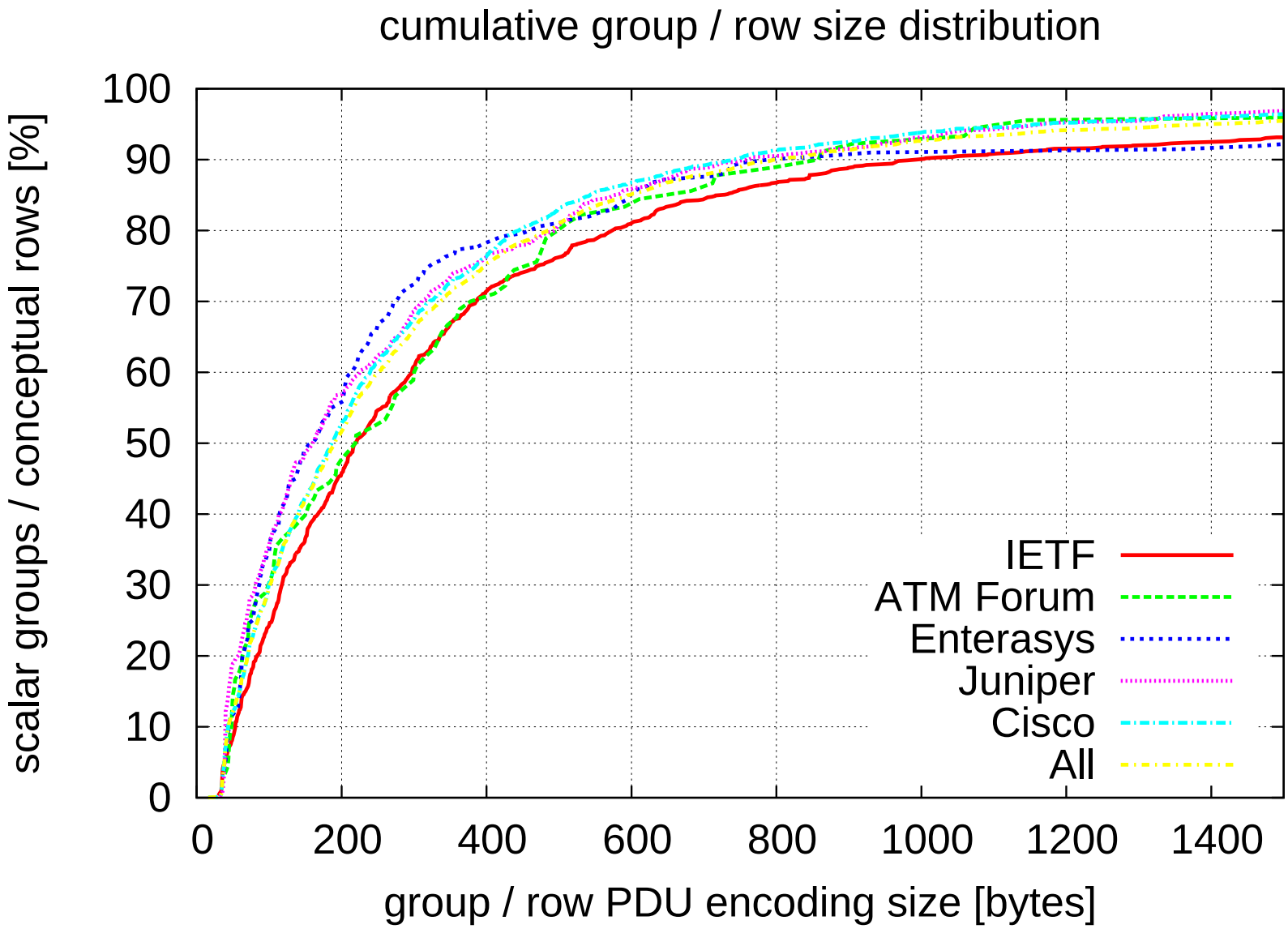
Module Set	read-write	read-only	notify	no-access
All Modules	32.6%	58.7%	0.5%	8.3%
IETF	29.1%	61.3%	0.2%	9.4%
ATM Forum	43.4%	46.9%	0.2%	9.4%
Cisco Systems	31.6%	60.8%	0.4%	7.2%
Enterasys	40.5%	50.4%	1.3%	7.9%
Juniper	38.8%	49.6%	1.0%	10.7%

- Cisco and IETF modules have a similar ratio of read-only and read-write objects
- ATM Forum, Enterasys and Juniper have significantly more writable objects defined

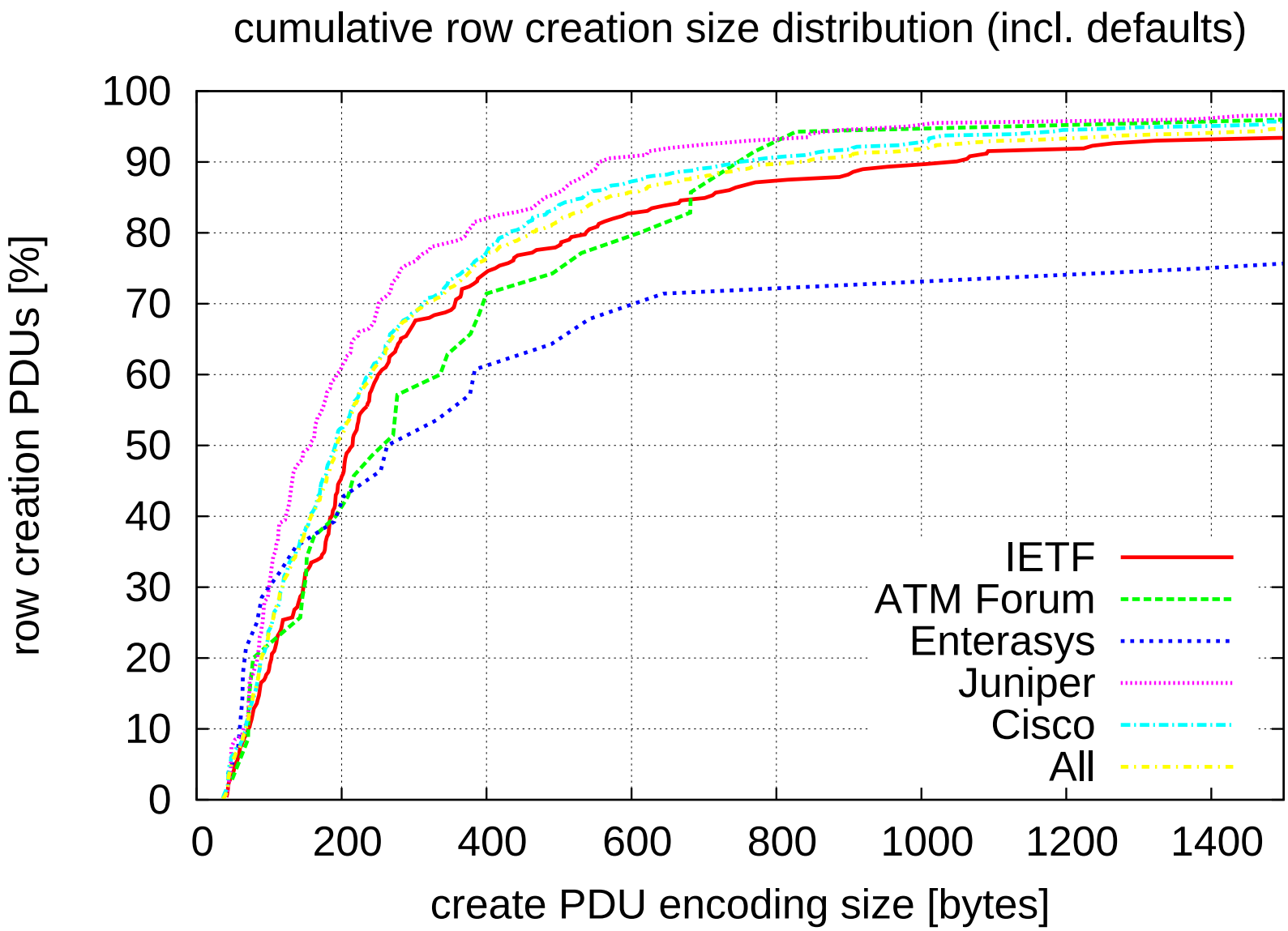
Index Length Distribution



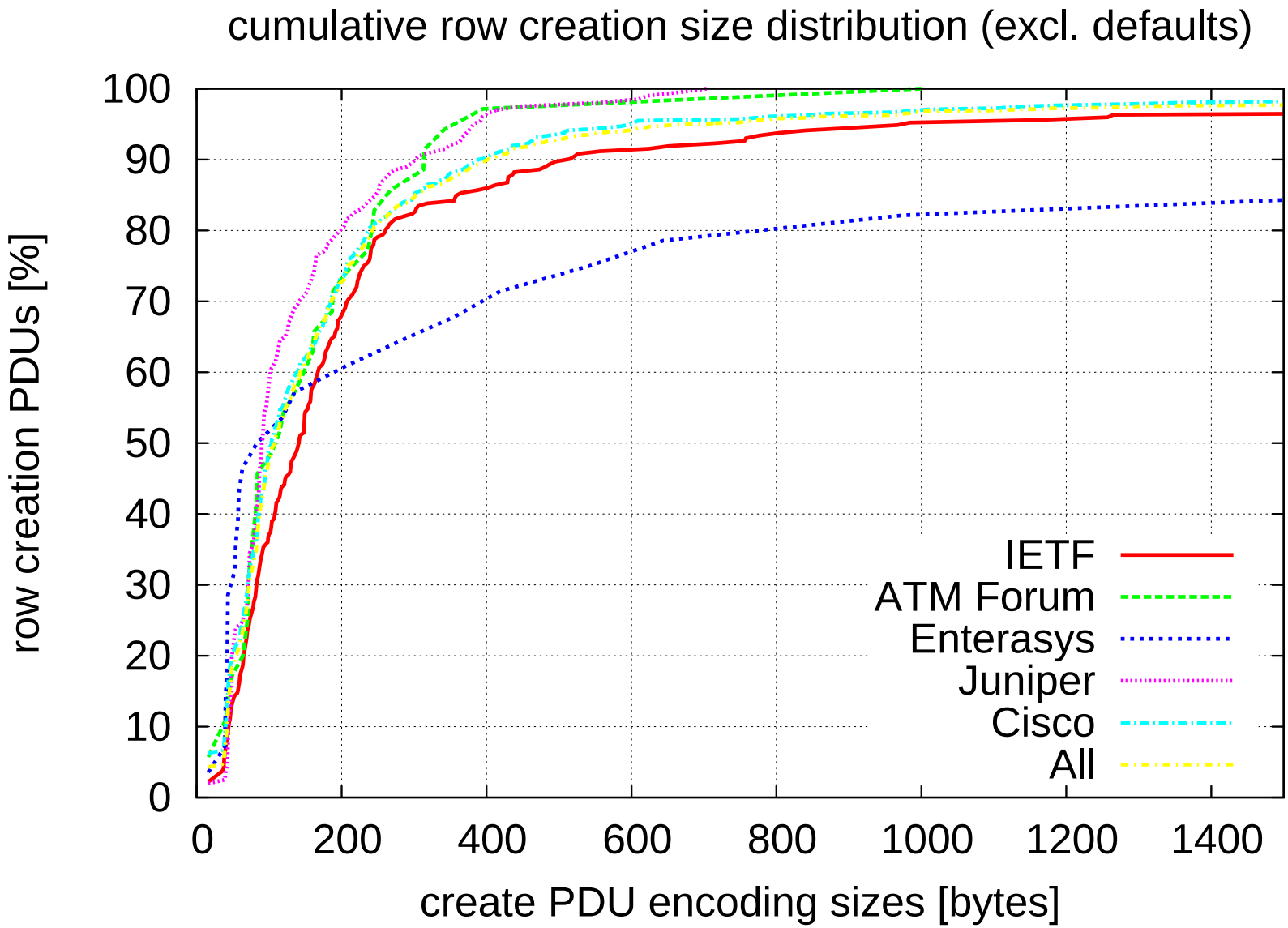
Row Encoding Size Distribution



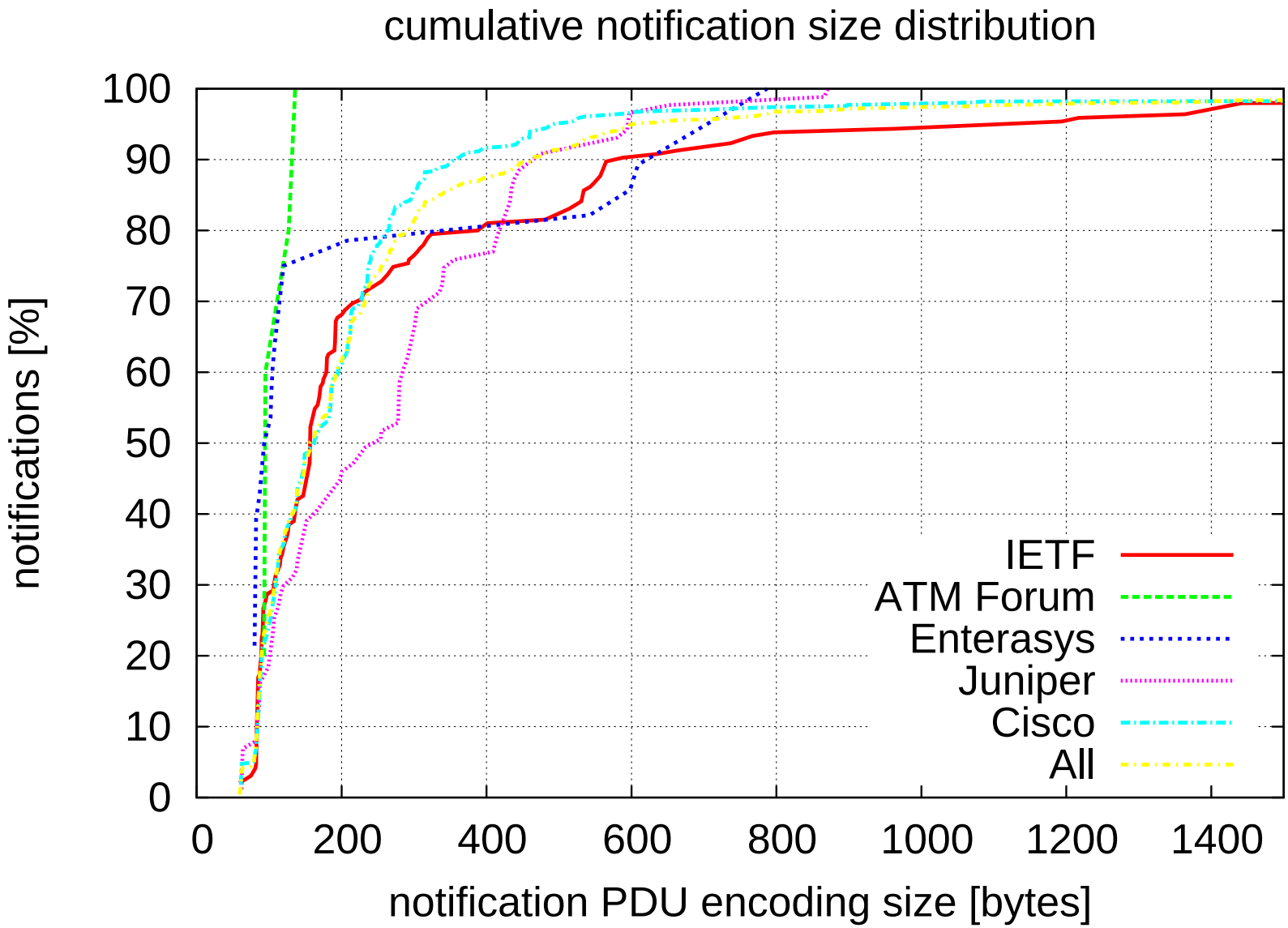
Create Encoding Size Distribution



Create Encoding Size Distribution



Notification Encoding Size Distribution



Referenced MIB Modules

ALL		IETF		ATM Forum	
SNMPv2-SMI	47.1%	SNMPv2-SMI	44.6%	SNMPv2-SMI	39.5%
SNMPv2-TC	22.6%	SNMPv2-TC	20.3%	SNMPv2-TC	32.2%
IF-MIB	6.8%	IF-MIB	6.8%	IF-MIB	10.8%
INET-ADDRESS-MIB	3.3%	PerfHist-TC-MIB	3.2%	ATM-TC-MIB	7.0%
CISCO-ITP-TC-MIB	2.2%	RMON2-MIB	2.8%	ATM-MIB	3.2%
. . .	18.0%	. . .	22.3%	. . .	7.3%

Cisco		Enterasys		Juniper	
SNMPv2-SMI	48.4%	SNMPv2-SMI	37.2%	SNMPv2-SMI	49.4%
SNMPv2-TC	22.0%	SNMPv2-TC	28.2%	SNMPv2-TC	26.6%
IF-MIB	5.8%	IF-MIB	8.7%	IF-MIB	10.2%
INET-ADDRESS-MIB	4.5%	P-BRIDGE-MIB	8.7%	Juniper-TC	7.0%
CISCO-ITP-TC-MIB	3.8%	INET-ADDRESS-MIB	5.1%	HCNUM-TC	1.3%
. . .	15.5%	. . .	12.1%	. . .	5.5%

Referenced Variable Definitions

ALL		IETF		ATM Forum	
ifIndex	49.6%	ifIndex	43.0%	ifIndex	56.7%
entPhysicalIndex	4.5%	hrDeviceIndex	4.9%	atmVclVpi	16.7%
ifIndex	4.0%	applIndex	3.9%	atmVclVci	13.3%
vsanIndex	2.8%	protocolDirLocalIndex	3.9%	atmVplVpi	10.0%
atmVclVpi	2.5%	dot1dBasePort	2.6%	ifIndex	3.3%
. . .	36.6%	. . .	41.7%	. . .	0%

Cisco		Enterasys		Juniper	
ifIndex	52.6%	ifIndex	55.6%	ifIndex	40.5%
entPhysicalIndex	7.0%	dot1dBasePort	13.9%	atmVclVci	6.0%
ifIndex	5.1%	dot1xPaePortNumber	11.1%	atmVclVpi	6.0%
vsanIndex	4.4%	dot1qVlanIndex	5.6%	atmVplVpi	3.6%
cgspInstNetwork	2.7%	dot1dStpPort	2.8%	acctngSelectionIndex	2.4%
. . .	29.2%	. . .	11.0%	. . .	41.5%

Comments?

- Any interesting metrics missing?
- Any other MIB module sets to consider?
- Ideas for better estimation of encoding sizes?
- . . .