Linuxflow: A High Speed Backbone Measurement Facility

ZhiChun Li (lizc@serv.edu.cn)

Hui Zhang (hzhang@cernet.edu.cn)

CERNET, Tsinghua Univ, China

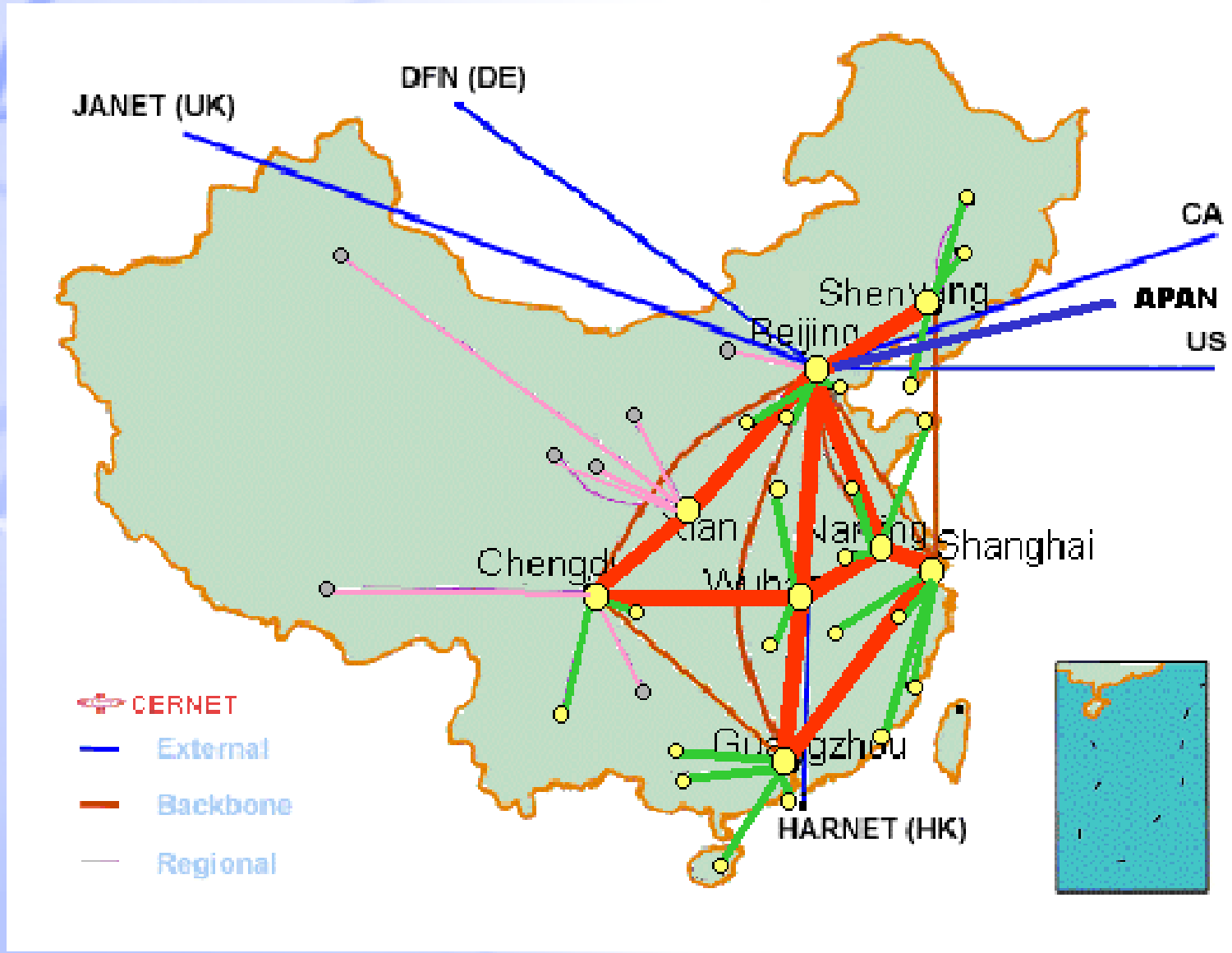
Outline

- Introduction to CERNET
- Motivation of Linuxflow
- Traffic collection method and environment
- Detailed approach: Linuxflow design
- Performance evaluation
- Applications based on Linuxflow
- Conclusions and Future work

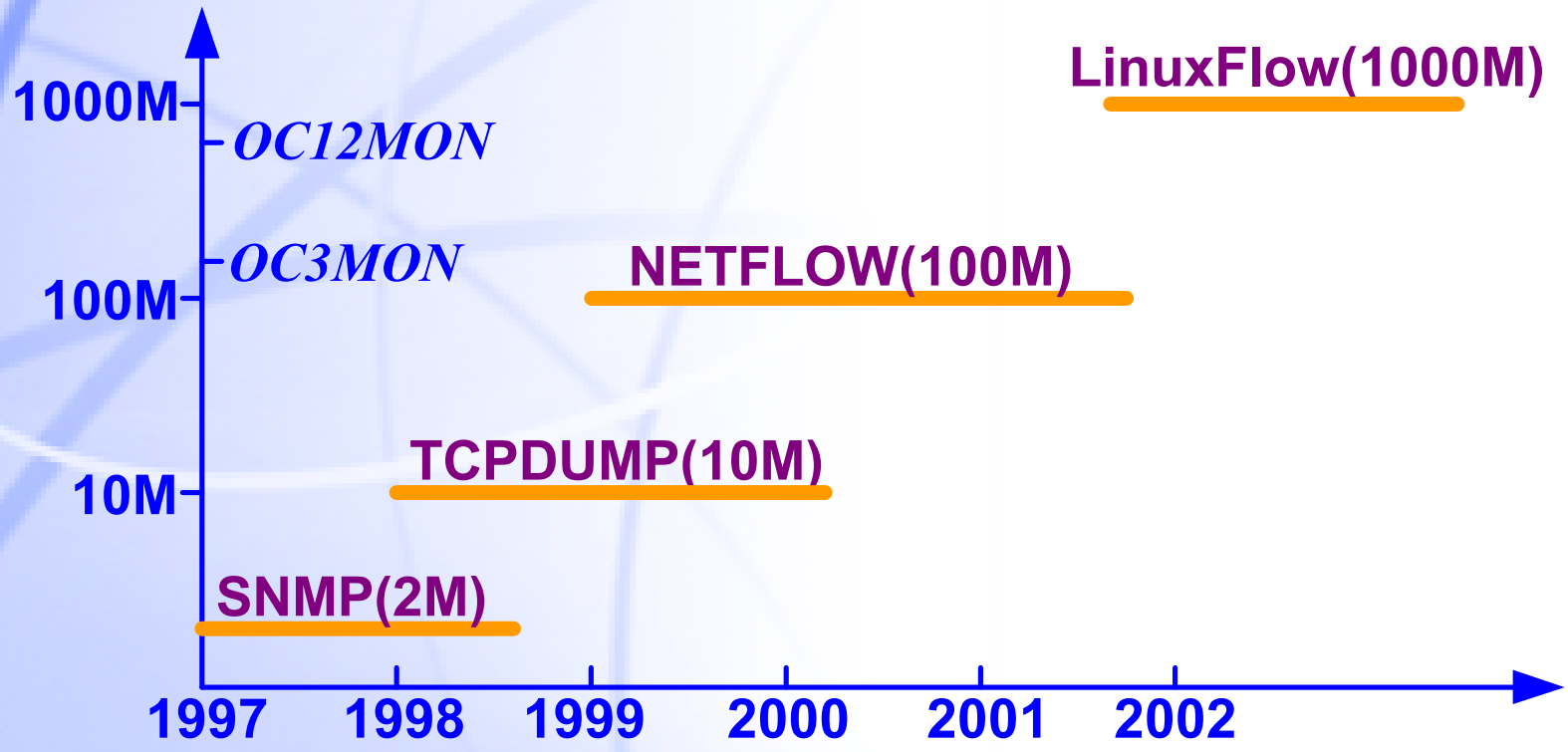
Introduction to CERNET

- One of the most significant and largest networks in Asia Pacific region
- 1000+ universities and education institutions
- 1.2 millions hosts
- 10 millions users
- Over 60 OC-48 and OC-3 links
- CIDR rank 35 in the world(88.625 /16 networks)

CERNET Topology



Network measurement facilities used in CERNET



new requirements of CERNET stimulate our approach to appear

- High-speed usage-based accounting and billing for "transatlantic" traffic (OC3 up to Gigabit)
- IP MONitoring Infrastructure for CERNET (40+ agents deployed on backbone)
- CERNET Network Management System
- User behavior analysis and traffic data mining for network security

Motivation of Linuxflow

- Measure gigabit or even more higher speed links
- Provide both packet level and flow level fine-grained information
- Base on commodity hardware
- Self-develop inexpensive software solution

How Linuxflow work?

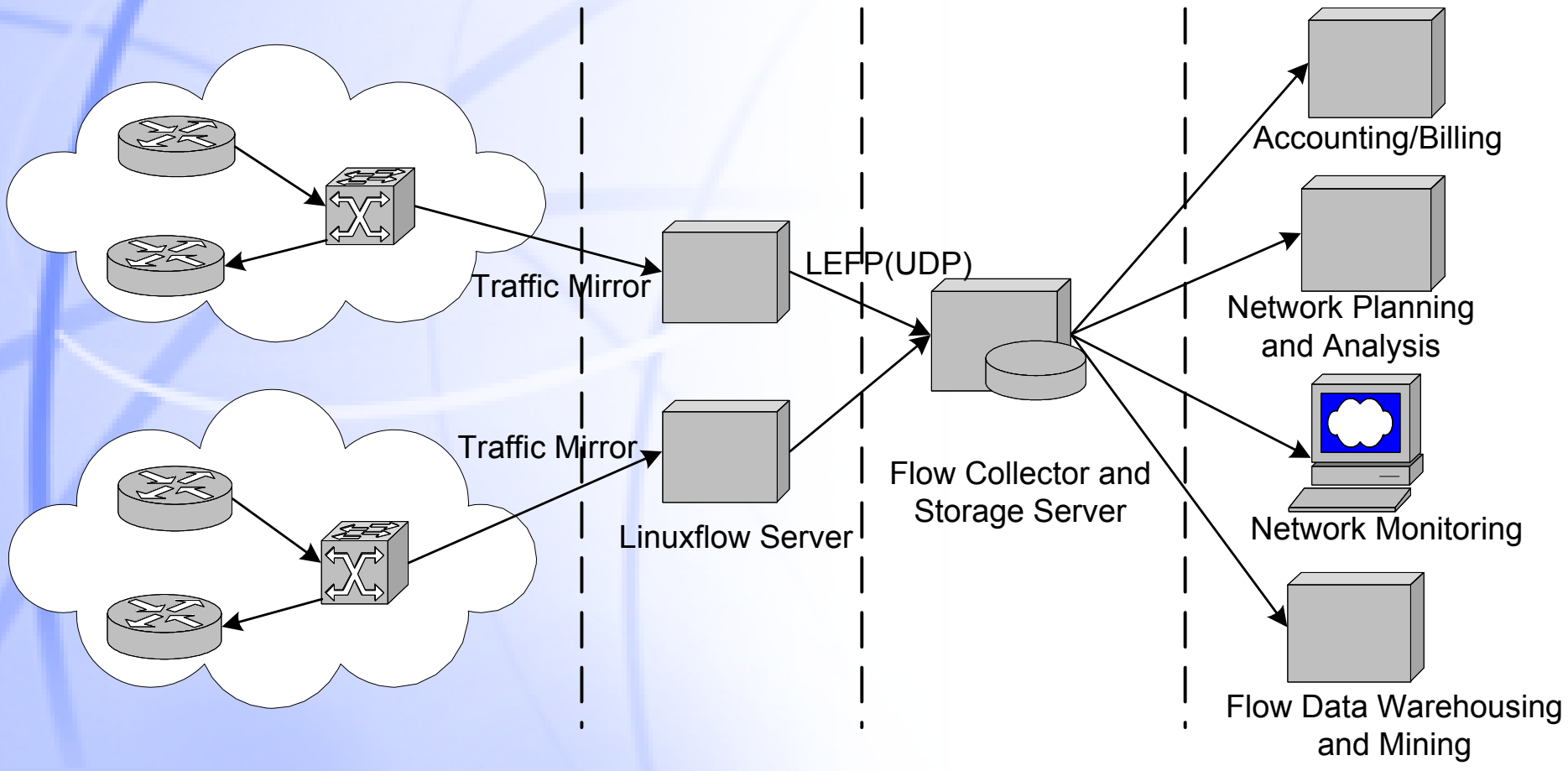
- 3 components: Linuxflow Agent, Linuxflow Collector, Linuxflow Manager.
- Agents run on a Linux box to sniff the traffic
  - self-designed special standalone network packet capture protocol stack
  - multi-thread flow aggregation daemon
- Collectors collect flows from different Agents, interfacing applications
- Managers control and monitor the status of each Agent and Collector

Methods of sniffing

- Insert a hub in network link, all ports of the hub can get a copy of data (10/100M half-duplex)
- Port or interface span, by means of which the traffic from one or more interfaces on a network switch can be mirrored to another one(s)
- Network tap, such as optical splitter

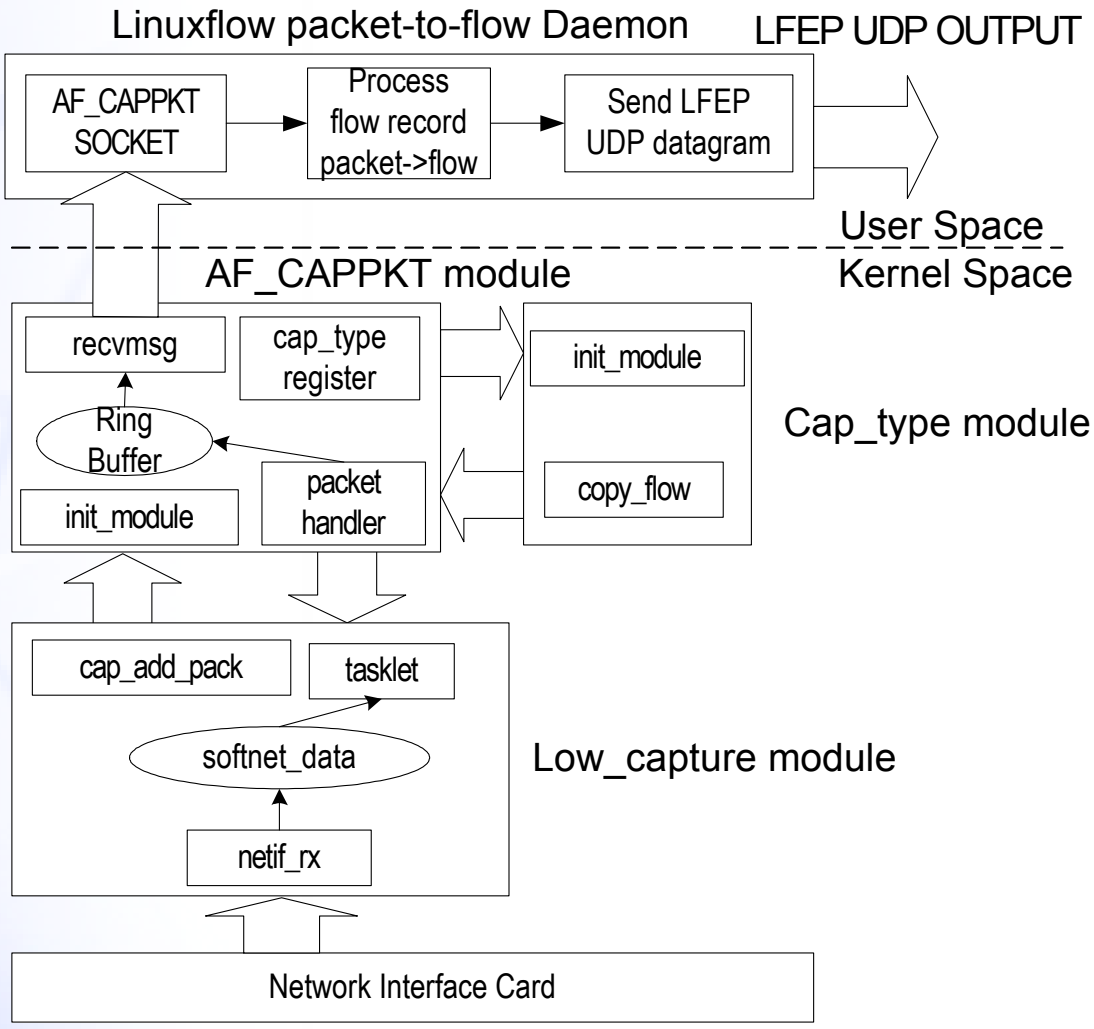
Traffic collection network environment

- Common environment



Detailed approach: Linuxflow Agent structure

- Based on Linux Kernel 2.4.x
- 3 modules implement the capture protocol stack
- Multi-thread flow aggregation daemon



Detailed approach: packet level capture

- **Standalone packet capture protocol stack**
  - Low capture module
    - redefine the netif_rx kernel symbol and define the tasklet to send the packet (skbuff) to our packet capture stack.
  - AF_CAPPKT module
    - This module registers AF_CAPPKT protocol family to Linux kernel, and implements the AF_CAPPKT socket
  - cap_type module
    - provides us with the ability to implement different filter to get selected fields

Detailed approach: packet level capture

- **Filters already defined**
  - Selective header fields used for stream level flow aggregation
  - All IP header and TCP/UDP/ICMP/IGMP header fields
  - Collect all IP packets
- **API in user space**
  - Open AF_CAPPKT socket:
    - sock = socket (AF_CAPPKT, CAP_COPY_FLOW, ntohs(ETH_P_IP))
  - Read data structure through the socket
- **Kernel Time-stamping**
  - Using kernel function do_gettimeofday() to get microsecond level timestamp (8 bytes)

Detailed approach: packet level capture

- **Factors influencing the packet level capture performance**
  - Network Bandwidth vs. NetCard capability
  - Network Bandwidth vs. PCI Speed
    - All packets will go through PCI bus, PCI133 (133Mhz 64bits) may handle OC48
  - Packets Per Second vs. NetCard Performance
    - NetCard RX buffer vs. CPU interrupt frequency
  - Packets Per Second vs. CPU Performance
- **NetCard driver level tuning to improve performance**

Detailed approach: flow level aggregation

- **flow definition**
  - RTFM flows are arbitrary groupings of packets defined only by the attributes of their endpoints (address attributes)
    - 5-tuple stream level (individual IP sessions)
    - 2-tuple IP-pair level (traffic between two host)
    - pair of netblocks(traffic between two IP address blocks)
  - Cisco NetFlow flows are stream level microflow
  - Linuxflow Agents produce stream level flow too
  - Linuxflow Collectors aggregate to high level flow

Detailed approach: flow level aggregation

- **Two types of timeout definition: active timeout and inactive timeout**
- **Stream level flow termination**
  - Flows which have been idle for a specified time (*inactive timeout*) are expired and removed from the flow table.
  - Long lived flows are reset and exported from the flow table, when they have been active for a specified time (*active timeout*).
  - TCP connections which have reached the end of byte stream (FIN) or which have been reset (RST)

Detailed approach: flow level aggregation

- **Long lived flow fragmentation**
  - Long lived flows are reset and exported from the flow table, when they have been active for a specified time (*active timeout*)
  - Consecutive packets of a long lived flow which has been exported will make up a flow with a *cont flag*, this can notify collector “I am not a new one”
  - In flow statistic analysis, the flow with *cont flag* will not count in new flow but accumulate to old long lived flow

Detailed approach: flow level aggregation

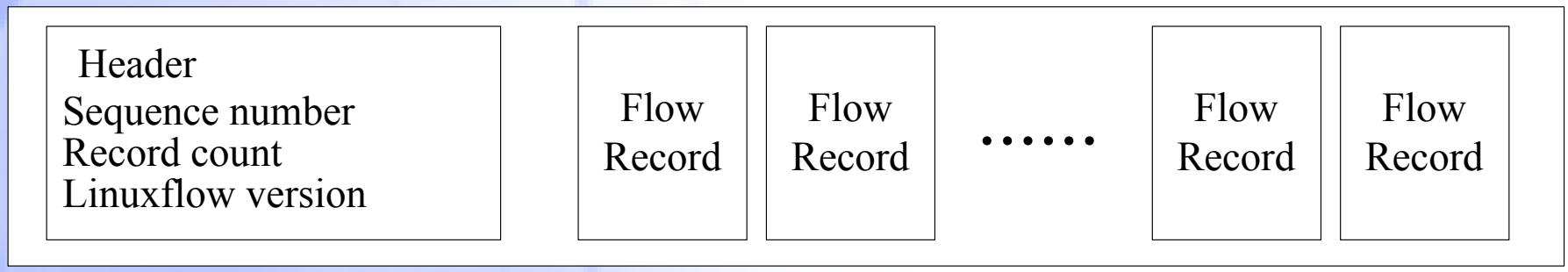
- **Multi-thread flow aggregation pipeline**
  - Reading thread: reading packet data from kernel to user space, buffering data
  - Processing thread: aggregating packet data to flow record, using packet classification algorithm, such as hash
  - Sending thread: assembling flow record into LEFP UDP packet and sending it to Linuxflow Collector for further analysis.

Detailed approach: flow level aggregation

- **Packet classification**
  - The current implementation uses hash function
    - Requires a large amount of fast memory
    - Collisions can be solved using a second hash function or a lookup tries
  - Recursive Flow Classification (RFC) is being studied, may test in next version of Linuxflow Agent

Detailed approach: LinuxFlow Export Protocol

- **Flow export protocol**
  - LinuxFlow Export Protocol (LEFP) is defined to send the flow records from Linuxflow Agent to Linuxflow Collector.
  - LEFP uses UDP protocol capable of sending flows to multiple collectors simultaneously via broadcast/multicast
  - LEFP UDP packet format is shown as follows



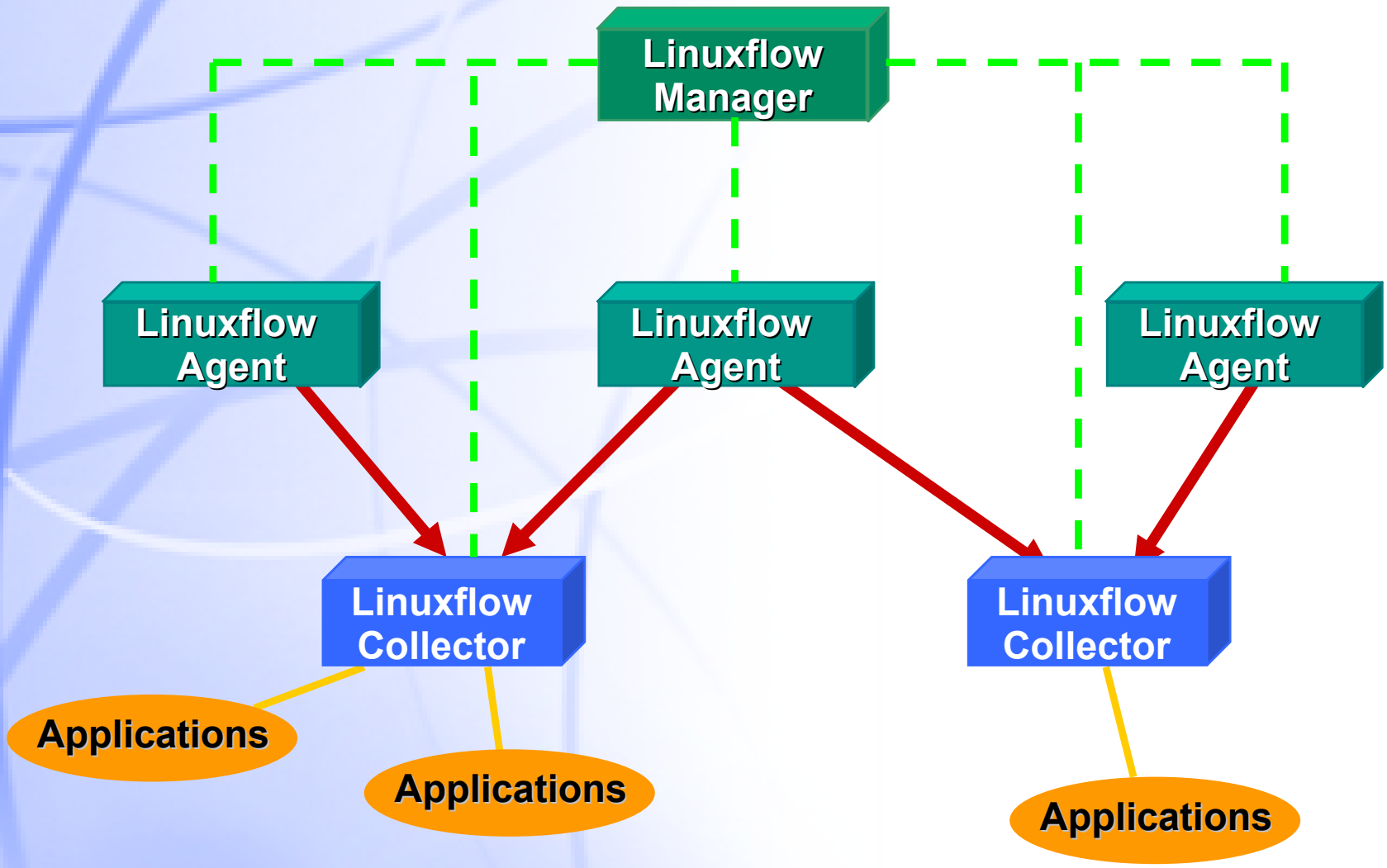
Detailed approach: Linuxflow Collector

- Collect flows from different Linuxflow Agents simultaneously
- Coexist with other flow analysis program in same machine, through IPC providing flow data sharing
  - AF_unix socket
  - Share memory

Detailed approach: Linuxflow Manager

- Refer to RTFM Flow Measurement Architecture
- Define SNMP based Linuxflow control and status MIB
- Use Linuxflow manger through SNMP to control multiple agents and collectors

Detailed approach: Linuxflow Architecture



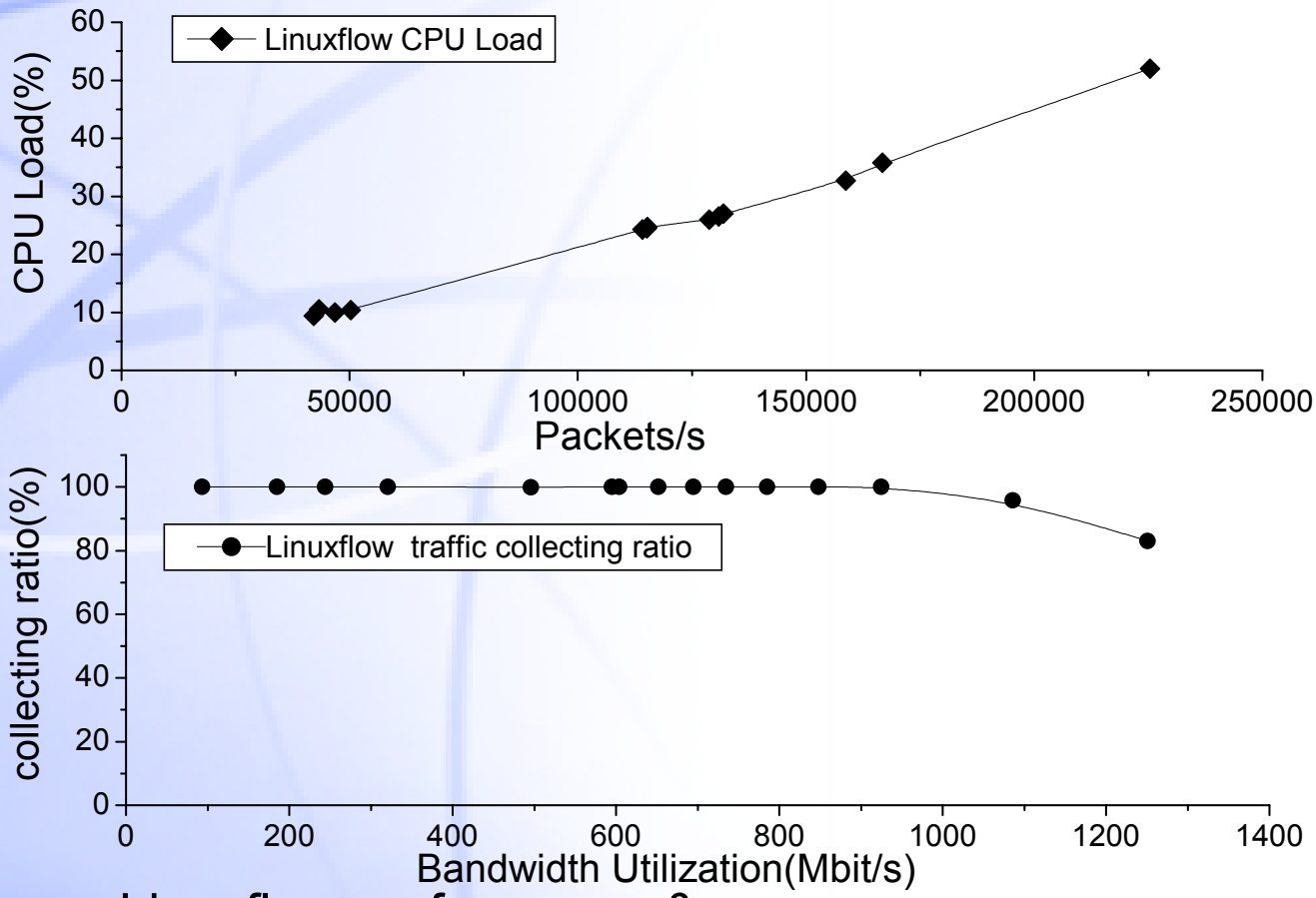
performance and accuracy test

- **Experimental environment**
  - Test Link: CERNET-CHINANET (China Telecom) Gigabit link interconnecting the biggest research network and biggest commercial network in China.
  - Test Linuxflow Agent Server:

Processor	PIII XEON 700Mhz *4
Memory	16GB DRAM
Accessory	64-bit/64MHz
Disk	35GB SCSI disk * 2
Network Card	Intel 1000BaseSX * 2

performance and accuracy test

experimental results



Linuxflow performance & accuracy curve

In commodity hardware we can get what?

- New Linuxflow Agent box capability

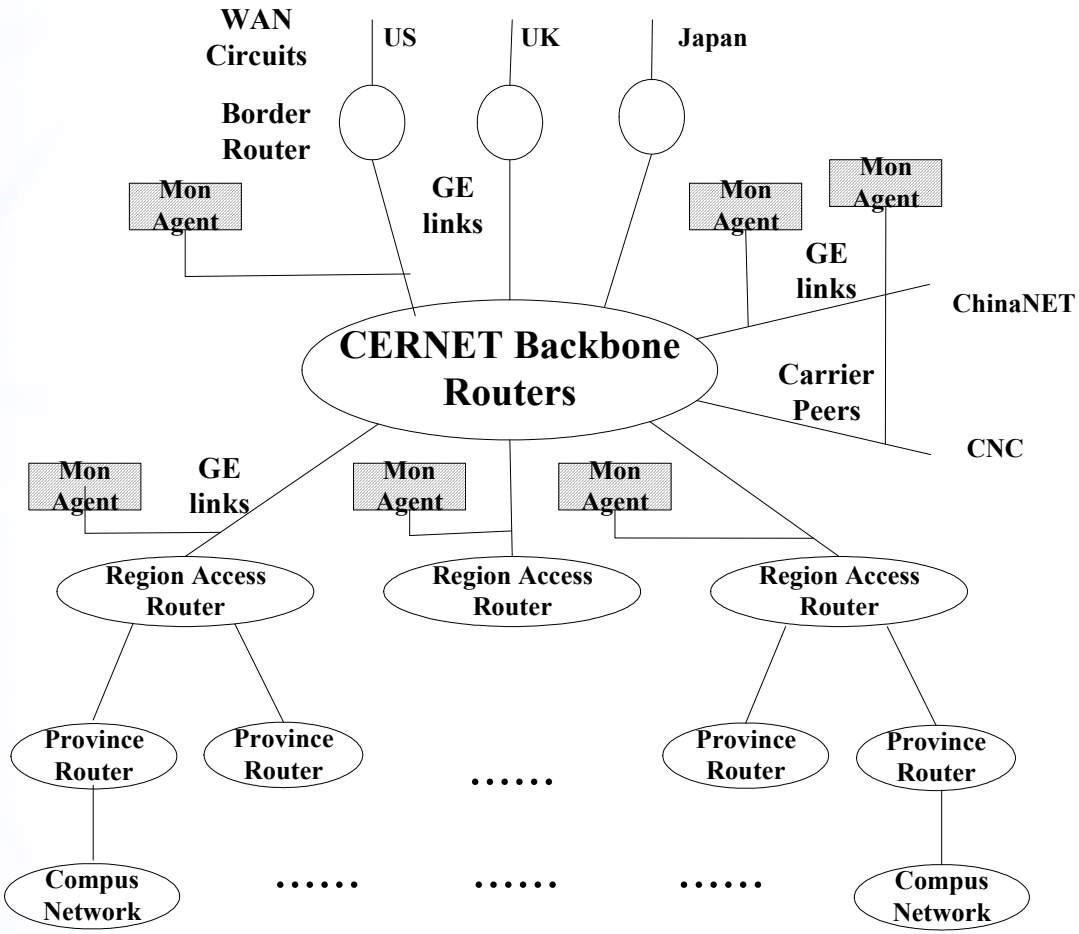
Hardware Price	$3000
Network	1.0Gbps
Processor	P4 XEON 2.0Ghz *2
Memory	64bits/333Mhz
Accessory	64bits/133Mhz
Handle Bandwidth	One box handle Gigabit Network both direction 2.0Gbps
Handle PPS	500Kpps

Applications based on Linuxflow

- IP MONitoring Infrastructure
- Accounting and Charging System
- Anomalies Detection System
- Anomalies Characterization and Traffic Data Mining

CERNET IP MONitoring Infrastructure

- Base on Linuxflow to construct monitoring agents
- Deploy monitoring agents across geographically wide area
- Measure network traffic
- Monitor network anomaly and misuse



Monitoring Agent's Capabilities

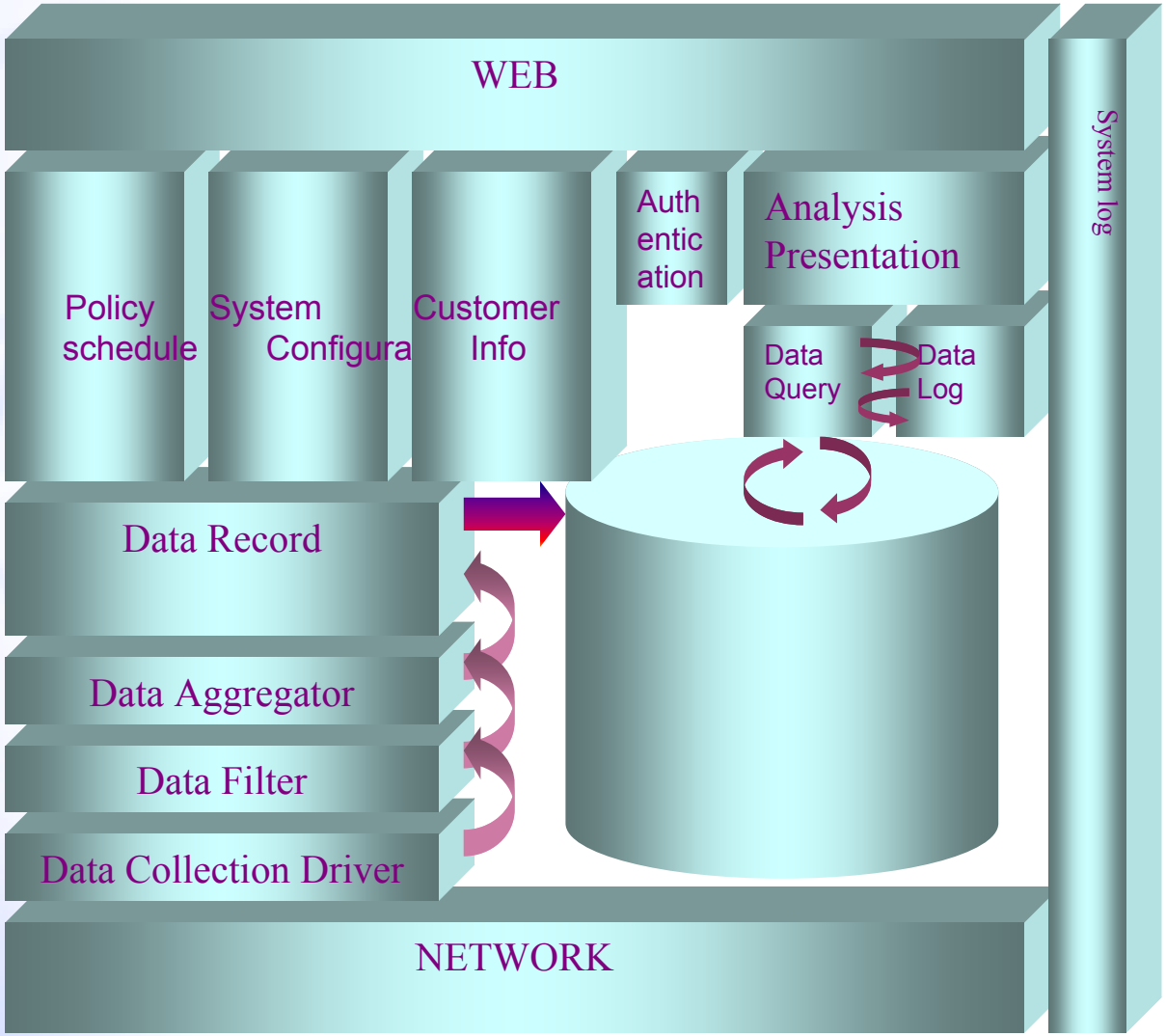
- Support data rate up to 1Gbits/sec
- Collect real-time IP packets from multiple carrier peering GigE links and regional access GigE links
- Classify ten thousands of IP packets into flows with timestamp with accurate enough fidelity
- Provide real-time measurements which characterize the status of link being monitored

Monitoring Agent's Capabilities

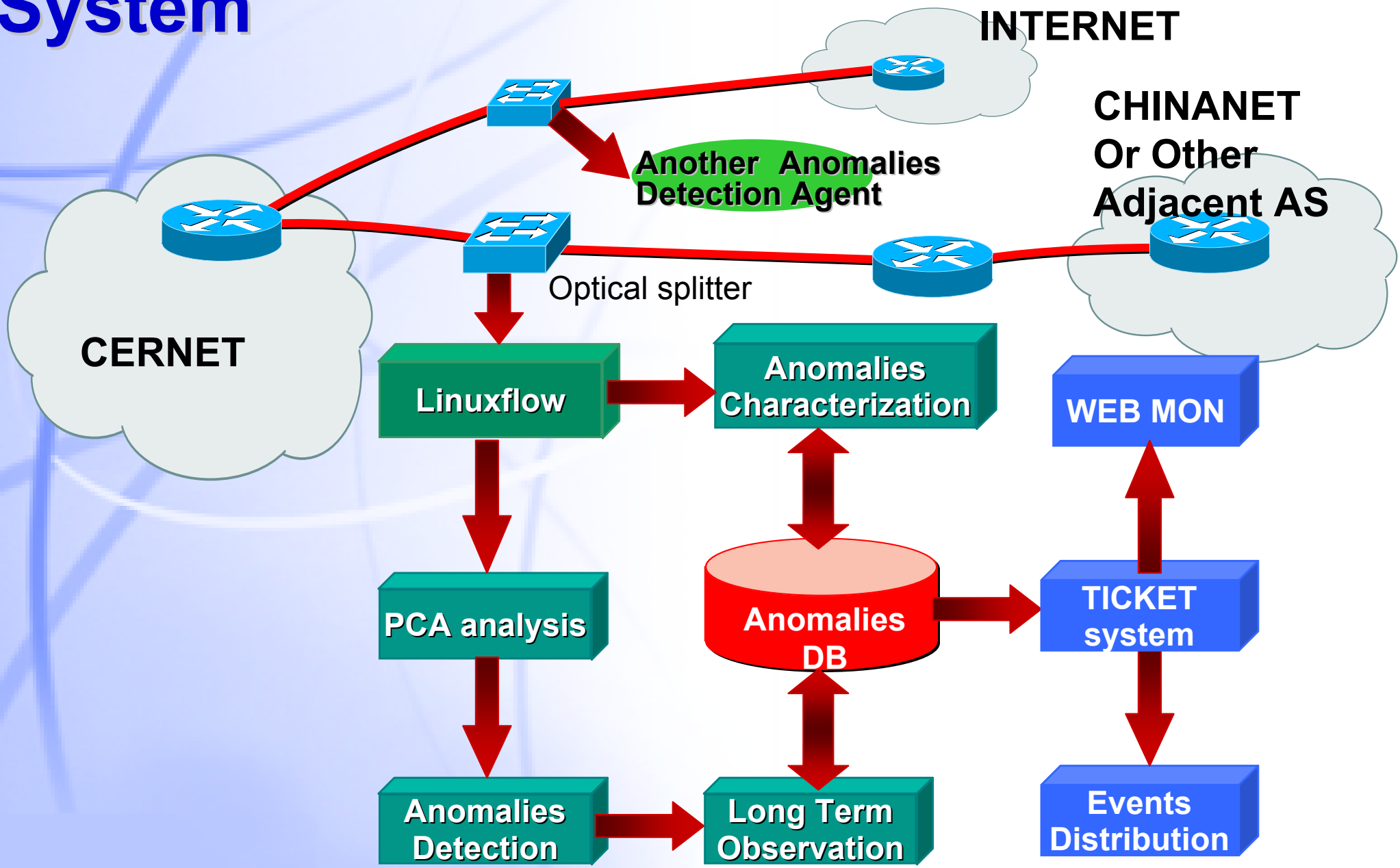
- Filter the anomaly signs according to a set of pre-defined signature in terms of multi-dimensions of network flow traffic
- Transfer the sampling IP packet data and flow data into data repository wherein previously unseen signatures are found off-line via data mining
- Provide identified records of traffic anomaly, network attacks, malicious mobile network worms

Flexible Usage-based Accounting, Charging and Billing System for CERNET

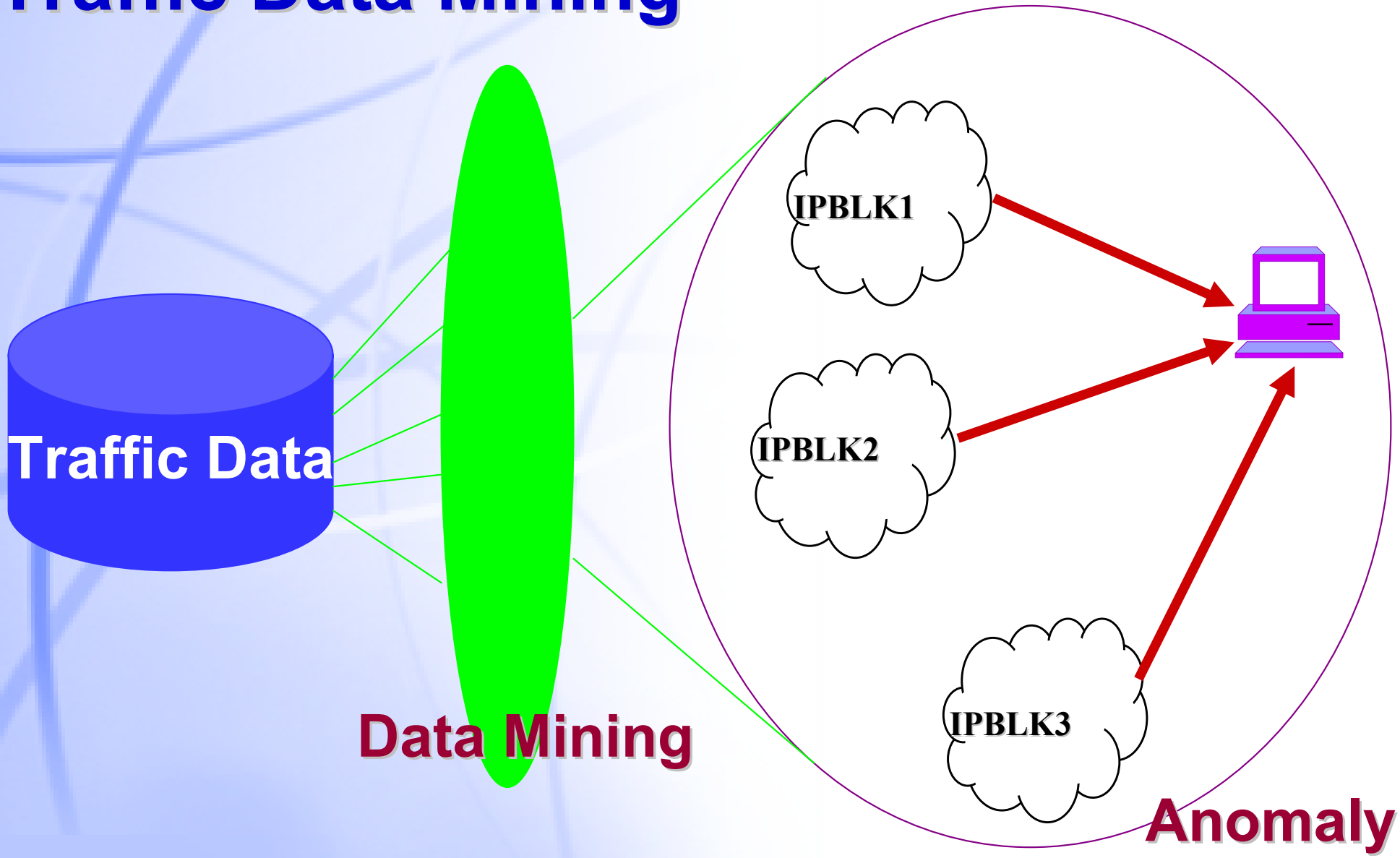
- Based on Linuxflow to collect IP packets
- Meter usage of network resources
- Charge customers by IP-accounting



CERNET Anomalies Detection System

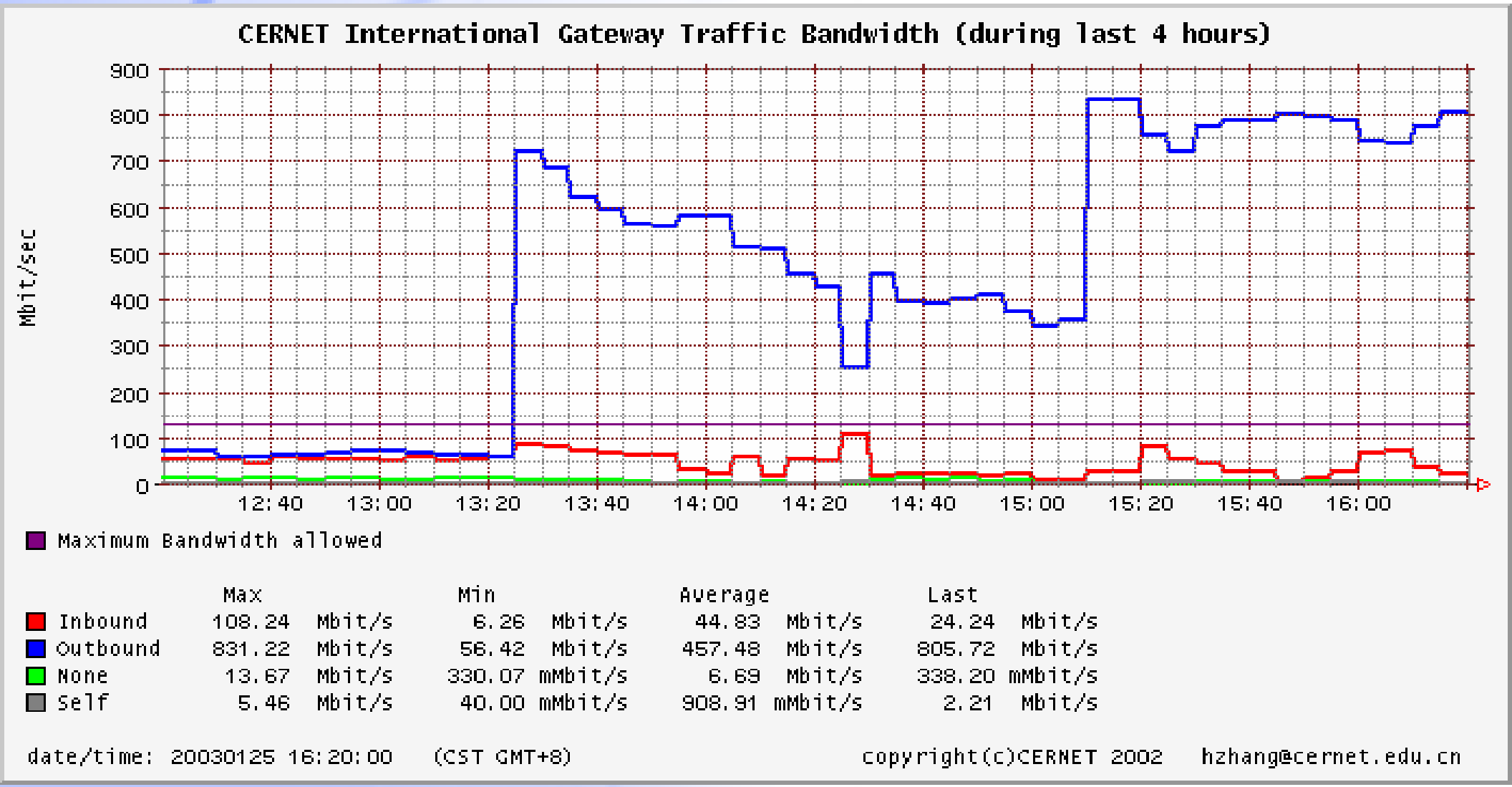


Anomalies Characterization and Traffic Data Mining



Graphical presentation on CERNET

- sharp increase in link utilization when MS-SQL Slammer worm broke out at 13:30 p.m. (CST) on Jan. 25, 2003



Conclusions and future work

- Linuxflow has been designed and implemented
- Linuxflow's capability of handling gigabit network backbone not only proven by special tests, but also by the fact that it has been used on CERNET backbone successfully
- Cluster/grid computing techniques will be used to make it more scalable and powerful to handle OC48/192 traffic
- Further research will be focused on applications based on Linuxflow

Thanks!