SNMP-based QoS Programming Interface MIB for Routers
(draft-kanada-diffserv-qospifmib-00.txt)

“Programming view” (active network view) is important for QoS control

- **The active networks**: Customizable, Downloading programs to nodes.
- **Policy-based QoS control**: Customizing resources, Downloading policies (rule-based programs) to nodes — the first step towards AN.
  - We need programs for configuration, because the function is so complex.
  - Program semantics must be specified formally to be interoperable.

**Design of the QoS MIB**

- A step toward establishing programming methodology for active networks.
- Features of the QoS MIB
  - **Rule Modularity** — very important for programming.
    - Each classifier, meter, or action is an if-then rule — separated by virtual flow labels.
    - The order of rules is not significant — the conditions are disjoint (no if-then-else).
  - **Classification of scheduling algorithms**: 1) First-in first-out (FIFO) scheduling, 2) Priority scheduling, 3) Packet-fair scheduling, 4) Byte-fair scheduling, 5) Bounded byte-fair scheduling
  - **Classification of dropping algorithms**: 1) Dropping all, 2) Tail dropping (non-early dropping), 3) Random early dropping (RED/WRED), 4) Deterministic early dropping (DED/WDED)

**Problems and solutions in current programming interfaces (PIFs)**

- Typical PIFs for QoS control: 1) SNMP & MIB, 2) COPS & PIB, 3) API
- Problems in PIFs by MIB (& PIB): Unsuitable for representing program semantics
  - Mismatching of syntactic structures and of unit or operation
- Possible solutions
  - To design a rule-based programming language for the interface.
  - This language may be implemented using any protocol.
    - The language semantics must be mapped to the protocol usage formally.
  - Or, the definition of a protocol must embed a language definition.
    - A new method of specifying protocols is required.