Patterns in SNMP-Based Network Management

Paul E. Sevinç
Swiss Federal Institute of Technology Zurich (ETH Zurich)

Jean-Philippe Martin-Flatin
University of Quebec in Montreal (UQAM), Canada

Rachid Guerraoui
Swiss Federal Institute of Technology Lausanne (EPFL)

ICSSEA 2004
Paris, France
1 December 2004

Outline

- Management application design: from niche to main-stream software engineering
- Architectural and design patterns in SNMP
- Research perspectives

Management Application Design: from Niche to Main-Stream Software Engineering

Problem Statement (1/3)

- Management application market has thrived on SNMP throughout the 1990s:
  - simple to use
  - small investments, guaranteed short-term ROI
  - now supported by most network devices in the world

Problem Statement (2/3)

- SNMP has exhibited major shortcomings over time:
  - only good at micro-managing network devices
  - inappropriate for managing services, e2e networks, etc.
  - does not scale:
    - → no standard way of organizing managers in a hierarchical or cooperative manner
  - requires many domain-specific skills:
    - → newcomers are not interested: they prefer to acquire skills that can be leveraged in many domains (e.g., HTTP, XML, WS)
    - → shortage of top-notch management application designers
  - some major design flaws:
    - → info model and comm model are tightly coupled
    - → data-oriented info model
    - → no easy way of automating configuration updates
  - unable to evolve in a timely manner (IETF WGs)

Problem Statement (3/3)

- The industry now looks for alternatives:
  - DMTF, TMF, etc.
  - from data-oriented to object-oriented info models (UML)
  - from domain-specific data-transfer protocol (SNMP) to domain-indep protocol (e.g., HTTP)
  - from domain-specific ways of representing data (BER-encoded SMI OIDs) to domain-indep ways (e.g., XML)
  - from specific to standard data compression
  - from proprietary to standard distribution of managers (hierarchy, P2P, MAS)
  - from domain-specific ways of exchanging data beween agents and managers to domain-indep middleware (e.g., WS, CORBA)
  - from micro-management to SOA and SOC

Our General Approach to this Problem

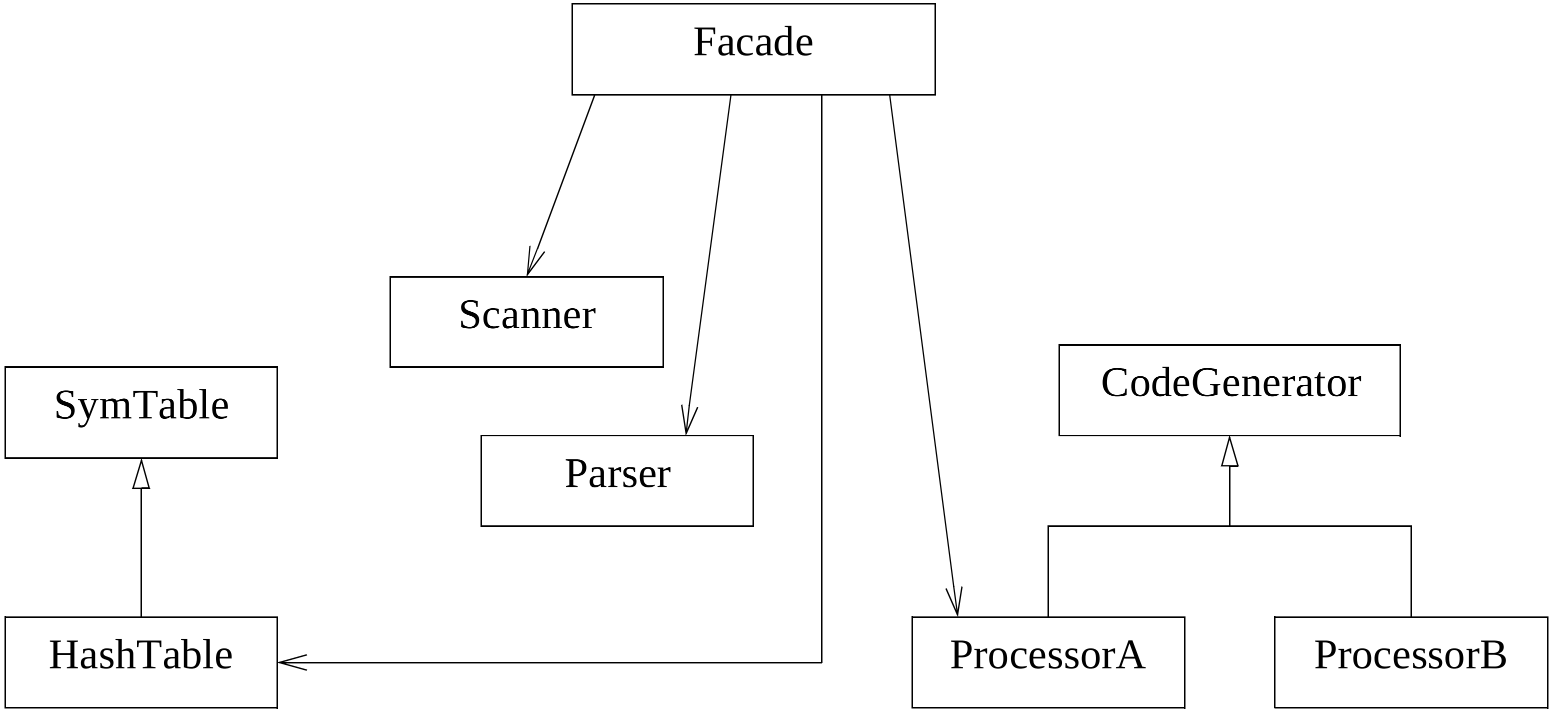
- A management application is yet another type of distributed application. To design it, we should avoid domain-specific solutions and use standard tools and techniques from:
  - software engineering
  - distributed systems
- By leveraging software architecture and patterns, we can:
  - focus on what is specific to management
  - define links between device, systems, e2e network and service management
  - focus on functional aspects
  - make the design of management applications somewhat independent of non-functional aspects
  - stop running after constant changes in middleware, prog languages, data representation, comm protocols, etc.

This Paper: Patterns

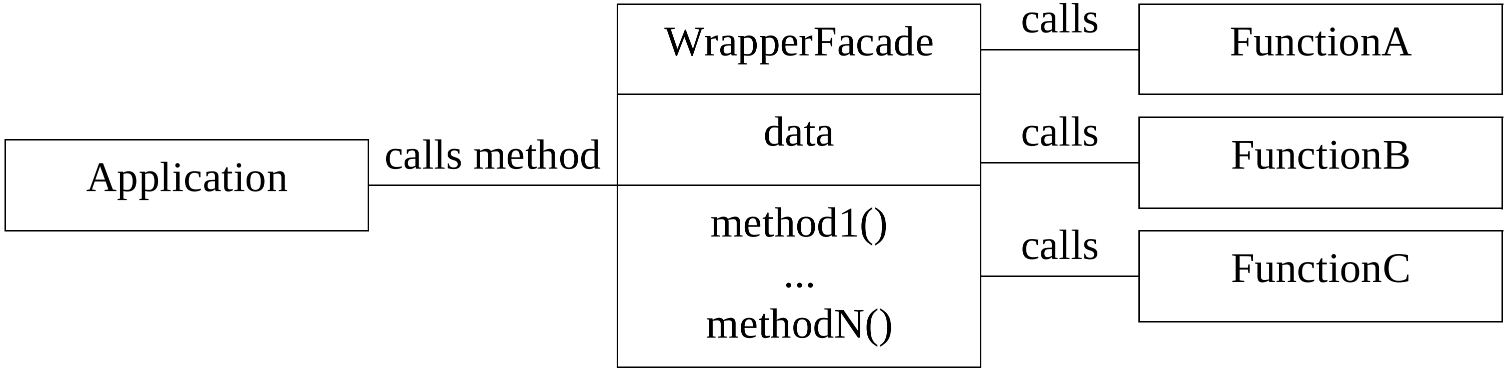
- Management information modeling:
  - the community is going from data-oriented models (SNMP MIBs) to object-oriented UML models (CIM models)
  - different standards bodies (IETF -> DMTF), different people
  - risk to lose know-how painfully accumulated over the years by the SNMP community
- Workflow and business processes:
  - from nothing to workflow and BP models
- Patterns make it possible to document how things are done (good and bad) and what lessons were learned (best practices):
  - shorter learning curve for new engineers
  - lingua franca to discuss design solutions
  - hopefully better design of future management apps

Architectural and Design Patterns in SNMP

Facade Pattern



Wrapper Facade Pattern

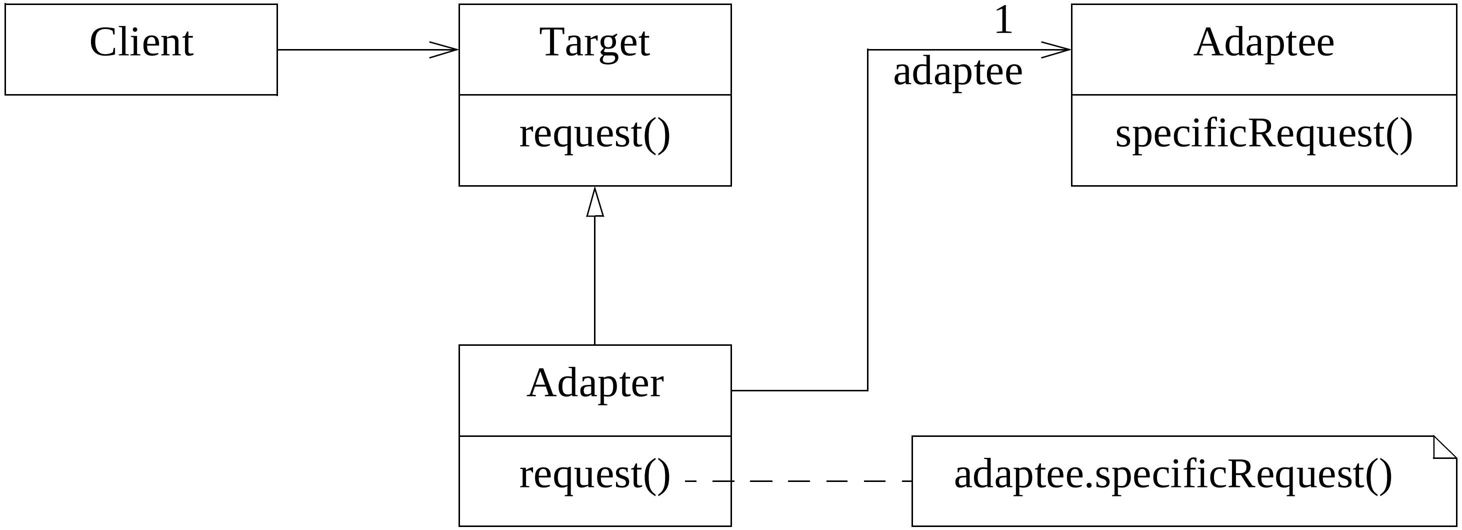


Wrapper Facade in SNMP

- *Occurrence:* Interface between an OO manager and a procedural application layer:
    - Do not invoke C functions directly from Java class via JNI, but via a wrapper facade

Object Adapter Pattern

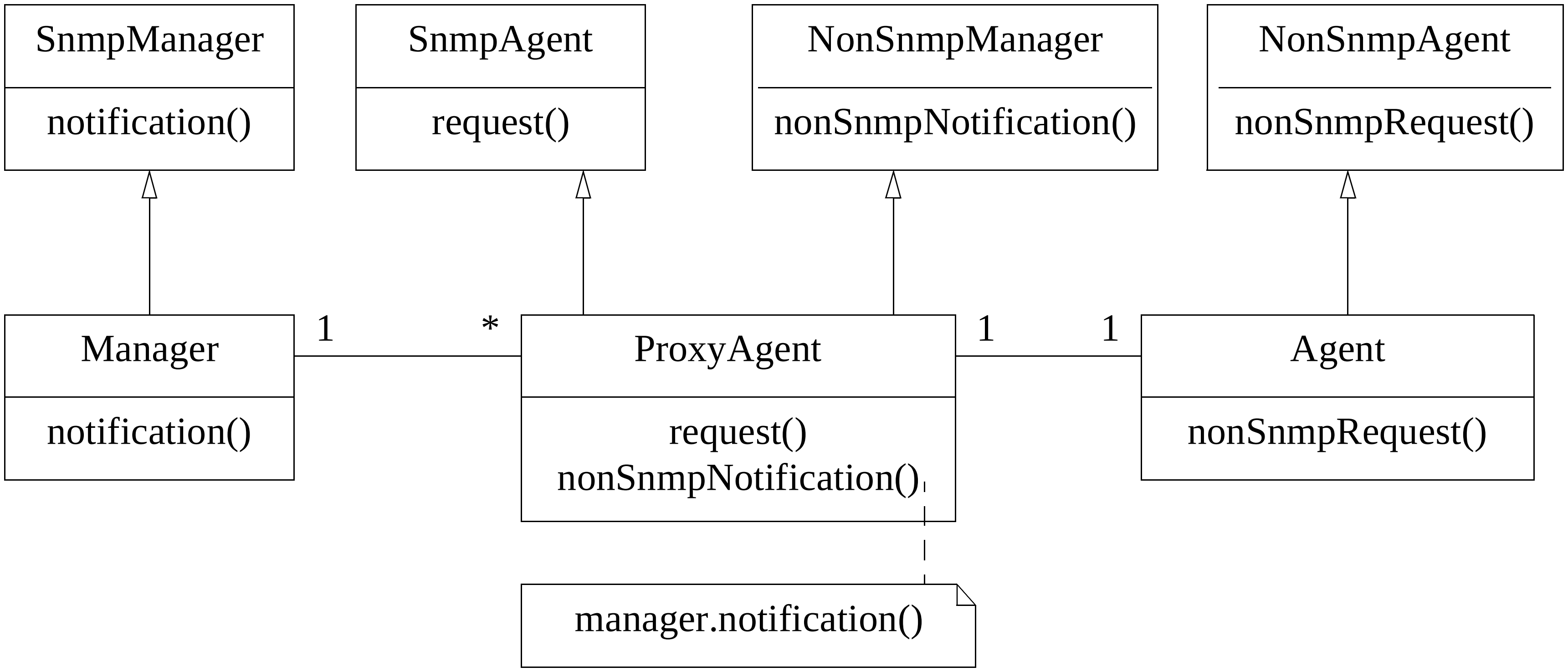
- Variant of the Adapter pattern based on object composition



Object Adapter in SNMP (1/2)

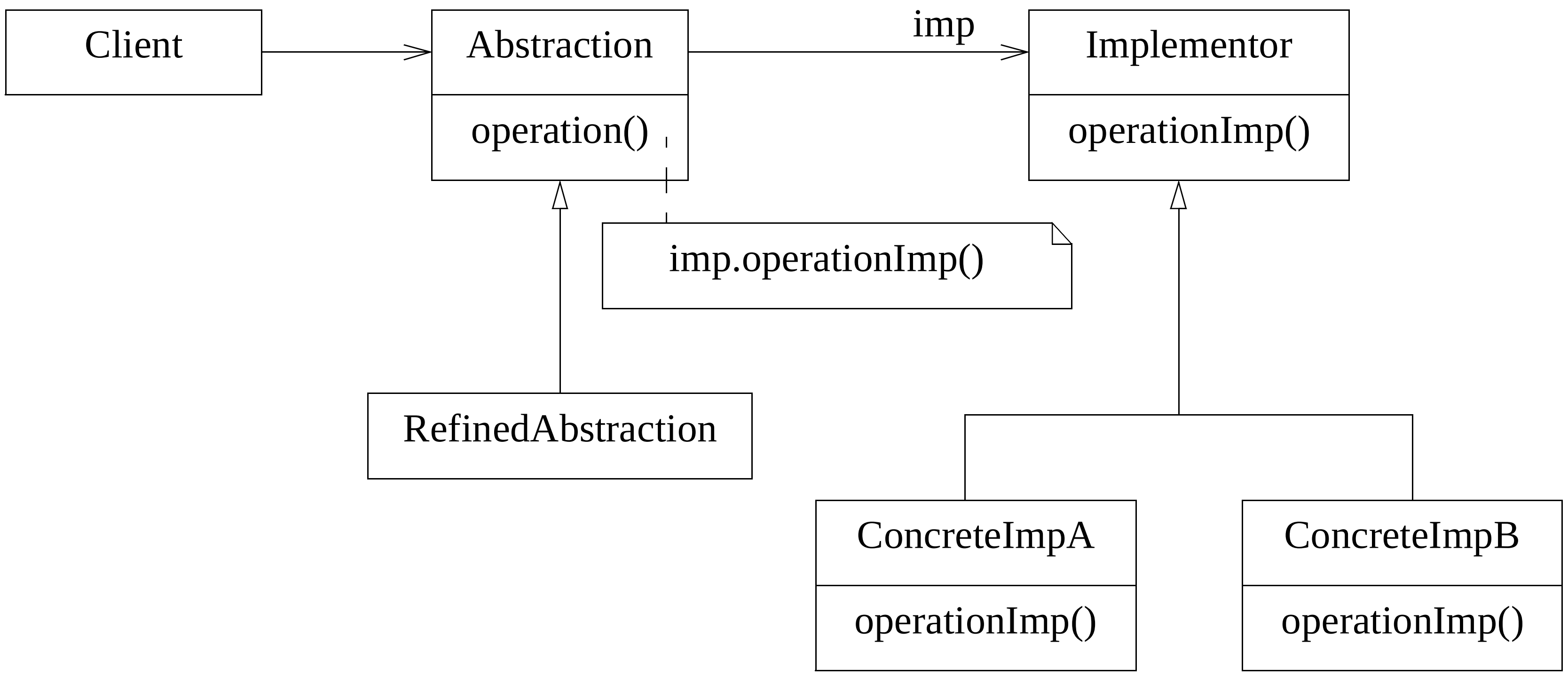
- *Occurrence:* proxy agents:
  - e.g., the managed entity does not run an SNMP agent
  - e.g., the managed entity runs several agents, and we need to access a virtual data repository (e.g., an OSI MIT) that is not available via SNMP
  - The manager plays the role of the Client
  - The managed entity plays the role of the Adaptee

Object Adapter in SNMP (2/2)



- In the case of notifications, the proxy agent can play the role of a two-way adapter

Bridge Pattern

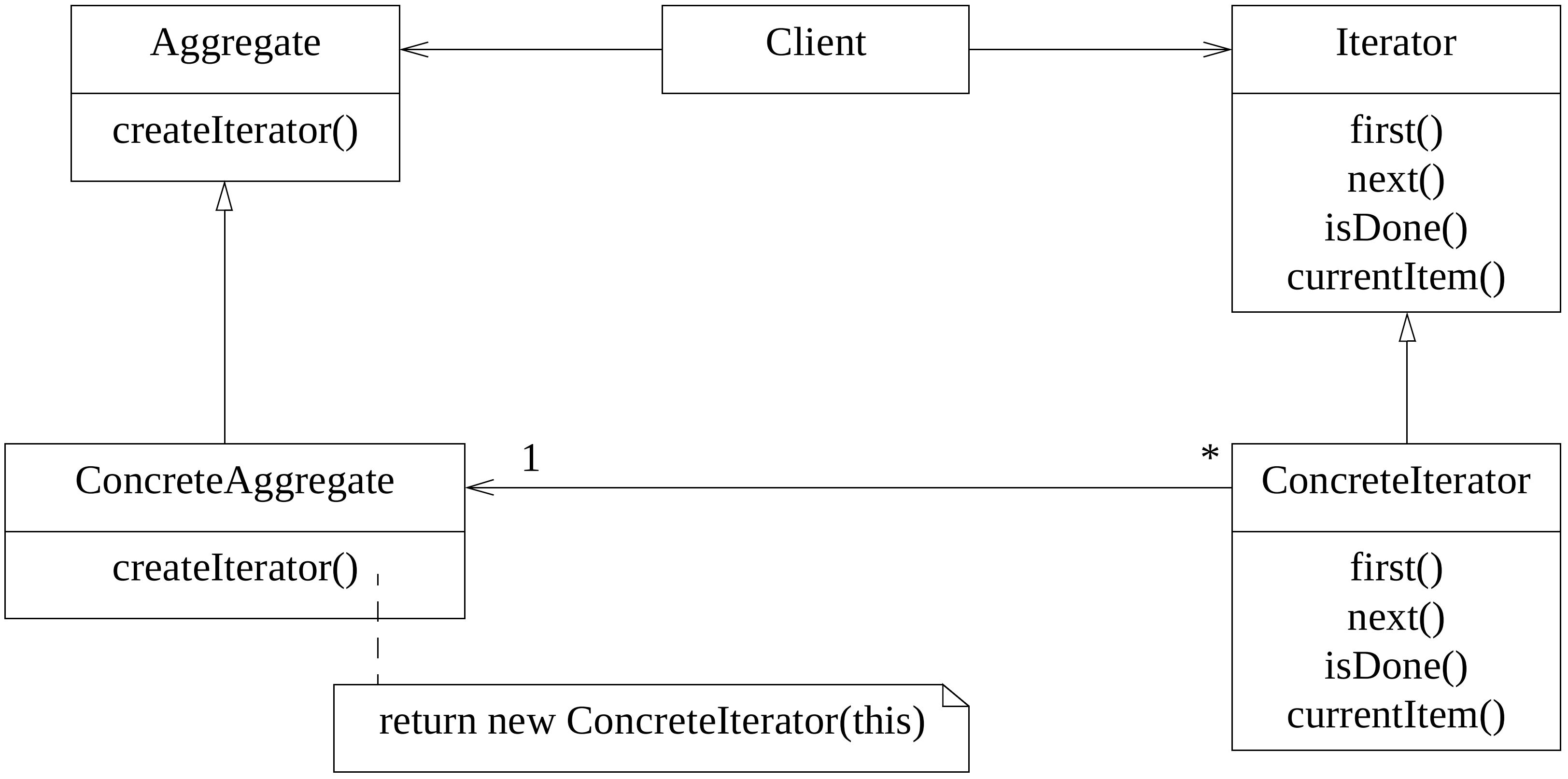


- Number of classes that have to be designed: number_of_refinements + number_of_implementations

Bridge in SNMP

- *Occurrence 1:* Can use different brands of relational databases (with vendor-specific SQL optimizations) for storing monitoring data, network map description data, etc.
- *Occurrence 2:* Can use different types of storage (e.g., RDBMS, LDAP directory, flat file) for archiving events (e.g., incoming notifications, events generated during on-the-fly data analysis)
- *Occurrence 3:* Encryption and compression schemes in SNMPv3:
    - the management application uses them transparently
    - the abstractions are completely decoupled from their implementations

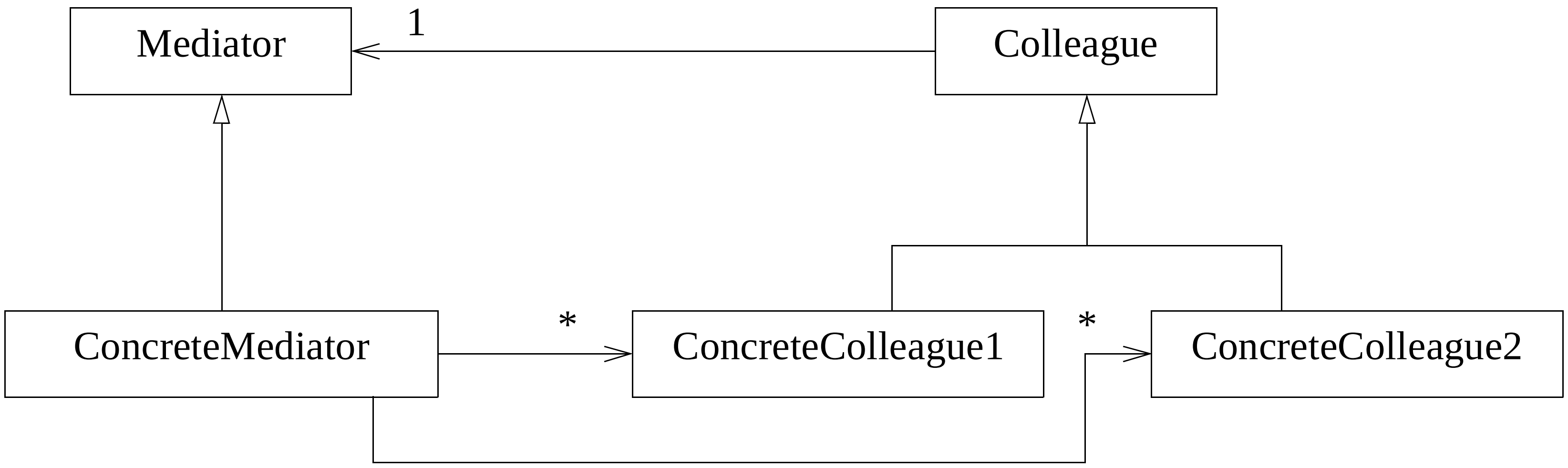
Iterator Pattern



Iterator in SNMP

- *Occurrence:* Retrieve an entire MIB subtree using `get-next` or `get-bulk` operations. If we place the iterator on the manager side, the manager need not know anything about the structure of the MIB on the agent side.

Mediator Pattern



Mediator in SNMP

- *Occurrence:* In the network map GUI:
  - When the state of a router changes to “down” and its icon changes color, the map Mediator should change:
    - → the states of all the nodes that are no longer reachable behind this router (to “undetermined”)
    - → the colors of the pertaining icons
  - The router icon does not know which of the other icons should change color

Research Perspectives

Directions for Future Work

- Still much work to be done to capture all current management practices in the form of patterns:
  - probably need to create a few new patterns
- Compare patterns in SNMP-based and CIM-based management
- Long-term objective: Provide a catalog of patterns for management application designers to choose from