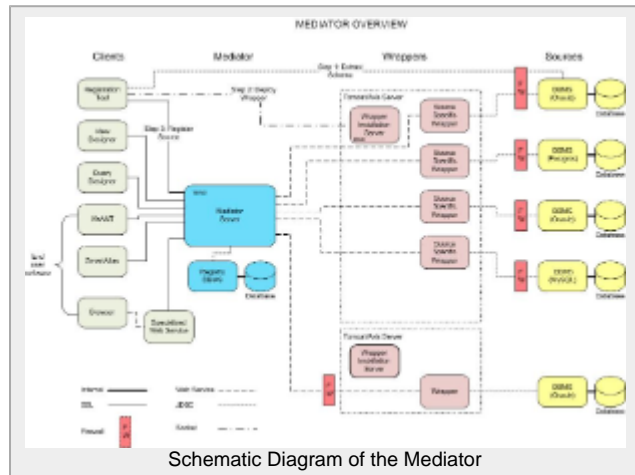


Mediator

The Mediator represents an approach to data integration in which the data remains in place--typically in relational databases which are referred to as "sources". This Mediator approach is a contrast to other means of data integration such as data warehousing in which data is gathered from various sources and stored in a single database or possibly a cluster of databases under central management.

The Mediator described here is a server application developed at the University of California, San Diego's Center for Research in Biological Systems (crbs.ucsd.edu) under a grant from NIH and developed to support several "test bed" projects with the aim of facilitating the sharing of biomedical research data generated and stored within their own institutions.

The Mediator maintains a repository of the schema of source databases and with that information can "mediate" queries across multiple heterogeneous sources. Again, the Mediator does not warehouse the data--only the information about the data.



Overview Diagram

The Mediator Overview diagram below shows how the Mediator works with client applications and sources to mediate queries and integrate multi-source data.

- The four color-coded vertical columns show the principal components.
- The Mediator column (second from the left) shows the Mediator Server with its Registry which is maintained in a dedicated database.
- Client applications (leftmost column) submit requests to the Mediator via a secure SSL connection.
- The Mediator "talks" to sources (rightmost column) via source-specific "Wrappers" (third column from the left)
- Typical institutional firewall options are shown as well.
- In response to client requests, the Mediator will in turn make read-only requests of the sources, retrieve data, integrate it (described below) and return a response to the client application.

Mediator Characteristics

Regarding the Mediator as a server application these are some of its characteristics.

- The Mediator is middle-ware.
- It does not have a user interface.
- Requests to and responses from the Mediator are streamed text XML documents.
- The client-server protocol is blocking request-response.
- Multiple simultaneous clients are supported.
- The Mediator is written in Java.
- Linux is the platform of choice but other Java-capable platforms should work as well.
- The Mediator listens and responds via an SSL port.
- Communication to and from wrappers is via web services.
- The Mediator uses a dedicated database for its Registry. Both MySQL and Oracle have been used, but any JDBC accessible database will suffice.
- Currently supported source databases are MySQL, PostgreSQL, Oracle, SQLServer, and Sybase, all of which can be integrated via the Mediator.
- The Mediator does not write to sources and can fully function with read-only access.
- Logging supports activity monitoring, metrics, and trouble shooting.
- In many client requests, the Mediator can retrieve partial integrated data even though some sources may be down.

Integrated Views

This is a key Mediator data-integration concept.

- A view that can span sources.
- Analogous to but opposed to a local view within a database.
- Can be created by the ViewDesigner client application.
- Stored in the Mediator's registry.
- Can be used as a component of a more complex query or another integrated view.

Mediator Internals

- The principal components are
 - Gateway - interface to clients.
 - Registry - an internal database and related application logic.
 - Planner - develops a plan for obtaining and integrating data across sources.
 - Executor - executes the plan, manages source-specific queries, and integrates data.

Registry Details

- An internal Mediator database.
- Stores metadata about sources' database schema.
- Does not contain actual data.
- Used by the Mediator to plan and execute queries.
- Provides information about sources to Mediator clients.
- Does store sources' user names or passwords--these are held in the wrappers.

The Registration Tool

- A specialized Mediator client.
- A Java GUI application.
- Can be installed on any Java-capable platform.
- Can be offered as Java Web Start application accessed via a browser.
- Reads source database schema.
- Stores database schema in the Registry.
- Generates and deploys wrappers.
- Registers a source with the Mediator.
- See the diagram and Steps 1, 2, and 3 for the Registration Tool
- Schema can be selectively exposed to the Mediator. In other words, some tables and columns can be registered, while others may be made invisible to the Mediator and thus as well to Mediator clients.

Wrappers

- A Java middle-ware application between the Mediator and a source DBMS.
- Adapts a request from the Mediator to a specific DBMS.
- Maps Mediator objects and datatypes to the source DBMS.
- Operates as a web service to the Mediator.
- Connects to a source's DBMS via JDBC.
- Buffers return data between the DBMS and the Mediator.

Wrapper Installation Server

- A Java server application used to deploy wrappers on a Tomcat/Axis server.
- Accepts requests from the RegistrationTool via a socket.
- Writes appropriate wrapper files to the Axis server.

Other Mediator client applications

QueryDesigner

- A Java GUI application.
- Designs and submits queries to the Mediator.
- Displays results

ViewDesigner

- A Java GUI application.
- Designs and registers integrated views with the Mediator.

Contact & Information

David Little, Vadim Astakhov
Center for Research in Biological Systems
University of California, San Diego
drlittle@ucsd.edu; astakhov@ncmir.ucsd.edu
858 822-0742; 858 729-8684