

# RTView Data Server

## Data Server - Benefits

By default, RTView clients connect to data sources (SQL, TIBCO, JMX, etc) directly. However, data from any source can be redirected through the Data Server.

Reasons for implementing the Data Server

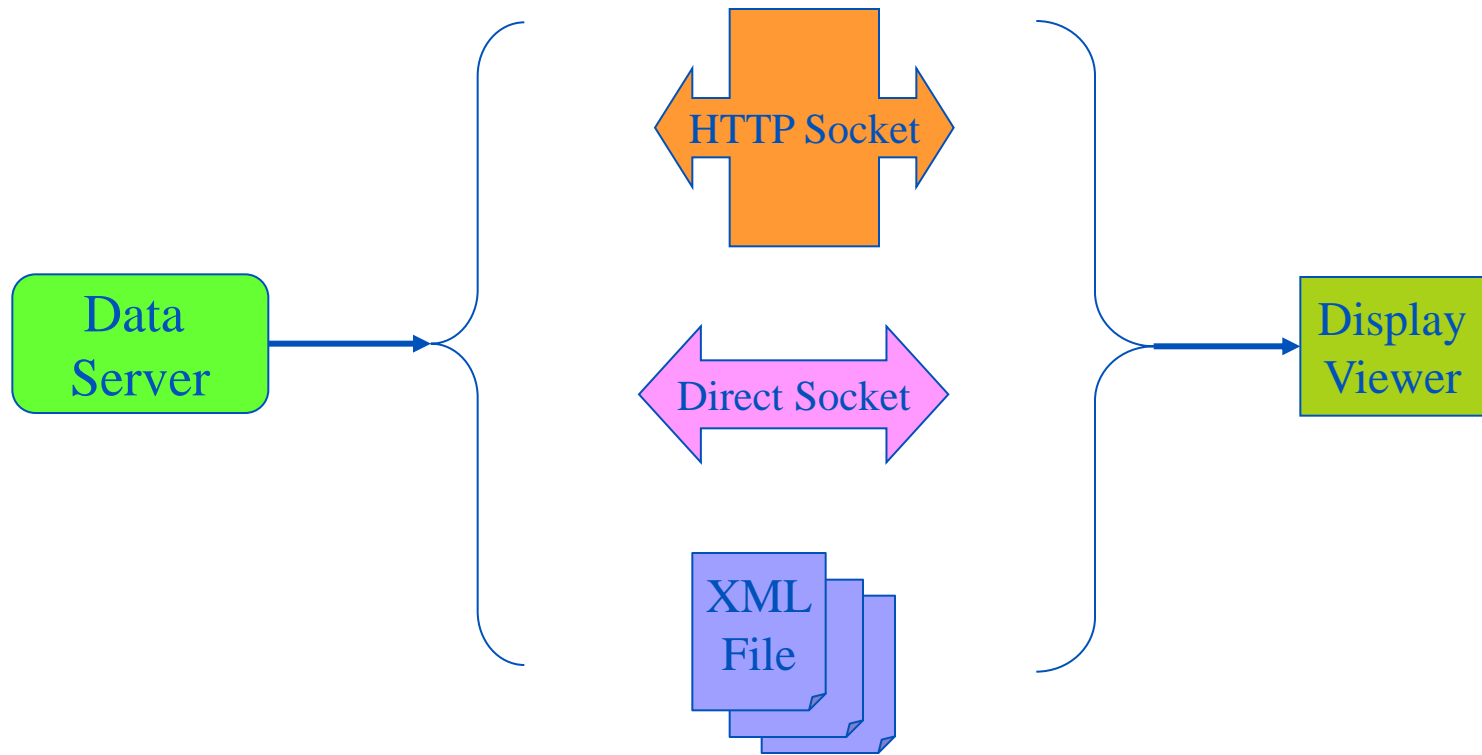
- **Data Access:** In some cases the data source resides in a sub-network where it is not possible or desirable to connect directly. For example, you may have a data source that communicates via a TCP/IP port that is not exposed across a security firewall. In this case, the Data Server would exist behind the firewall and connect to the chosen display technology via HTTP or an available exposed port.
- **Data Centralization:** While a direct TCP/IP connection to the data sources may be possible, it may not be desirable for each data source and each client. For example, with a Display Viewer Application deployment you might not want a direct TCP connection from each client to every data source (e.g., database). In this case, the Data Server could act as a proxy by directly connecting to all data sources and providing data to each Java application client.

## Data Server - Benefits

Reasons (continued)

- **Data Reduction/Aggregation:** There may be a lot of raw data that does not need to be sent except upon demand. Using the Data Server can reduce network traffic by performing data calculations from within a sub-network and providing either only the aggregated data or raw data on demand to the chosen display technology.
- **Scalability:** The Data Server has several data processing features including: an in memory data cache, a persistent alert rules engine, and the ability to perform data calculations. This processing load can be scaled out by dividing the labor across multiple Data Servers that will connect to the chosen display technology.
- **High Availability:** It is possible to designate one or more backup Data Servers to take over during a failover event when a RTView client discovers that the current Data Server is no longer accessible. Backup servers may be run in hot standby mode (default) or warm standby mode (designated as a command line option).

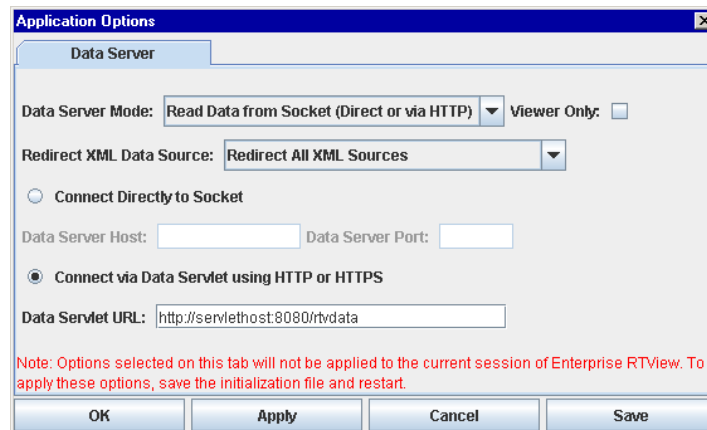
## Data Server Connectivity



## Data Server – Socket Connectivity

### Socket Connection

- Most efficient form of connection
- RTView clients send data requests informing the Data Server exactly what data is needed for current displays, so no configuration files are necessary.
- Serialized data objects are output over a socket to update displays.
- Clients requesting the same data will read from queue (1 request on back-end database will service several client requests).
- If data has not changed between requests, no data is pushed out to the clients (reduces network load)

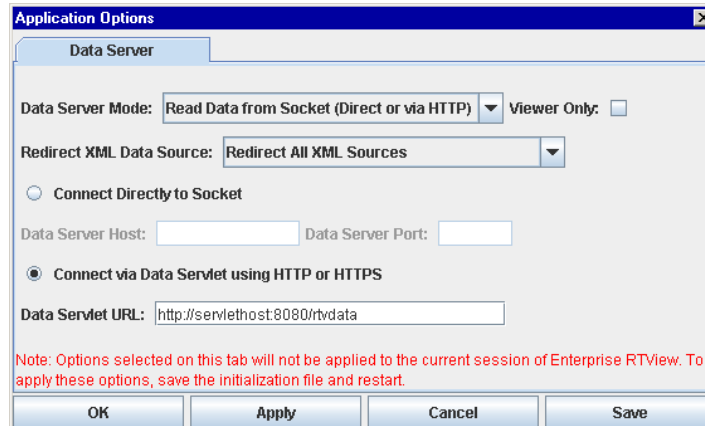


The screenshot shows the 'Application Options' dialog box with the 'Data Server' tab selected. The 'Data Server Mode' is set to 'Read Data from Socket (Direct or via HTTP)'. The 'Viewer Only' checkbox is unchecked. The 'Redirect XML Data Source' is set to 'Redirect All XML Sources'. The 'Connect Directly to Socket' radio button is unselected, while the 'Connect via Data Servlet using HTTP or HTTPS' radio button is selected. The 'Data Server Host' and 'Data Server Port' fields are empty. The 'Data Servlet URL' field contains 'http://servlethost:8080/rtvdata'. A note at the bottom states: 'Note: Options selected on this tab will not be applied to the current session of Enterprise RTView. To apply these options, save the initialization file and restart.' The dialog has 'OK', 'Apply', 'Cancel', and 'Save' buttons.

## Data Server – Socket/http Connectivity

### HTTP Connection

- Most complex form of connection
- Most secure form of connection
- Data Server and data sources reside behind a firewall.
- Dataservlet acts as intermediary between client and dataserver
- Only HTTP port is opened on firewall
- Behaves the same as socket connection



The screenshot shows the 'Application Options' dialog box with the 'Data Server' tab selected. The 'Data Server Mode' is set to 'Read Data from Socket (Direct or via HTTP)'. The 'Viewer Only' checkbox is unchecked. The 'Redirect XML Data Source' is set to 'Redirect All XML Sources'. The 'Connect Directly to Socket' radio button is unselected, and the 'Connect via Data Servlet using HTTP or HTTPS' radio button is selected. The 'Data Server Host' and 'Data Server Port' fields are empty. The 'Data Servlet URL' is set to 'http://servlethost:8080/rtvdata'. A note at the bottom states: 'Note: Options selected on this tab will not be applied to the current session of Enterprise RTView. To apply these options, save the initialization file and restart.' The dialog has 'OK', 'Apply', 'Cancel', and 'Save' buttons.

Application Options

Data Server

Data Server Mode: Read Data from Socket (Direct or via HTTP) Viewer Only:

Redirect XML Data Source: Redirect All XML Sources

Connect Directly to Socket

Data Server Host:  Data Server Port:

Connect via Data Servlet using HTTP or HTTPS

Data Servlet URL:

Note: Options selected on this tab will not be applied to the current session of Enterprise RTView. To apply these options, save the initialization file and restart.

OK Apply Cancel Save

## Data Server – Data Servlet Deployment

- Installing the Servlet
  1. %RTV\_HOME%/servlets/rtvdata directory contains files (JSP, HTML, classes, properties) necessary to run the JSP servlet.
  2. Build war using “make\_war.bat”
  3. Deploy war file to app server.
  4. Use “install\_to\_tomcat.bat” if using Tomcat. CATALINA\_HOME points to Tomcat install directory
  
- The JSP servlet reads the file servlet.properties.
  - ServiceHost=localhost
  - ServicePort=3278
  - ServiceTimeout=15

## Data Server Deployment – Socket Mode

### Server side (Data Server - socket)

- \*OPTIONS.ini (for all datasources)
- RVALIAS.ini (optional)
- JMSALIAS.ini (optional)
- DATASERVER.ini
- TRANSPORTS.ini (optional)

### Web Server

- index.html
- gmsjrtview.jar, gmsjmodels.jar, iText.jar, gmsjrtvreport.jar, J2PrinterWorks.jar, activation.jar, mail.jar
- myclasses.jar (optional) – contains jpg, custom classes
- OPTIONS.ini
- \*.rtv
- PANELS.ini (optional)



## Data Server - Setup

### Server configuration

- Configuration tab of Data Server
- Select type of connectivity
  - File mode - requires selection of rtv files to scan
  - Socket mode - requires selection of a port number
  - Http mode - Same as socket mode + installation of servlet
- Save options before exiting
- Changes require a restart of Data Server
- Data Server settings for server are saved in DATASERVER.ini

## Data Server - Setup

### Client configuration

- Data Server tab of File->Options
- Select type of connectivity
  - File mode - enter name of file to use (rtvdata.xml)
  - Socket mode - enter host and port number for Data Server
  - Http mode - enter URL for the XML Servlet
- Save options before exiting
- Changes require a restart
- Data Server settings for client are saved in OPTIONS.ini

## Data Server

- Running the Data Server with a GUI
  - `run_dataserver.bat`
  
- Running the Data Server as daemon process
  - `run_dataserver -daemon` (ideal for deployment on headless environments)
  - Allows running as a Unix server daemon process
  - Allows running Data Server as a windows service
  
- Debugging flags for data sources are applicable to the data server

```
run_dataserver -jmxtrace:3 -sqltime
```

## Data Server

- Monitoring Data Server connections
  - Run the Data Server with GUI
  - Click on „Connections’ tab
    - Number of clients will be displayed
    - Number of bytes served to each client is displayed
  
- Data Server flags for fail-over
  - -dbretry:(milliseconds)
    - On startup, xmlserver will retry connection until it succeeds
  - -dbfailedlimit:(number of failed queries)
    - After n failed SQL queries, a db connection will be re-established

## Exercises

### Ex 1: Create a script to run the Display Server with Data Server

1. On windows create rundata.bat file using notepad. Add this line:

```
run_dataserver -daemon -verbose2 -port:3278 -socket %*
```

2. Create rundisp.bat using notepad. Add this line:

```
run_displayserver -daemon -verbose -dataserver:remote://localhost:3278 %*
```

*Note: cutting and pasting lines directly from this page usually fails, since control chars are added.*

3. run the scripts:

```
start rundata.bat  
start rundisp.bat
```

## Exercises

### Ex 1: Create a script to run the Display Server with Data Server (cont)

4. In the browser, type in the tomcat URL to confirm tomcat is running

`http://localhost:8068`

If the main tomcat page appears, then tomcat is running.

5. Type

<http://localhost:8068/rtvdisplay>

or

`http://localhost:8068/rtvdisplay/panels.jsp`