codestreet

ReplayService™ for IBM Websphere MQ

Record and Replay MQ messages

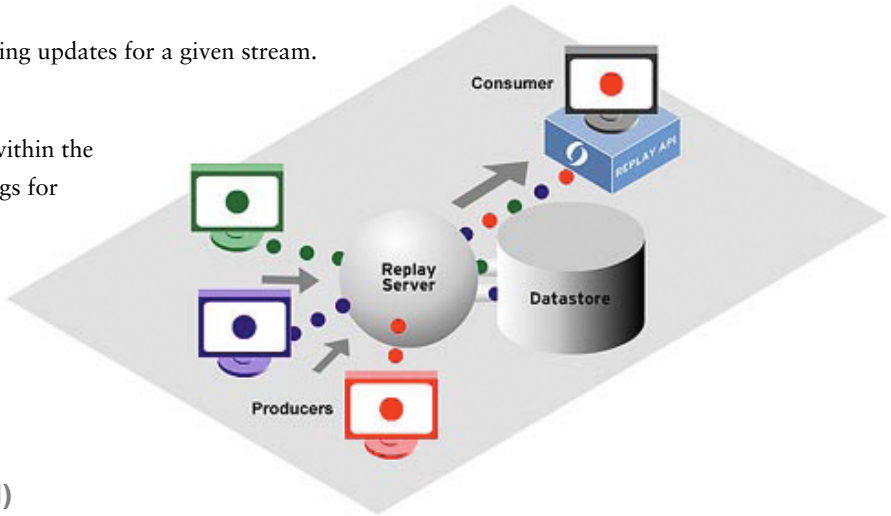
The CodeStreet ReplayService is like a TiVo/DVR for your IBM Websphere MQ message platform. Now you can record, and repeatedly replay, pause, fast forward and rewind through the flow of messages. You can also vary the speed of the message stream and filter selected message subsets. Messages can also be searched, edited and re-inserted into the message stream. ReplayService is not just a record and replay engine; It's a better store-and-forward messaging system with a broad range of business applications.

Products

CodeStreet ReplayService for MQ records and replays messages from the WebSphere family of message oriented middleware from IBM. The MQ version of ReplayService is available for Windows, Linux, and Solaris.

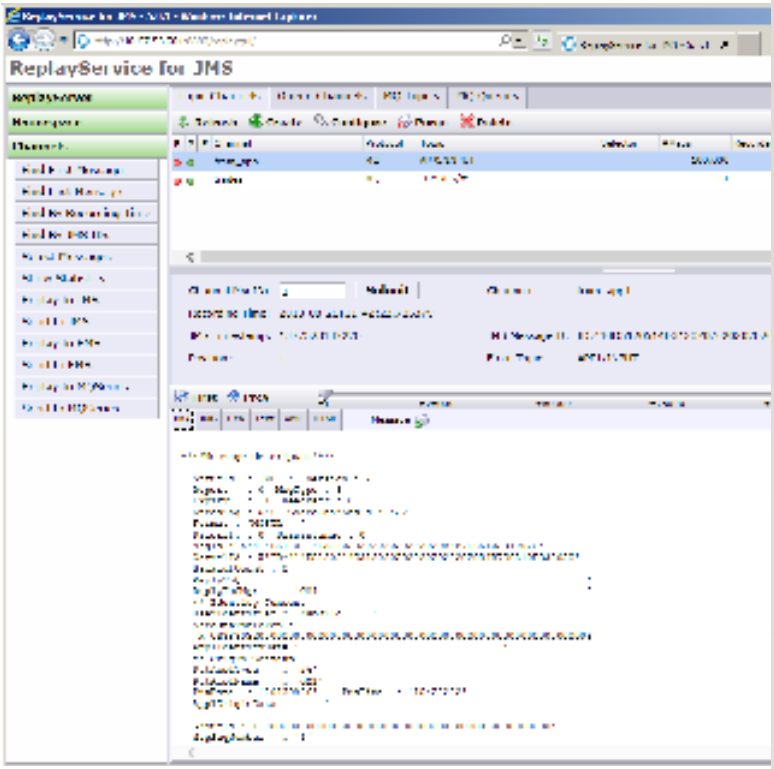
Benefits

- **Simpler Store-and-Forward Delivery**
  Message senders and consumers can operate at their own message rates without pre-registration of consumers.
- **Application State Recovery**
  Simplifies building fault-tolerance and state recovery into applications.
- **Data Distribution**
  Applications throughout the enterprise automatically receive data updates, so there is no need to centralize data access or struggle with complex replication schemes.
- **Last Value Cache**
  Provides both last value snapshots and ongoing updates for a given stream.
- **Auditing of Business Activity**
  ReplayService records information moving within the enterprise, so it's easy to maintain activity logs for business and regulatory purposes.
- **Functional and Performance Testing of Applications**
  Application performance can be tested by varying the replay rate and by simulating a live stream.
- **Business Activity Monitoring (BAM)**
  ReplayService gathers the data required for complex Business Activity Monitoring and can simply transfer the data to various databases.



ReplayService for IBM Websphere MQ

Features



- Simultaneously records multiple IBM Webshpere MQ topics and queues, including wildcards.
- Provides extensive flow control. Replays messages at a flat rate or as a function of the original message spacing.
- Both message consumers and administrative applications have the ability to suspend and resume replay streams, and to adjust the replay rate. The average replay rate is automatically throttled down when client applications consume messages too slowly.
- Applications transition seamlessly from retrieving stored messages to consuming real-time messages, making it unnecessary to synchronize replayed and real-time portions of the stream.
- Guarantees message order — messages from a given group of channels are always forwarded/replayed in the same sequence in which they were recorded.
- Easy to use per-channel message rate statistics.
- Extensive real-time monitoring and management of the ReplayServer and Replay API for client applications.
- ReplayService fault-tolerance and SSL support.
- The ReplayService API is an implementation of the JMS Version 1.1 specification, so using the service in an existing JMS application is as simple as modifying the syntax of the JMS message selector.
- Simple and easy to deploy and configure; no changes are required to the existing IBM Websphere MQ Messaging System architecture. Integrates seamlessly with your current MQ or JMS applications.
- Replayed messages can be selected by specifying a group of channels, topics or queues, and by additional criteria such as JMS message selectors and IBM Websphere MQ Message IDs.
- Works with or without the ReplayService API. The API is used when the message consuming application wants to control features such as pause and resume. When native APIs are used, ReplayService is controlled via an administration application.

For more information, please contact:

Codestreet Market Data Sales

sales@codestreet.com
646-442-2800

ReplayService for MQ is a valuable tool for IBM Websphere MQ users and applications developers

CodeStreet, LLC
200 Park Avenue, 17th Floor
New York, NY 10166
Telephone 646 442 2800
www.codestreet.com