ADVANCED MESSAGING DATABASE

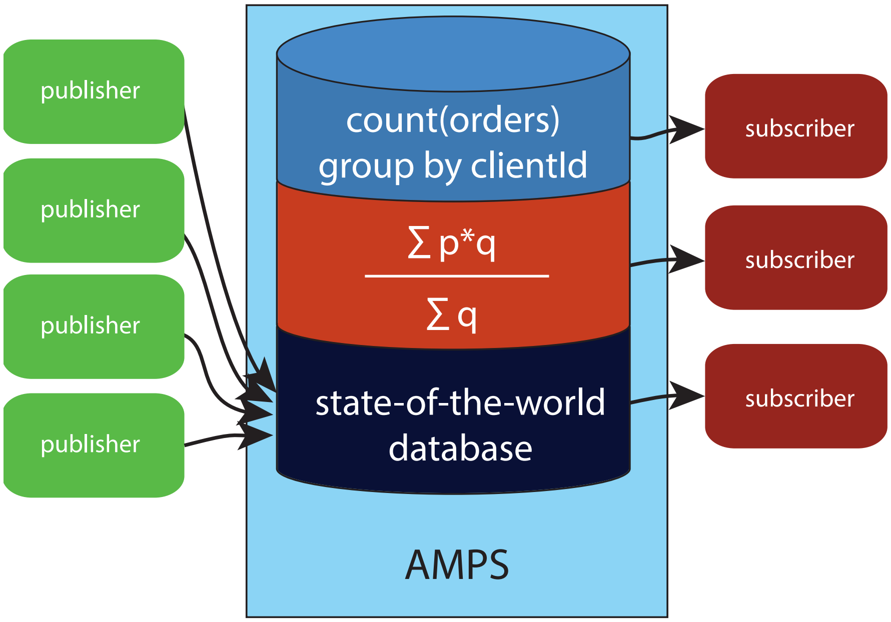
FEATURE

Cranking Up Aggregation

Get The Whole Picture

Some messaging systems act as though each message is a complete universe, with no relationship to other messages. Those systems require clients to assemble, correlate, and aggregate messages. Some systems even force clients to discard messages that the client can't use, which often means wasted bandwidth and processing power on dozens or hundreds of messages thrown away for each message processed.

At 60East, we know that each message is a part of the whole picture. We also know that you need both the whole picture and the individual message. AMPS lets you aggregate and do core analytics on messages published to a single topic, and like a relational database AMPS lets you join different topics to aggregate, analyze and enhance messages.



SOW: The Current Message Cache

The AMPS current message cache, known as the State-of-the-World or SOW, stores the current value of unique messages. As message values are updated, AMPS updates the messages in the SOW. Subscribers can query the SOW to retrieve the current state of the cache. This solves "late-subscriber" problems: whenever a client subscribes to the topic, that client can retrieve the full current state. In addition, an application that only needs to know the current value of a set of messages can query for current values, with no need to receive or process other messages.

Real-Time Streaming Joins and Aggregation

If you need aggregation for one topic—for example, a running total of order volume for a single item—that's easy to do with AMPS. You describe how to aggregate the messages based on the underlying topic. AMPS creates a new topic for the aggregation. The aggregation topic behaves like any other topic with a SOW. Subscribers do not need to know that the topic is an aggregation, or the underlying source of the messages. Each time a change is published that changes the value of the aggregation, AMPS sends a message to the aggregate topic. AMPS allows you to group messages for common analytics such as running totals and performing math such as multiplying one field by another, calculating averages, or similar tasks.

But that's not all. AMPS goes further, allowing you to aggregate multiple topics, in the same way that a relational database can create materialized views for a JOIN statement. In this case, AMPS handles the update logic for each message in the underlying topics. As with SOW and simple aggregation, subscribers handle updates using simple query and subscribe commands. As with simple aggregation, there is no need for a subscriber to be aware that the topic is the result of multiple topics.

Crank It Up

If you're ready to take your messaging to the next level, and if you need state-of-the-art aggregation, filtering, and point in time querying, then you're ready to CRANK UP THE AMPS. Contact us at info@cranguptheamps.com or visit our website to download a free evaluation of AMPS and get cranking!