

# Transactional Publish / Subscribe: The Proactive Multicast of Database Changes

Arvola Chan

TIBCO Software Inc.

3165 Porter Drive

Palo Alto, CA 94304

1-650-846-5046

arvola@tibco.com

## ABSTRACT

For many years, TIBCO (the Information Bus Company) has pioneered the use of Publish / Subscribe – a form of push technology – to build flexible, real-time, loosely-coupled distributed applications. Today, Publish / Subscribe is used by 300 of the world's largest financial institutions, deployed in 6 of the top 10 semiconductor manufacturer' factory floors, utilized in the implementation large-scale Internet services like Yahoo, Intuit, and ETrade, and chosen by many of the world's leading corporations as the enterprise infrastructure for integrating disparate applications. In this paper, we will:

- Contrast the Publish / Subscribe event-driven interaction paradigm against the traditional demand-driven request-reply interaction paradigm;
- Explain the concepts of subject-based addressing and self-describing messages, the cornerstones of Publish / Subscribe;
- Describe the scalable implementation of Publish / Subscribe via multicast and broadcast, and the proposed Pragmatic General Multicast Internet standard; and
- Categorize the qualities of service needed by different kinds of event-driven applications.

Today, TIBCO products support:

- *Reliable delivery* for front-office applications which require update notifications only while they are online;
- *Guaranteed delivery* for back-office applications that cannot afford to lose messages despite network and application failures; and
- *Transactionally guaranteed* delivery for those applications that must update databases, consume messages on one set of subjects, and publish messages on another set of subjects, all within properly bracketed atomic transactions.

The exactly-once delivery semantics for transactionally guaranteed delivery simplifies the implementation of back-office workflow applications by automating the common recovery logic.

Three different implementations of transactional Publish / Subscribe can be found in:

- A generic, database independent implementation embodied in TIBCO's Enterprise Transaction Express (ETX) product. ETX optimizes two-phase commit for those applications that span a single database and the messaging system by using the last resource manager optimization. It also supports more complicated transactions by playing the role of an XA-compliant resource manager, leaving the transaction coordination to standard-based transaction monitors.
- An Informix Universal Server specific extension package called the TIBCO Message Blade. This extends the SQL language with TIBCO-provided User Defined Routines (UDRs) for synchronous Publish / Subscribe operations. In general, UDRs can be used inside stored procedures and triggers to publish and consume (potentially complex) structured messages. The need for two-phase commit is finessed by storing messages in the same database that houses application tables.
- A bidirectional bridge between Oracle 8's Advanced Queuing (AQ) facility and TIBCO's TIB/Rendezvous guaranteed message delivery implementation. Oracle AQ supports enqueue and dequeue operations to queues (actually implemented as Oracle tables) that can be performed as part of database transactions. The bridge dequeues from Oracle queues and republishes on the Information Bus. Conversely, the bridge subscribes to TIB/Rendezvous messages and enqueues them to Oracle queues for consumption by Oracle applications. Multiple bridges can be used to route AQ messages from one Oracle database to another.

## Keywords

Publish / subscribe, event-driven, subject-based addressing, multicast, reliable delivery, guaranteed delivery, transactionally guaranteed delivery.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.  
SIGMOD '98 Seattle, WA, USA  
© 1998 ACM 0-89791-995-5/98/006...\$5.00