

Publish/Subscribe for High-Performance Computing Runtimes



Gregory M. Farmer
Florida A&M University
Research Alliance in Math and Science
Computer Science and Mathematics Division, Oak Ridge National Laboratory
Mentors: Dr. Richard Graham and Dr. Geoffroy Vallée



http://wiki.ornl.gov/sites/rams09/g_farmer

Introduction

- Scalable Tools and Communication Infrastructure (STCI)
 - Delivers developer tools to supercomputers
 - Provides a scalable runtime for tools such as debuggers and MPI implementations
- Publish/Subscribe
 - Asynchronous communication system
 - Used to propagate event notifications in distributed systems

Background

- High-performance computing
 - Message Passing Interface
 - Scalability
- Target high-performance computing platforms (e.g., Cray Jaguar, Cray Kraken)

Research Objectives

- Implement the storage sub-system using a database
- Implement the publish/subscribe local engine
- Integrate the components into STCI

Storage Sub-system

- Store key/value pairs
- Load key/value pairs based on queries (filters)

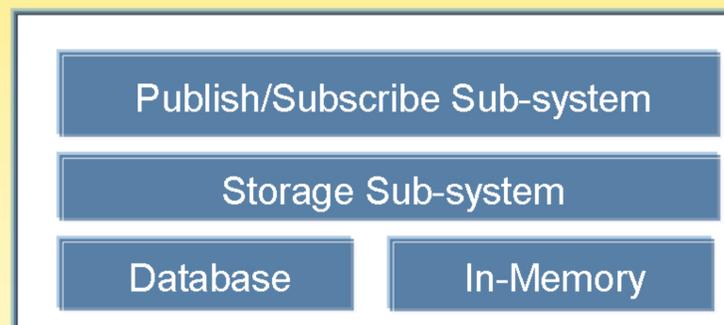


Figure 1. System architecture

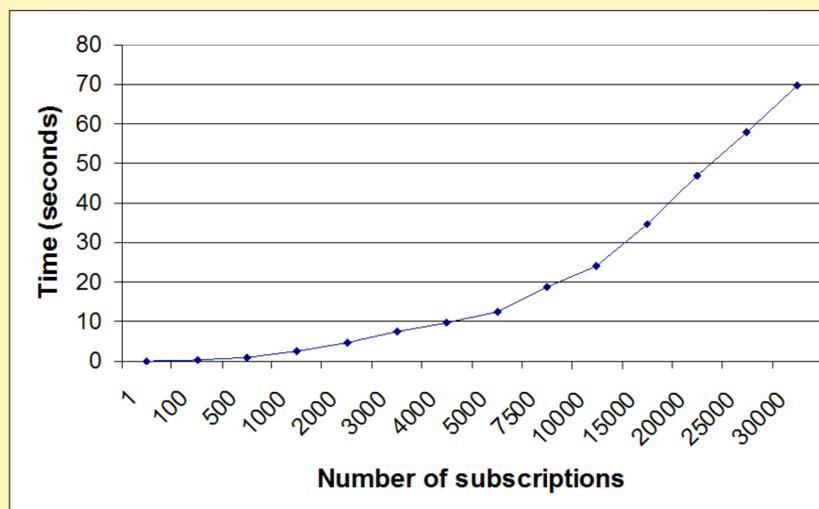


Figure 2. Subscription time

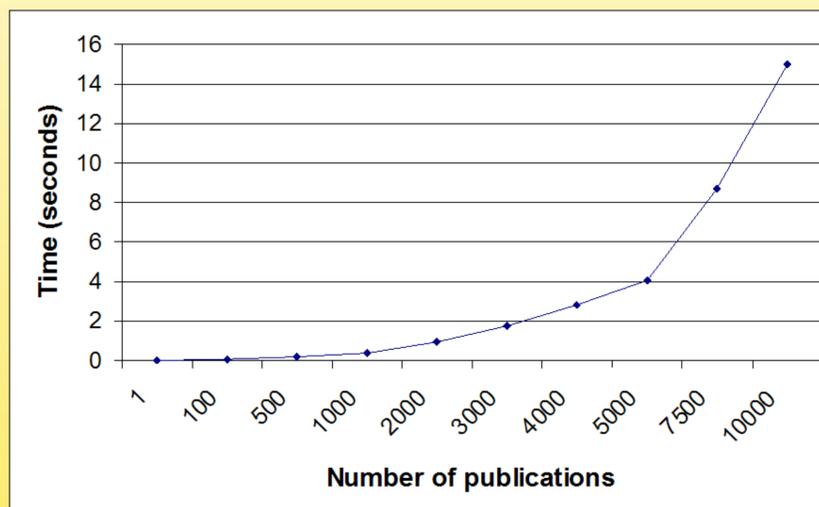


Figure 3. Publish time for one subscriber

Publish/Subscribe Sub-system

- Asynchronous messaging paradigm
- Anonymous delivery of messages
- Subscription to “topics”
- Messages published to topic / messages delivered to all subscribers

Methods

- Storage sub-system: implementation based on SQLite
- Implementation in C language
- Integrated into STCI and uploaded into the up-stream subversion repository

Results

- Experiments completed on the ORNL qcluster system (AMD dual core 2GHz – 1GB of memory)
- Two experiments
 - Subscription scalability (Figure 2)
 - Publication scalability (Figure 3)
- Consistent results (low standard deviation)
- Code integrated into STCI and available to STCI developers

Future Research

- Implement transport layer (distributed systems)
- Implement “in-memory” storage sub-system