Boosting Topic-Based Publish-Subscribe Systems with Dynamic Clustering

Tova Milo, Tal Zur, Elad Verbin
Presented by: Herodotos Herodotou
February 19, 2008

**Outline**
- Motivation
- Background on Pub-Sub Systems
- Topic Clusters
- Cost Model
- Dynamic Clustering Algorithm
- Subscription Algorithm
- Implementation
- Experimental Results

**Motivation**
- **Challenges:**
  - Dynamic Nature of P2P Pub-Sub Environment
  - Decentralized Nature of P2P Pub-Sub Environment
- **Goal:**
  - Maximize performance

**Background: Pub-Sub Systems**
- **Three Components:**
  - Subscribers
    - Express interest in events
  - Publishers
    - Create topics and generate events
  - Infrastructure
    - Match events with interests and deliver events

**Background: Pub-Sub Systems**
- **Categories:**
  - Topic-Based
    - Subscribers specify interests as "topics"
    - Publishers label events with topics
  - Content-Based
    - Subscribers specify interests through event filters
    - Published event content matched against filters

**Topic-Based Pub-Sub Systems**
- **Operations:**
  - Create
  - Publish
  - Subscribe
  - Unsubscribe
- **Costs:**
  - Transmission Cost
  - Maintenance Cost
**Topic Clusters**

- **How to decrease maintenance cost?**
  - Group topics with similar sets of subscribers into a "virtual" topic
  - Unify underlying supporting structures
  - Eliminate redundancy in structures of individual topics

- **Side effects:**
  - Users can subscribe to topics or topic-clusters
  - Need local filters to eliminate irrelevant events

**Cost Model**

- **Maintenance Cost:**
  \[
  MC = \sum_{c \in C} \text{cost}_m(\text{size}(t)) + \sum_{c \in C} \text{cost}_m(\text{size}(c)) + \text{cost}_c(\{c\})
  \]

- **Dissemination Cost:**
  \[
  DC = \sum_{c \in C} \text{freq}(t) \times (\text{cost}_d(\text{size}(t)) + \sum_{c \in C} \text{freq}(c) \times (\text{cost}_d(\text{size}(c))))
  \]

- **Overall Cost:**
  \[
  OC = MC + DC
  \]

**Dynamic Clustering Algorithm**

- **Cluster Update Operations:**
  - Topic 1 + Topic 2 → New Cluster
  - Topic + Cluster → New Cluster
  - Cluster 1 + Cluster 2 → New Cluster
  - Old Cluster → New Cluster + Topic
  - Old Cluster → Topic 1 + Topic 2

- **Algorithm:**
  - Perform Update if and only if it is beneficial

**Adding Topic to Existing Cluster 1/2**

- **Benefit Estimation:**
  - Update is beneficial when:
  \[
  |d| > \frac{1 + \text{freq}(t)(S_c + 1)}{1 + \text{freq}(t)}
  \]

- **Triggering Benefit Estimation:**
  - Frequency of events in topic
  - Number of subscribers to cluster
  - Size of subscriber subset registered to topic

**Adding Topic to Existing Cluster 2/2**

- **Trigger Benefit Estimation Probabilistically:**
  - When user subscribes to topic \( t \), perform benefit estimation with probability \( \omega \).
  - Conduct survey among \( c \) cluster's subscribers.
  
  \[
  |\mathcal{V}| = \left( \frac{\text{fraction of common subscribers to } t \text{ and } c}{\text{cluster size}} \right) \cdot (\text{cluster size})
  \]

- **Notes:**
  - \( \omega > 1 - (1 - \varphi)^1 \)
  - where \( \varphi \) = configuration parameter

**Merging two Clusters 1/3**

- **Benefit Estimation:**
  - Update is beneficial when:
  \[
  |d| > \frac{\alpha P_1 \cdot \alpha + aP_2 \cdot \beta + \alpha P_1 \cdot \beta + \alpha P_1 \cdot \beta + \alpha P_1 \cdot \beta + \alpha P_1 \cdot \beta + \alpha P_1 \cdot \beta + \alpha P_1 \cdot \beta + \alpha P_1 \cdot \beta}{\alpha + \beta}
  \]

- **Triggering Benefit Estimation:**
  - Frequency of events in topics
  - Number of topics in each cluster
  - Number of subscribers in each cluster
  - Number of common topics and subscribers
Merging two Clusters 2/3

- Estimate number of common topics:
  \[
  c_1, c_2, c_3, c_4
  \]
  Topics
  \[
  t_1, t_2, t_3, t_4
  \]
  Clusters
  \[
  t_1 \rightarrow c_1, t_2 \rightarrow c_1, t_3 \rightarrow c_2, t_4 \rightarrow c_2
  \]
  Topics in each cluster
  \[
  l_1, l_2, l_3, l_4
  \]
  List of other clusters each topic belong to
  - Number of common topics = number of times \( c_2 \) appears in the lists of \( c_1 \)
  - Lists maintenance through cluster update notifications

Merging two Clusters 3/3

- Estimate number of common subscribers:
  - Maintain overall number of subscribers
  - Sample subscribers periodically to learn which other clusters they are subscribed to
    \[
    \text{Fraction of common subscribers to } c_i \text{ and } c_j = \left( \frac{\text{number of subscribers}}{\text{number of common topics}} \right)
    \]
  - (Other cluster update notifications)

Subscriptions

- Optimal Subscription Problem:
  - Find set of topics and clusters to subscribe, that cover all interests and minimizes cost
  - NP complete

- Greedy Subscription Algorithm
  - Treat each individual request for adding/removing a topic heuristically

Greedy Subscription Algorithm

- User \( s \) wishes to subscribe to topic \( t \)
  - If \( t \) belongs to a cluster \( s \) is subscribed to,
    update the filter
  - Otherwise two choices:
    - Subscribe \( s \) to \( t \) directly
    - Subscribe \( s \) to new cluster that includes \( t \)

Implementation

<table>
<thead>
<tr>
<th>Tamara Demo Application</th>
<th>Tamara API</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tamara Core</td>
<td>P/S API</td>
</tr>
<tr>
<td>PubSub/Subscribe</td>
<td>DHT API</td>
</tr>
<tr>
<td></td>
<td>DHT</td>
</tr>
<tr>
<td></td>
<td>Internet</td>
</tr>
</tbody>
</table>

Experiments on Synthetic Data 1/4

- Events Frequency Models
  \[
  \text{(a) Average} \quad \text{(b) Frequent} \quad \text{(c) Infrequent}
  \]

- Quality of Clustering
  \[
  \begin{array}{|c|c|c|c|}
  \hline
  \text{Event/Alg} & \text{Precision} & \text{Recall} & \text{F1}\text{-score} \\
  \hline
  \text{Average} & 0.21 & 0.96 & 0.21 \text{ (en)} & 0.37 \text{ (en)} \\
  \text{Frequent} & 0.37 & 0.96 & 0.37 \text{ (en)} & 0.57 \text{ (en)} \\
  \text{Infrequent} & 0.11 & 0.69 & 0.11 \text{ (en)} & 0.11 \text{ (en)} \\
  \hline
  \end{array}
  \]
Experiments on Synthetic Data 2/4

- Operation Cost Over Time

Experiments on Synthetic Data 3/4

- Effect of User Subscription Rate

Experiments on Synthetic Data 4/4

- Effect of Maintenance Interval

Discussion

- Stability of Dynamic Algorithm is not addressed
- Functions used in Cost Model are too generic
  - Do not take network distance into consideration
- Adjustment cost is ignored
- Excessive use of heuristics
  - Triggering benefit estimation process
  - Greedy subscription algorithm