Data Centric Networking (R202)

Scalability and Expressiveness of Event Notification Services

Routing in Content-Based Networks

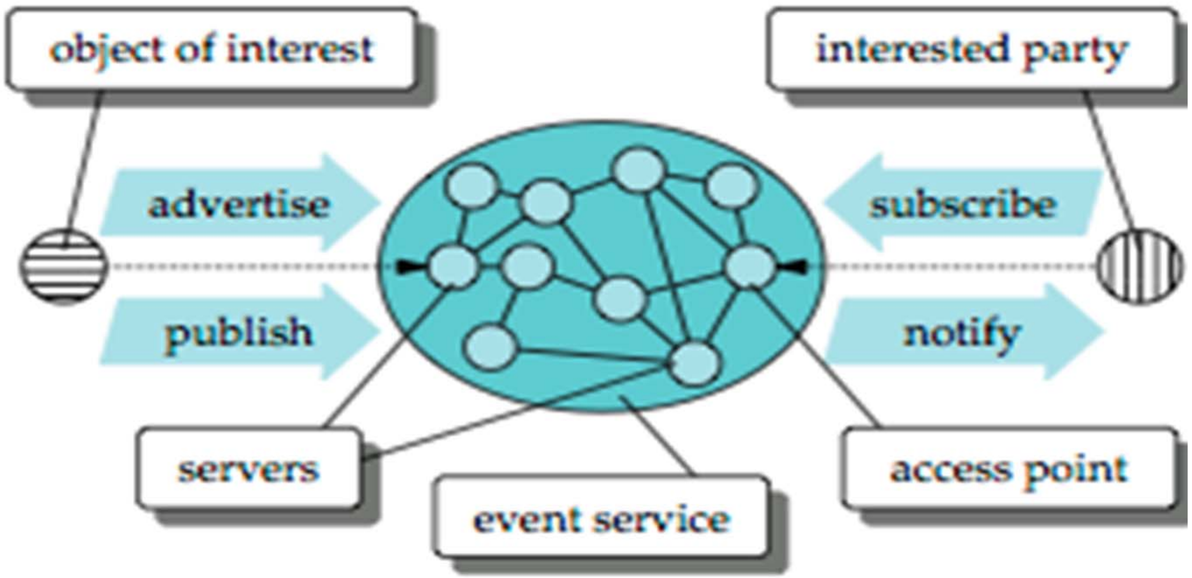
Todor Minchev
27 January 2011

UNIVERSITY OF CAMBRIDGE

Event-notification systems

- Collection of loosely-coupled autonomous components
- Components interact by emitting notifications
- System performs actions in response to events
- Provides middleware infrastructure for component interaction

SIENA



- Network of distributed servers
- Access points provide pub/sub interface
- Objects of interest (providers)
- Interested parties (consumers)
- SIENA matches subscribers with providers
- Best effort service
- Possible race conditions

A. Carzaniga, D.S. Rosenblum, A.L. Wolf

Scalability & Expressiveness

- Number of subscribers & publishers
- Number of notifications & subscriptions
- Power of the model
- Expressiveness affects the routing and forwarding algorithms
- Trade off between expressiveness & scalability

The SIENA data model

- Advertise in addition to pub/sub interface functions
- Unsubscribe and unadvertise
- Untyped notifications

string	*class*	=	*finance/exchanges/stock*
time	*date*	=	*Mar 4 11:43:37 MST 1998*
string	*exchange*	=	*NYSE*
string	*symbol*	=	*DIS*
float	*prior*	=	*105.25*
float	*change*	=	*-4*
float	*earn*	=	*2.04*

A. Carzaniga, D.S. Rosenblum, A.L. Wolf

Notification Selection

- Event filters – single notification matcher

$$\begin{array}{||l r c l||}
string & class & >* & finance/exchanges/ \\
string & exchange & = & NYSE \\
string & symbol & = & DIS \\
float & change & < & 0
\end{array}$$

- Patterns – multiple notifications matcher

$$\begin{array}{||l r c l||}
string & what & >* & finance/exchanges/ \\
string & symbol & = & MSFT \\
float & change & > & 0
\end{array}$$

$$\bullet$$

$$\begin{array}{||l r c l||}
string & what & >* & finance/exchanges/ \\
string & symbol & = & NSCP \\
float & change & > & 0
\end{array}$$

- Pattern considerations – latency & timestamps

A. Carzaniga, D.S. Rosenblum, A.L. Wolf

Covering Relations

Subscription filter– multiple constraints interpreted as a conjunction

$$f \sqsubseteq_S^N n \Leftrightarrow \forall \phi \in f : \exists \alpha \in n : \phi \sqsubseteq \alpha$$

subscription		notification
string what = alarm	$\sqsubseteq_S^N$	string what = alarm time date = 02:40:03
string what = alarm integer level > 3	$\not\sqsubseteq_S^N$	string what = alarm time date = 02:40:03
string what = alarm integer level > 3 integer level < 7	$\not\sqsubseteq_S^N$	string what = alarm integer level = 10
string what = alarm integer level > 3 integer level < 7	$\sqsubseteq_S^N$	string what = alarm integer level = 5

Covering Relations

Advertisements

$$a \sqsubset_A^N n \Leftrightarrow \forall \alpha_n \in n : \exists \phi_a \in a : \phi_a \sqsubset \alpha_n$$

advertisement		*notification*
string what = *alarm* *string what* = *login* *string user any*	$\sqsubset_A^N$	*string what* = *alarm*
string what = *alarm* *string what* = *login* *string user any*	$\not\sqsubset_A^N$	*string what* = *alarm* *time date* = *02:40:03*
string what = *alarm* *string what* = *login* *string user any*	$\sqsubset_A^N$	*string what* = *login* *string user* = *carzanig*
string what = *alarm* *string what* = *login* *string user any*	$\not\sqsubset_A^N$	*string what* = *logout* *string user* = *carzanig*

SIENA Architecture

Interconnection topology

- Hierarchical client/server topology
- Acyclic peer-to-peer
- General peer-to-peer

Routing algorithm

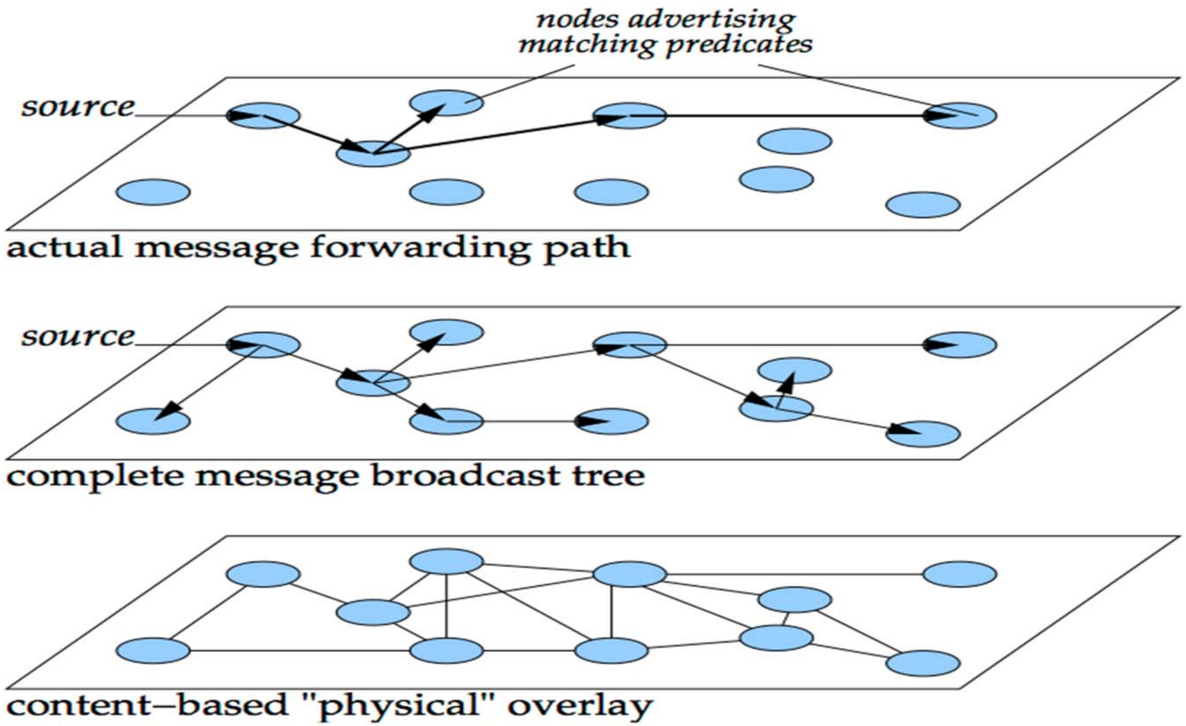
- Downstream replication
- Upstream evaluation
- Subscription forwarding
- Advertisement forwarding

Routing in a Content Based Network

- No explicit destination addresses
- Content matched against predicates
- Propagate predicates and topological information
- The objective is loop-free and minimal forwarding paths

Network Architecture

- Overlay point-to-point
- Routing – synthesizing distribution paths
- Forwarding – determining the next hop destinations



A. Carzaniga, M.J. Rutherford, A.L. Wolf

Combined Broadcast and Content-Based (CBCB) protocol

Router runs two protocols

- Broadcast routing protocol
- Content-based routing protocol
  - Processes predicates defined by nodes
  - Based on “push-pull” mechanism
  - Set of typed attributes(messages) exchanged by routers
  - Nodes implement interface functions

  send_message(m) and set_predicate(p)

Broadcast Layer

- Delivers messages from any node to any other node
- Propagates routing information
- Broadcast function B(s,i) called for each node – set of output interfaces
- Can be implemented as minimal spanning tree or shortest-path tree

Content-based layer

- Delivers messages only to interested parties
- Content-based address of a node is defined as predicate
- p(m) & selects(p)
- Covering relation between content-based addresses (p2 is covered by p1)

$$\forall m : p_2(m) \Rightarrow p_1(m)$$

$$p_2 \prec p_1$$

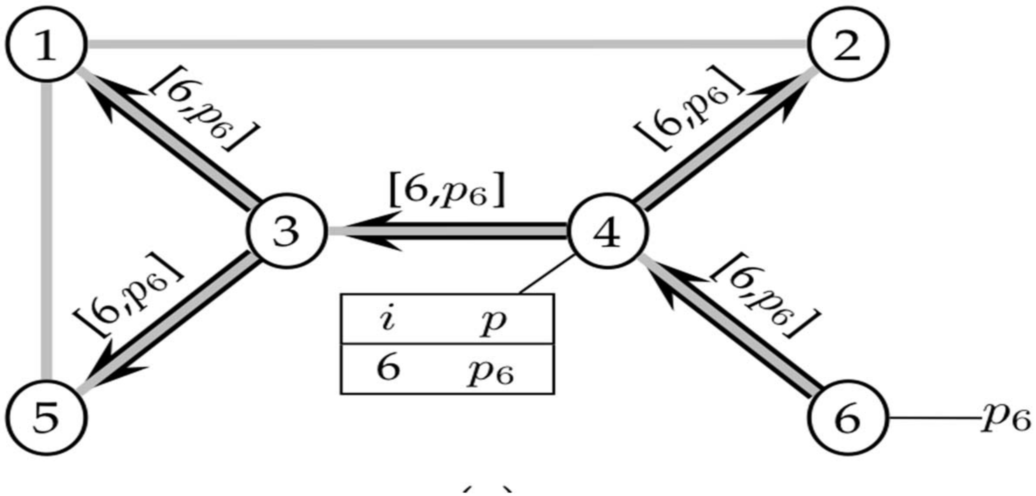
Content-Based Routing Protocol

- Two mechanisms for propagation of routing information
  - Push ***receiver advertisements(RA)***
  - Pull ***sender requests(SR)*** and ***update replies(UR)***
- RAs issued periodically when the p_0 changes
- Push routing information from receiver to potential senders

issuer
predicate
...

Receiver Advertisements Propagation

- Content-based RA ingress filtering
- Broadcast RA propagation – B(r,i) to compute the outgoing interfaces
- Routing table update (routing table inflation)



Sender Requests Propagation

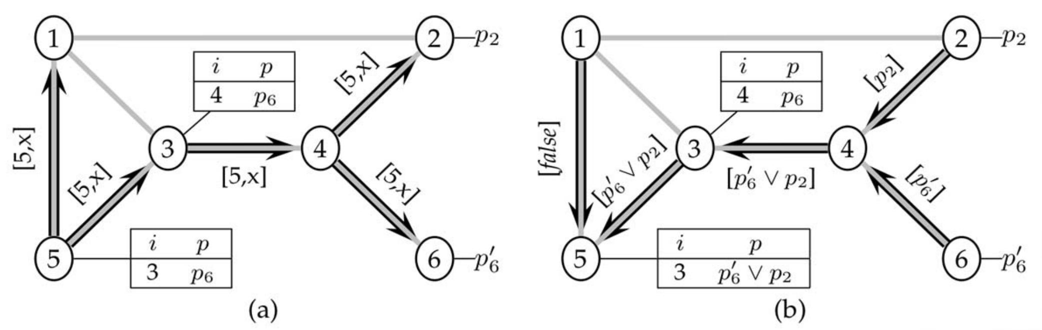
- Pull content-based addresses from all receivers
- SRs result in Update Replies (URs)
- Balances routing table inflation
- Compensates for lost RAs

issuer
request number
timeout

Update Replies

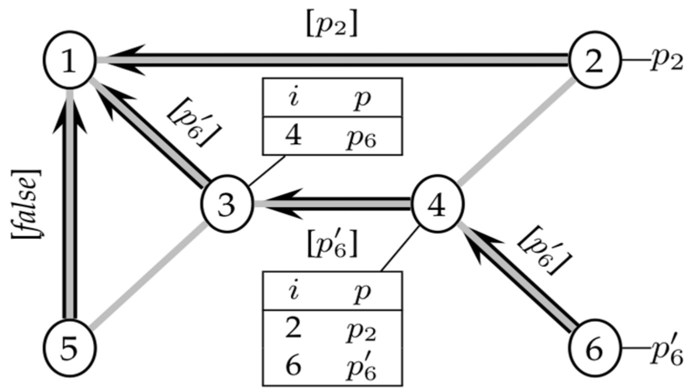
- Sent upstream only in reply to SRs
- Leaf nodes send UR immediately
- Non-leaf node adds their p_0 to URs
- SR issuer updates its routing table for incoming interface with UR

SR issuer
SR number
predicate ...



SR/UR Optimization

- N routers -> each SR generates 2N packets
- Limit the use of SRs to selected interfaces
- Cache and reuse URs
- Message counter linked with an outgoing interface



Discussion

- How secure is SIENA in its current form?
- Can SIENA be used to prevent the distribution of copyrighted material by forcing publishers and subscribers to use special attributes in their notifications?
- How scalable is the system? Can it support hundreds, thousands or millions of nodes?
- Can a quality-of-service (QoS) mechanism be implemented into the proposed routing protocol?
- Are there too many/few control messages exchanged between nodes?
- Do you think that this routing protocol will scale well in a network with millions of hosts?