On purely pub/sub security protocols
— or — vice versa?

Pekka Nikander, Ericsson Research Nomadic Lab
Giannis F. Marias, Athens University of Econ. and Business

16th SPW, April 17 2008, Cambridge, UK

Outline

- Why pure publish/subscribe?
  - DoS, applications, optics & radio
- What is *pure* publish/subscribe?
  - Only information names
  - No receiver/sender names
- Security in pure pub/sub
  - Mostly open questions

Why?

- Better resistance against flooding DoS
  - Receiver’s consent needed
- More natural to many applications
  - Content delivery, asynchronous message delivery (e.g. e-mail), even transactions
  - Maybe efficient for all traffic (incl. interactive)
- Better fit with modern physical layers
  - All optical, mesh radio, sensor, …

The real reasons…

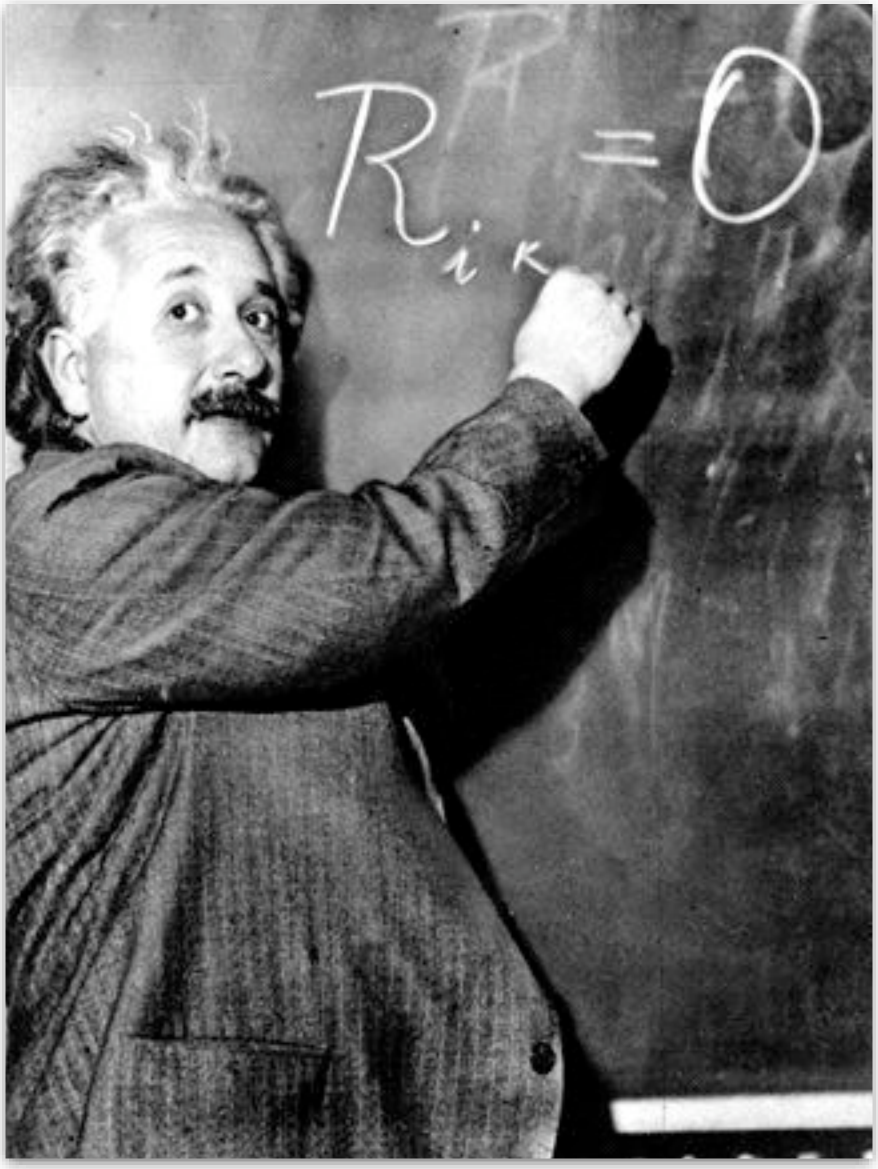
- A project for doing something really interesting

- Clean slate (get rid of IP)
- Apply state of the art
  - Econosec, mechanism design, Theory U, …

- Try to be as different as possible
  - Re-doing IP would be boring…

What?

- Network as a (rough) extension of the blackboard IPC paradigm
- Each scribble (piece of info) tagged with a unique tag
- Receivers and senders are anonymous (to the net)
- We'll handle scalability
    - scoping, recursion, multicast, caching, some clever tricks, …



Some characteristics

- Network does more
  - Matching pubs and subs
  - Caching messages
- Network has many parts
  - Do we need to trust some of them?
  - If so, how much and why?
- Tags can be long, becoming semi-private
  - Work as “weak” cryptographic keys

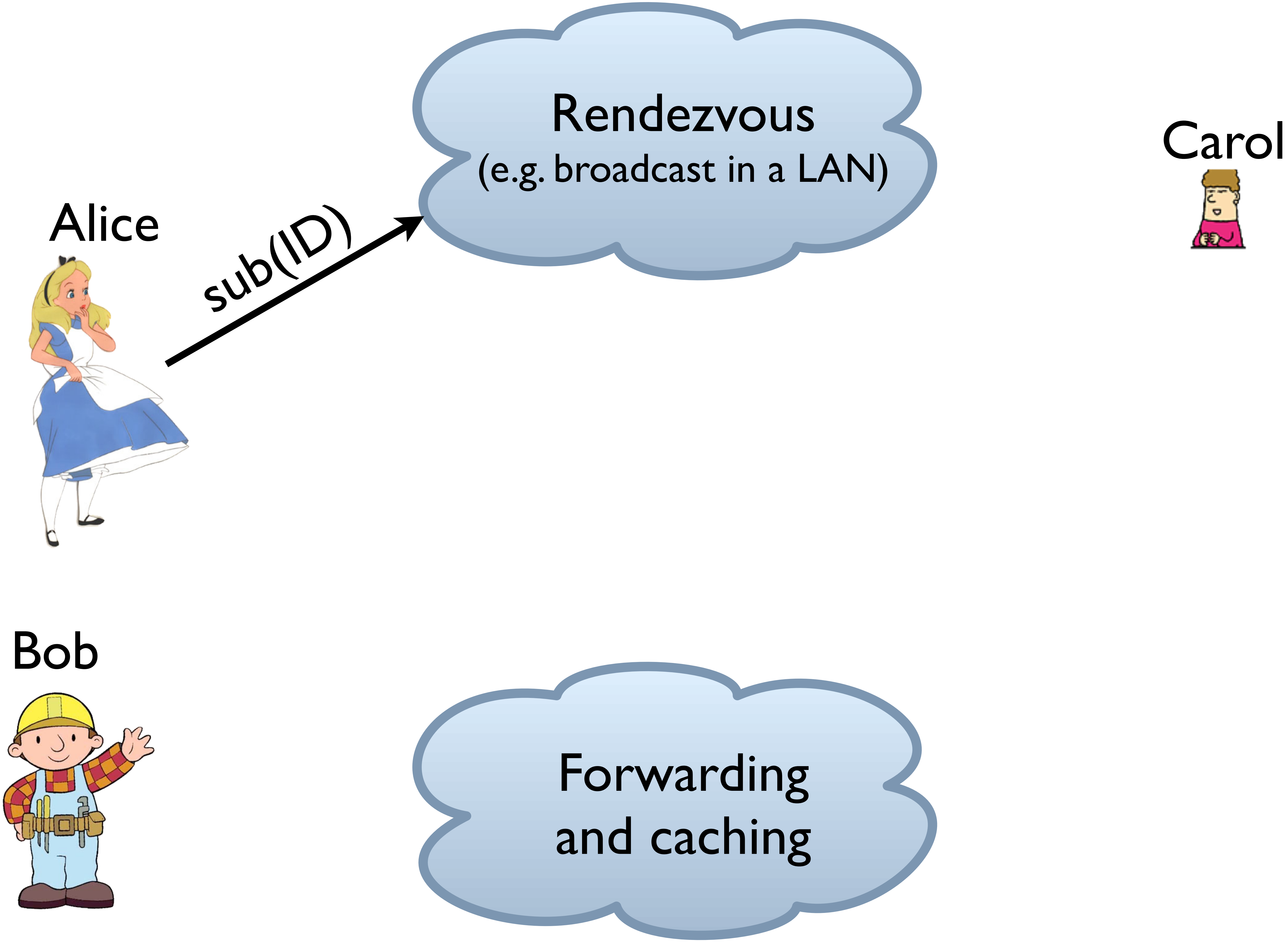
Rendezvous
(e.g. broadcast in a LAN)

Carol

Alice

Bob

Forwarding
and caching



Rendezvous

(e.g. broadcast in a LAN)

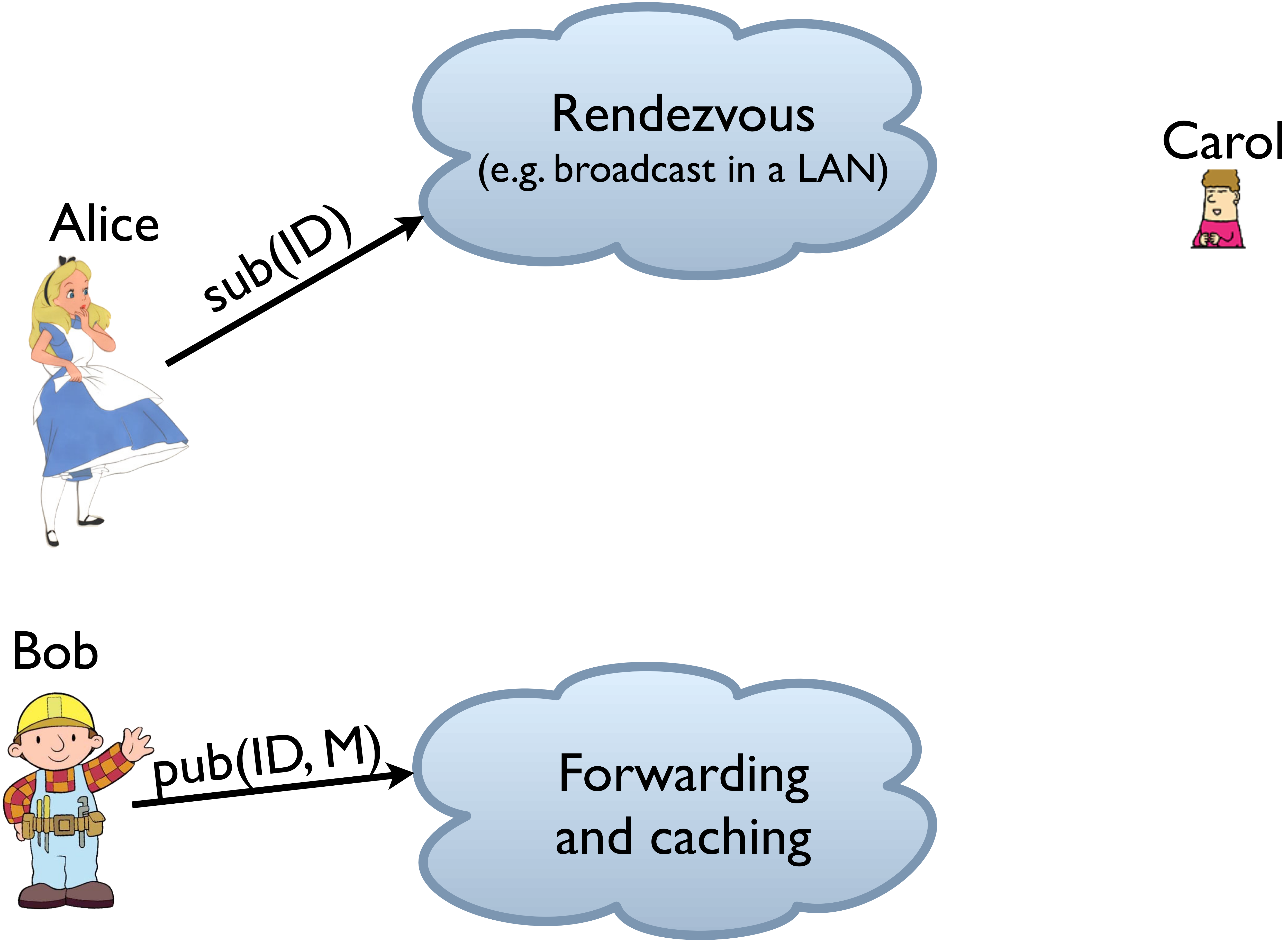
Carol

Alice

sub(ID)

Bob

Forwarding and caching



Rendezvous
(e.g. broadcast in a LAN)

Carol

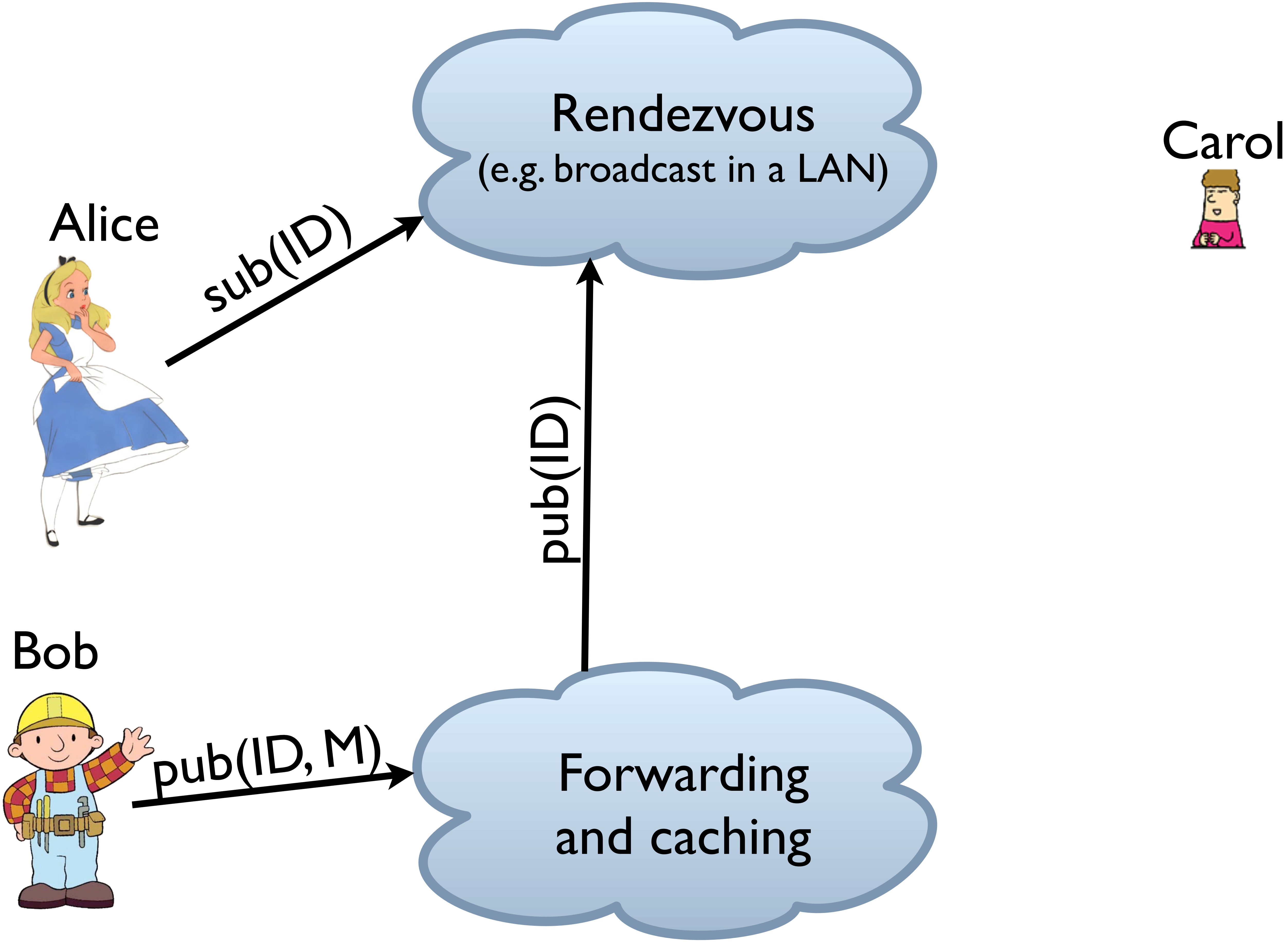
Alice

sub(ID)

Bob

pub(ID, M)

Forwarding
and caching



Rendezvous
(e.g. broadcast in a LAN)

Carol

Alice

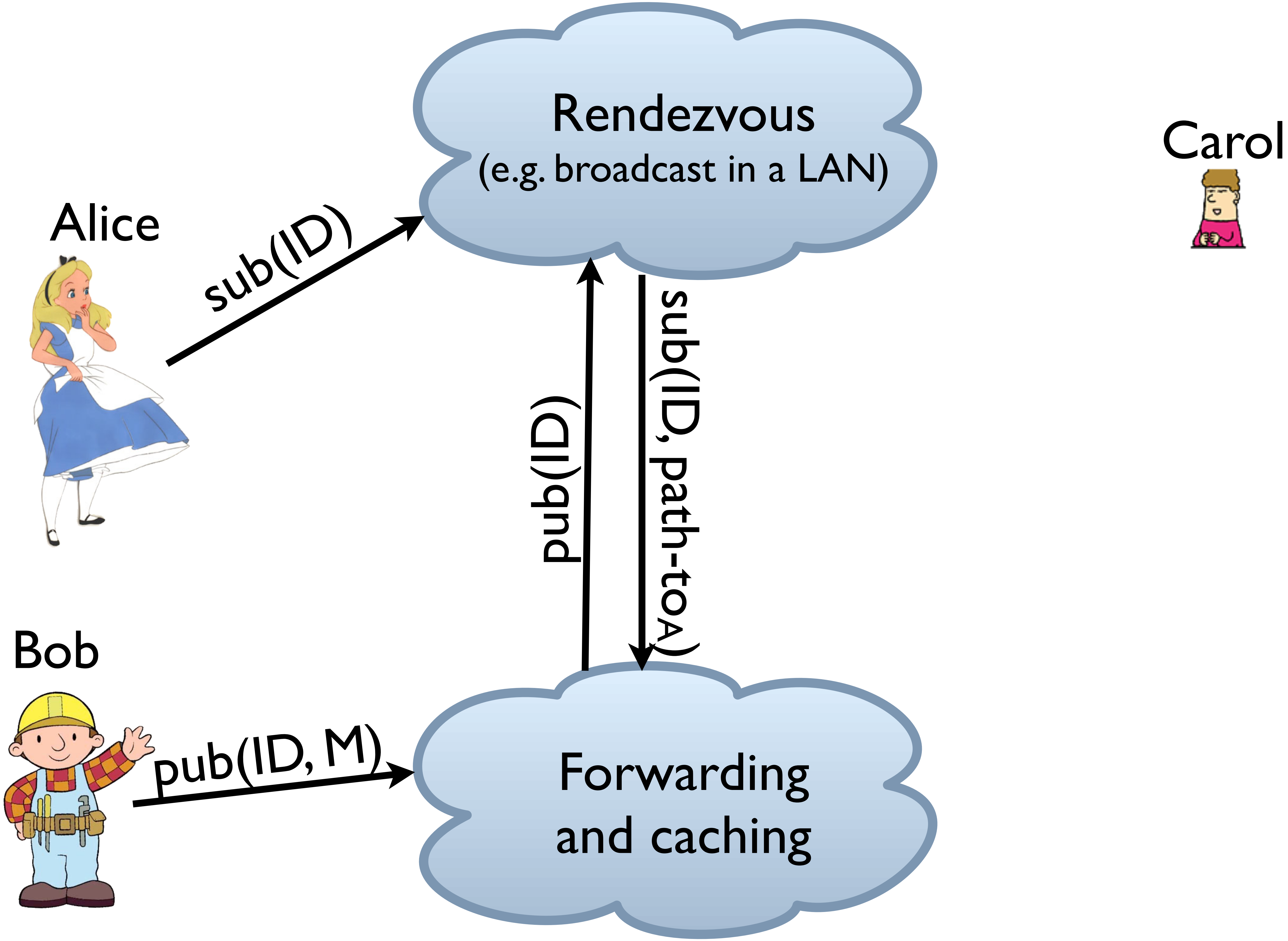
sub(ID)

pub(ID)

Bob

pub(ID, M)

Forwarding
and caching



Rendezvous

(e.g. broadcast in a LAN)

Carol

Alice

sub(ID)

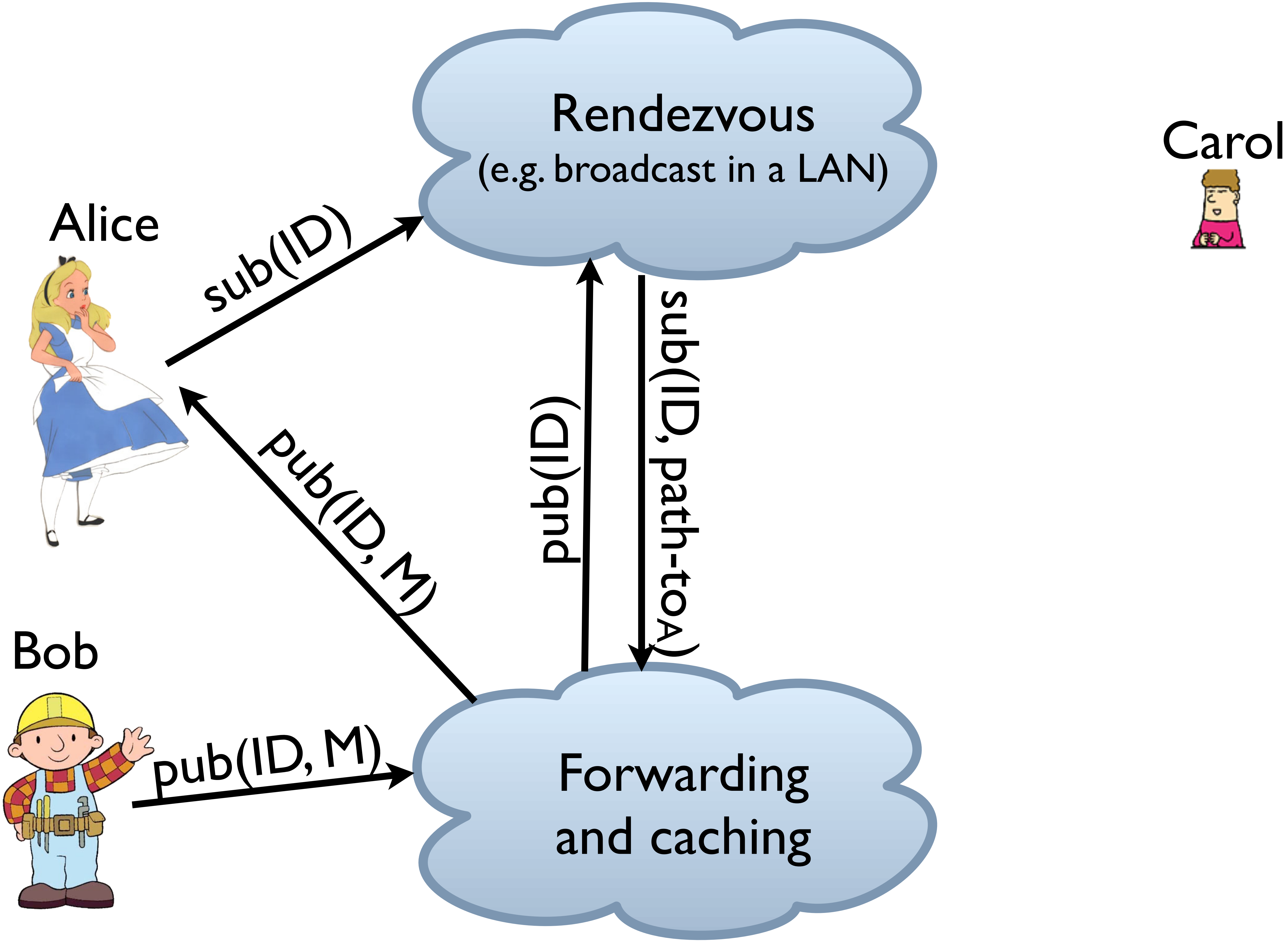
pub(ID)

sub(ID, path-toA)

Bob

pub(ID, M)

Forwarding and caching



Rendezvous
(e.g. broadcast in a LAN)

Carol

Alice

sub(ID)

pub(ID, M)

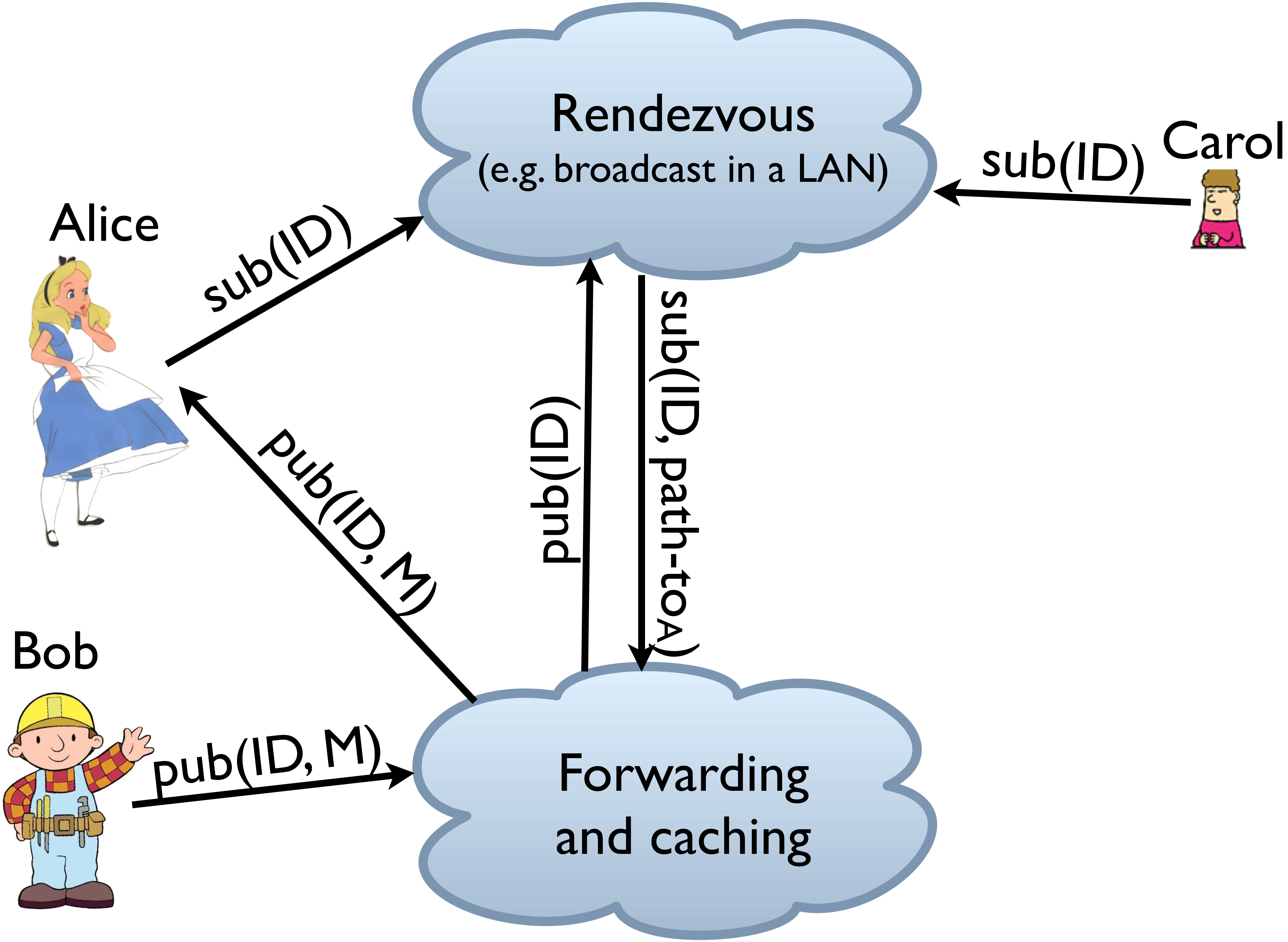
pub(ID)

sub(ID, path-toA)

Bob

pub(ID, M)

Forwarding
and caching



Rendezvous
(e.g. broadcast in a LAN)

Carol

sub(ID)

Alice

sub(ID)

pub(ID, M)

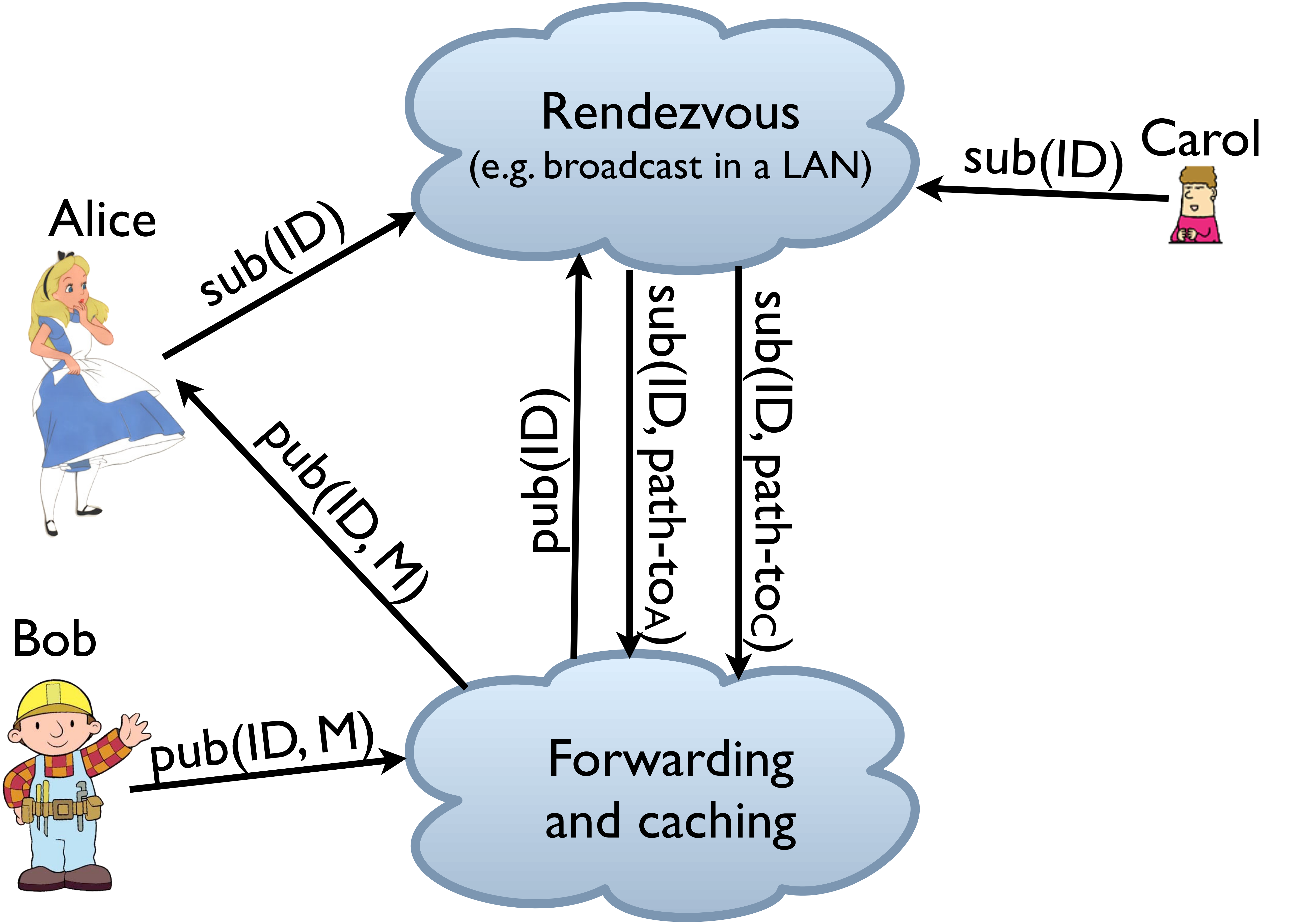
pub(ID)

sub(ID, path-to$_A$)

Bob

pub(ID, M)

Forwarding and caching



Rendezvous

(e.g. broadcast in a LAN)

Carol

sub(ID)

Alice

sub(ID)

pub(ID, M)

pub(ID)

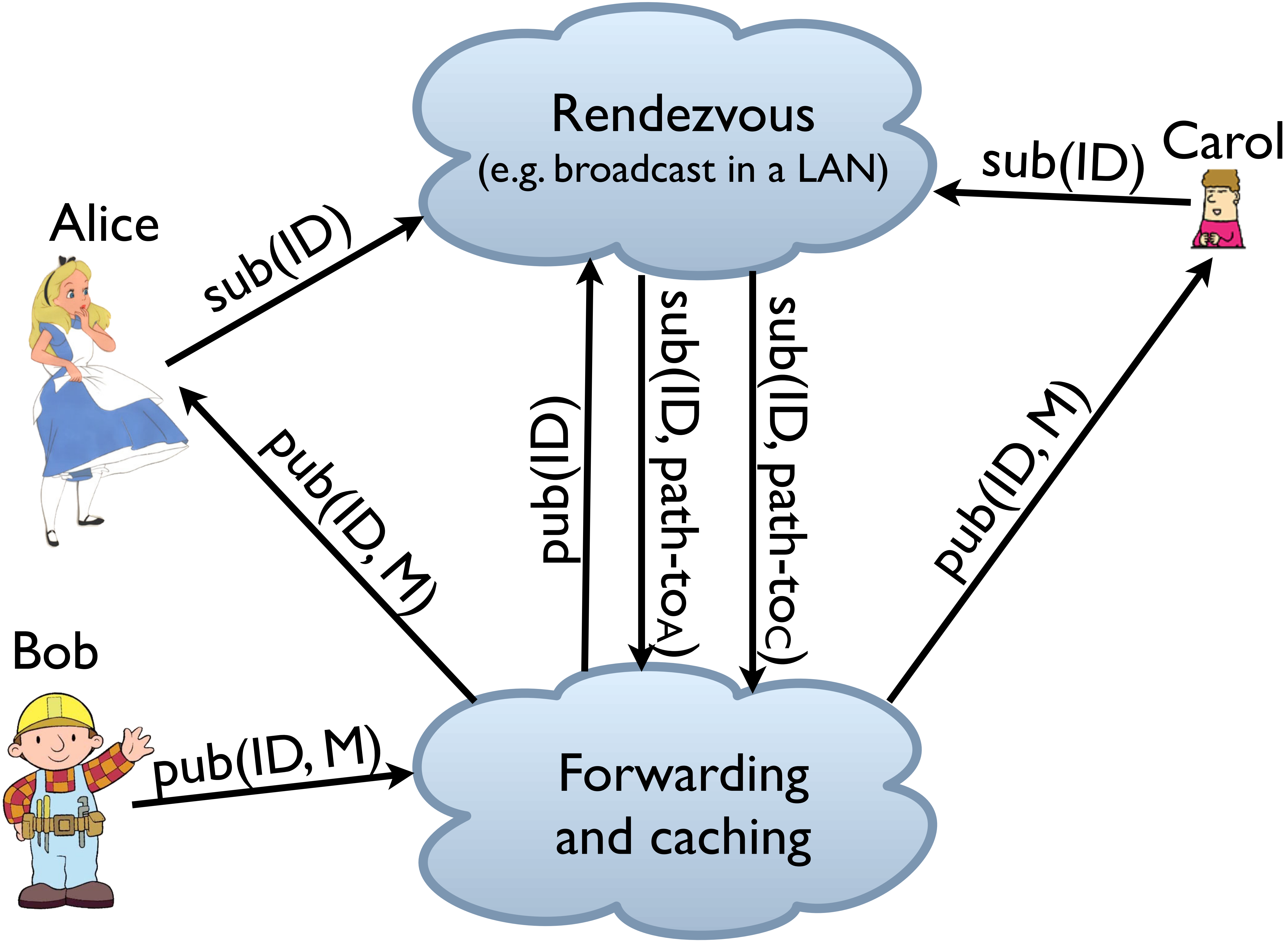
sub(ID, path-toA)

sub(ID, path-toC)

Bob

pub(ID, M)

Forwarding and caching



Rendezvous

(e.g. broadcast in a LAN)

Alice

sub(ID)

Carol

sub(ID)

pub(ID, M)

pub(ID)

sub(ID, path-toA)

sub(ID, path-toC)

pub(ID, M)

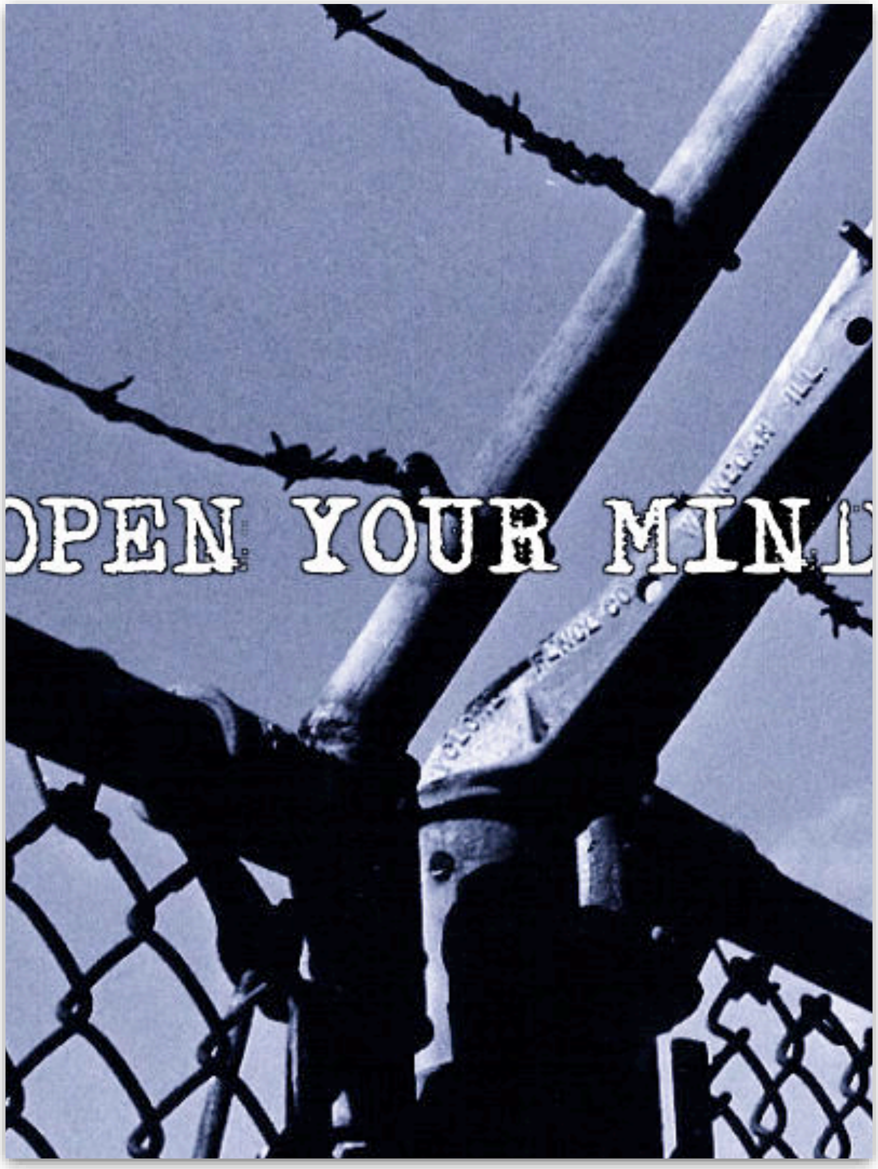
Bob

pub(ID, M)

Forwarding and caching

Security

- Early ideas
  - Integrity and modification of messages
  - Message composition
  - Algorithmically computed message IDs
- Some open questions



Integrity and modification of individual messages

- Static, immutable data
  - ID = hash(data)
  - Algorithm agility? (Have multiple IDs?)
- Mutable messages
  - ID = pk:tag
    - Or use hash(pk) if it pk doesn’t fit
  - message = < ID, data, ..., timestamp, seq#, sig >

Message composition

- Immutable, large publications
  - ID = hash(metadata)
  - metadata = < ID_1, ID_2, …, ID_n, hash(content) >
- Mutable large publications
  - ID = pk:tag
  - metadata = < ID, ID_1, …, ID_n, …, seq#, sig >

- Sequences (e.g. real time voice): next slide

Sequences

- Sequence of messages to be sent
  - Content not known beforehand

- ID = hash(metadata)
- metadata = an algorithm for creating IDs

Open questions 1: Fundamentals

- How to model authentication of (complex) data instead of authentication of principals?
  - How does this translate to transactions?
  - What is the semantics of message composition?
- Group communication questions
  - How to model multicast? Concast?
  - How to secure concast against DoS?
- What is the role of the infrastructure?
  - Resource control? Fairness? Compensation?

Open questions 2: Modelling

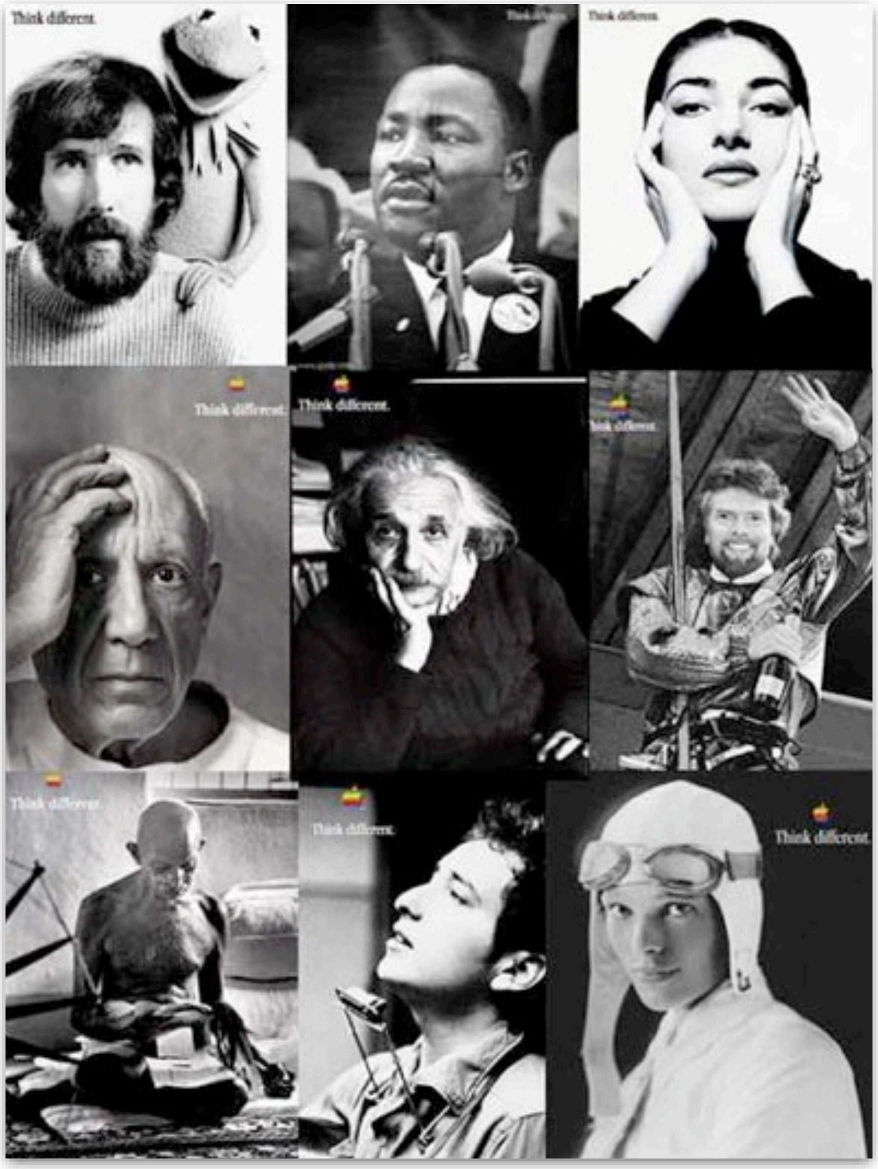
- How to model the assumptions, goals, and beliefs?
  - Some principals may be anonymous or pseudonymous
  - Even the basic communication beliefs may change, e.g.

$$A \mid\equiv \{\ \exists B:\ B \mid\sim \mathrm{sub}(\mathrm{ID}) \land \ldots\ \}$$

- How to model the network?
  - Seems fairly easy with Spi calculus or strand spaces…
- Information theoretic models?
  - We've got no clue here

Summary

- Think different
- Network as a rough extension of the (tagged) blackboard
  - No principal names
- Lots of open questions
  - Read the paper ☺



Backup slide:
a bootleg formal model for OR

$$
\begin{array}{lll}
(or_1) \rightarrow: & (N_1). & I_R; A; B; \{|N_1; I_R; A; B|\}^s_{K_{AS}} \\
(or_2) \rightarrow: & (N_2). & I_R; A; B; \{|N_1; I_R; A; B|\}^s_{K_{AS}}; \{|N_2; I_R; A; B|\}^s_{K_{BS}} \\
(or_3) \rightarrow: & (K). & I_R; \{|N_1; K|\}^s_{K_{AS}}; \{|N_2; K|\}^s_{K_{BS}} \\
(or_4) \rightarrow: & & I_R; \{|N_1; K|\}^s_{K_{AS}}
\end{array}
$$

$$
\begin{array}{l}
r(I_{AB}, I_R; A; B; \gamma_1). \\
f(N_2). \\
p(I_{BS}, I_R; A; B; \gamma_1; \{|N_2; I_R; A; B|\}^s_{K_{BS}}). \\
r(I_{SB}, I_R; \gamma_2; \{|N_2; K|\}^s_{K_{BS}}). \\
p(I_{BA}, I_R; \gamma_2)
\end{array}
$$