Middleware for Semantic Multicast in Spontaneous Multi-hop Networks

Paolo Bellavista
Carlo Giannelli

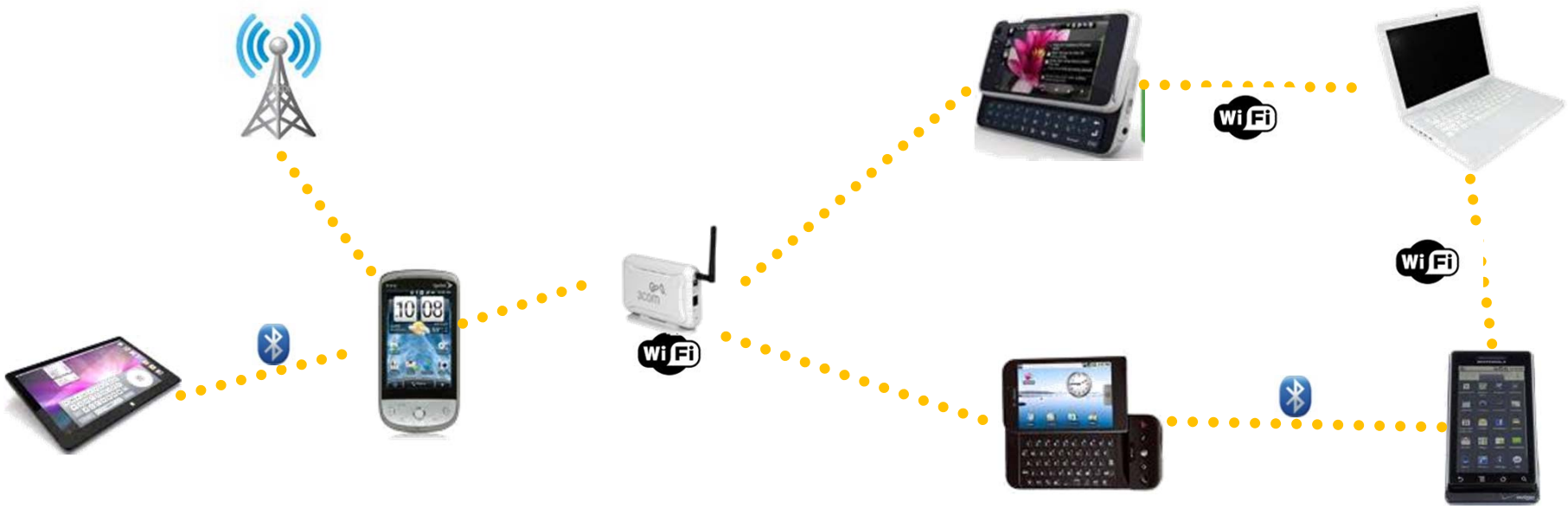
13.11.2012 - Mobilware'12

DISI, Università di Bologna
Viale Risorgimento, 2 - 40136 Bologna Italy
paolo.bellavista@unibo.it

Agenda

- ***Spontaneous networking***
  - opportunities and technical challenges
- ***3-layer multicast model for spontaneous networking***
  - traditional IP
  - spontaneous multi-hop
  - semantic dispatching
- ***Semantic-enabled packet delivery***
  - direct and blind multicast
  - semantic forward

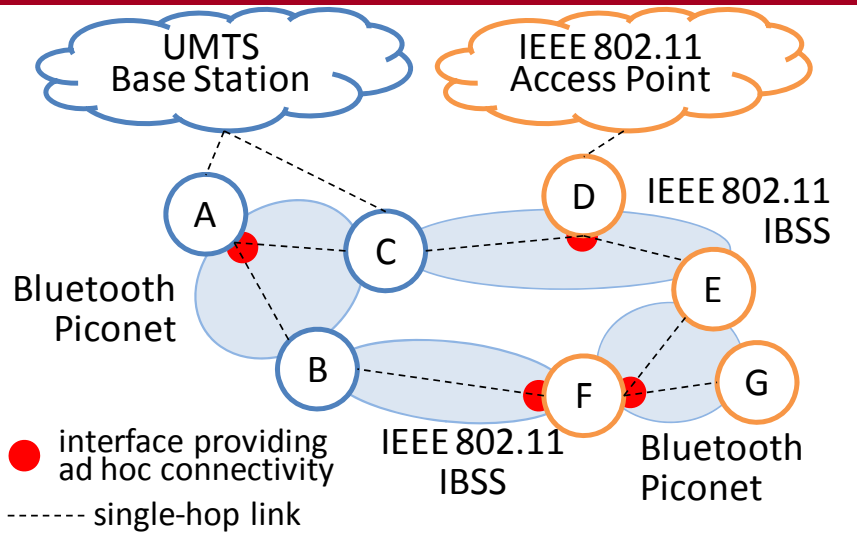
Spontaneous Networking (1)



- ***Impromptu interconnection of mobile and fixed nodes***
  - users willing to share content and resources
- ***Maximize resource/service utilization by potentially interconnected nodes***
  - heterogeneous wireless technologies
  - both infrastructure and ad-hoc connectivity
  - multiple connectivity opportunities
  - sporadic/opportunistic Internet connectivity

Spontaneous Networking (2)

- ***Node cooperation to***
  - provide single-hop connectivity
  - manage multi-hop connectivity
  - support peer-to-peer services



- ***Peer-to-peer resource sharing (Internet connectivity, file sharing, …)***
  - service advertising: NodeA provides lesson notes
  - service discovery: NodeF looks for nodes that share files
  - service invocation: NodeF browses and downloads notes stored on NodeA
- ***NodeA and NodeF may reside in different IP networks***

Spontaneous Networking: Technical Challenges

- ***Heterogeneous nodes and connectivity***
  - IEEE 802.11, Bluetooth, Ethernet
  - several operating systems
- ***Decentralized and loosely-coupled network management***
  - localized provisioning of layer-2/3 connectivity
  - interconnection of heterogeneous layer-3 networks
- ***Erratic and unpredictable behavior***
  - nodes abruptly create/destroy pieces of network
  - nodes dynamically join/move/leave

- ***Scenario and management complexity*** makes hard the development of novel applications from scratch => ***need for middleware solutions***

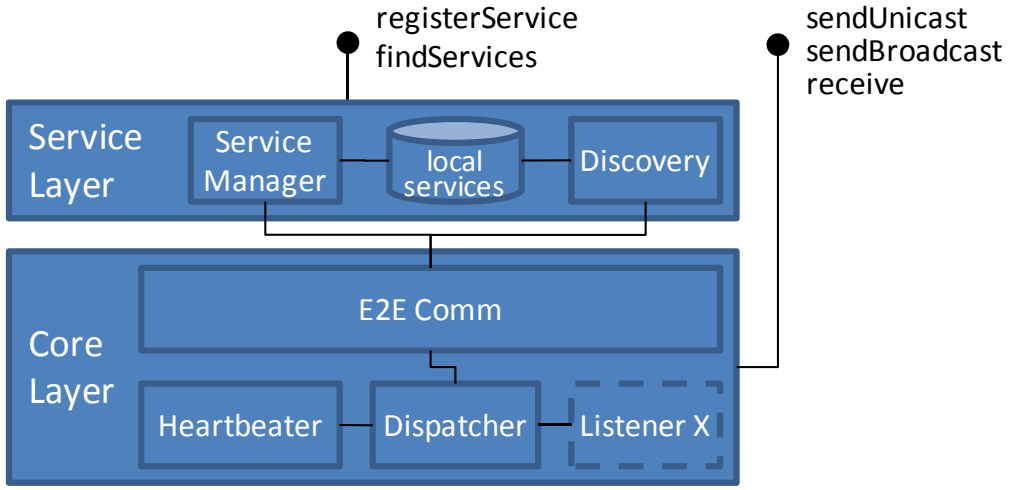
RAMP: Real Ad-hoc Multi-hop Peer-to-peer

- ***Easy-to-use middleware supporting spontaneous network management, transparent in relation to***
  - operating systems
  - wireless technologies
  - layer-3 network configurations
  - node mobility

- ***Unicast and broadcast comm. support***
  - per-packet `sendUnicast`, `sendBroadcast`, `receive`
- ***Peer-to-peer service provisioning and discovery***
  - per-service `registerService`, `findService`
- ***RAMP Java prototypes available for MS Windows XP/Vista/7, Linux, Mac OS X, and Android***

RAMP Architecture

- ***Service Layer***
  - high-level features for peer-to-peer service offering and discovery
  - Discovery: mission-oriented TTL-bound broadcast
  - ServiceManager: registration and advertising
  - service invocation via Core Layer



- ***Core Layer***
  - low-level primitives for end-to-end communication
  - E2EComm: communication primitives for data en/decapsulation into RAMP packets
  - Heartbeater for local IP addresses gathering and single-hop neighbors discovery
  - Dispatcher: actual inter-node packet forwarding
  - listener-based plug-in for run-time packet management

3-layer Multicast Model

- Spontaneous networking enables user-centric content sharing
  - ***huge volumes*** of user-generated contents, related to different categories, interests, topics, applications, …
  - content discovery can be ***resource consuming***, e.g., discovery and browsing of several file sharing service replicas to retrieve jazz music
    - additional complexity for final users
    - additional communication overhead
- Novel **3-layer multicast model** to retrieve content
  - ***different mechanisms*** for different scenarios
    - traditional IP, spontaneous multi-hop, semantic dispatching
  - multicast communication based on ***node cooperation***
  - ***different opportunities*** about ***which nodes*** are suitable to be ***receivers*** for multicast messages (see the following)

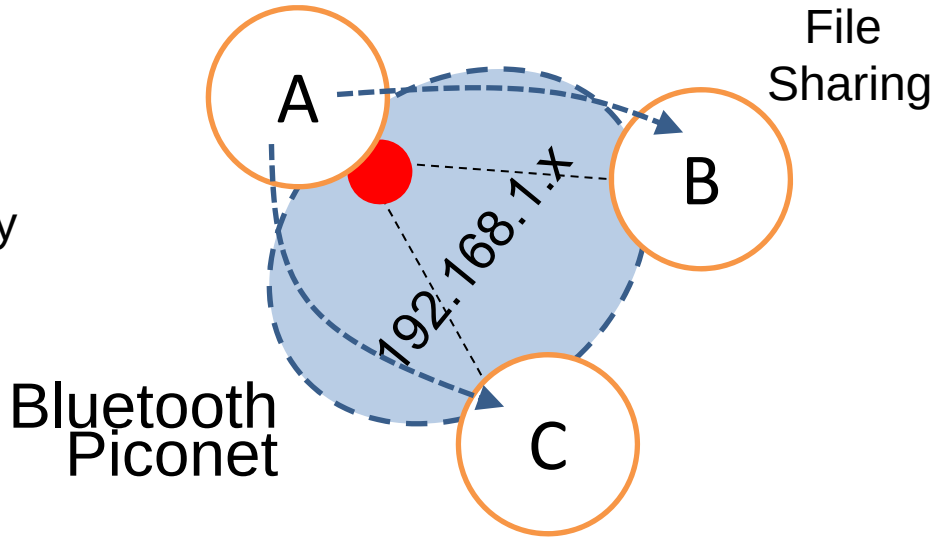
Traditional IP Layer

- Single IP subnet with ***private address space***
- Nodes identified by their IP address
- IP-based single-hop broadcast and multicast
- ***Well-known discovery protocols*** such as UPnP
- Node cooperation for ***service provisioning***

Service Discovery
(single-hop)

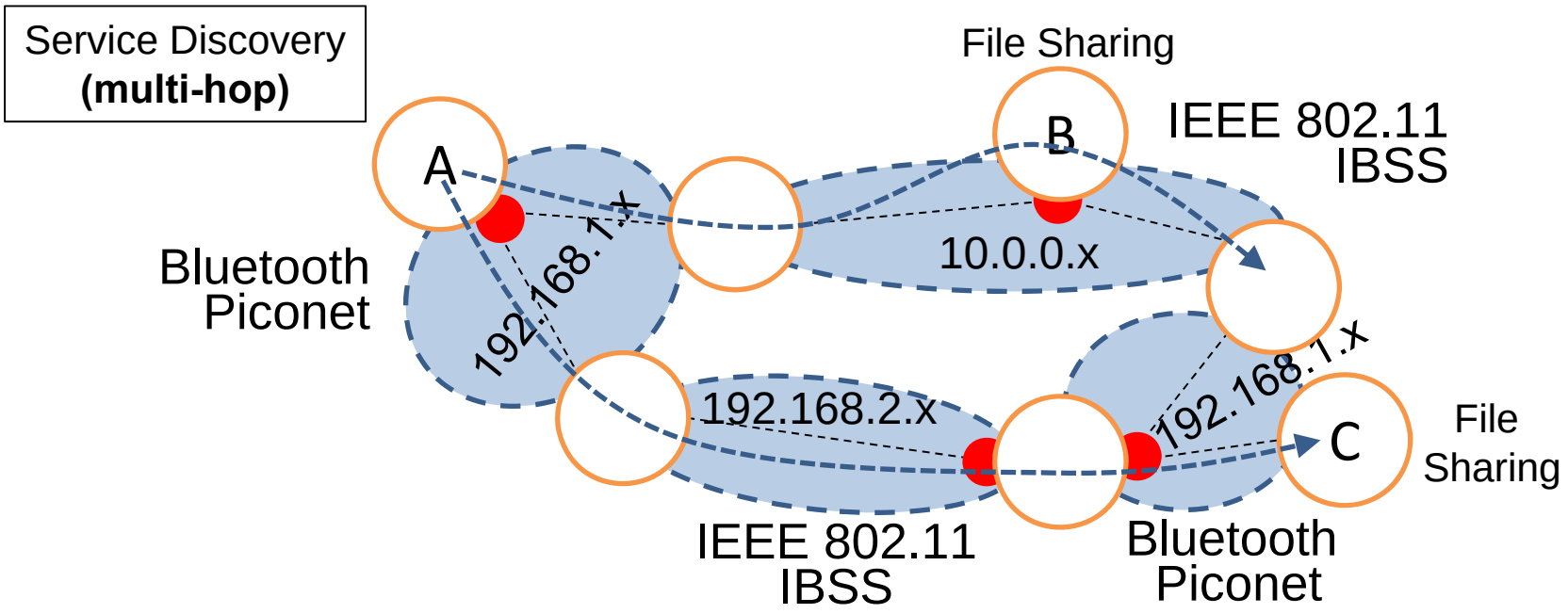
interface providing connectivity

discovery request



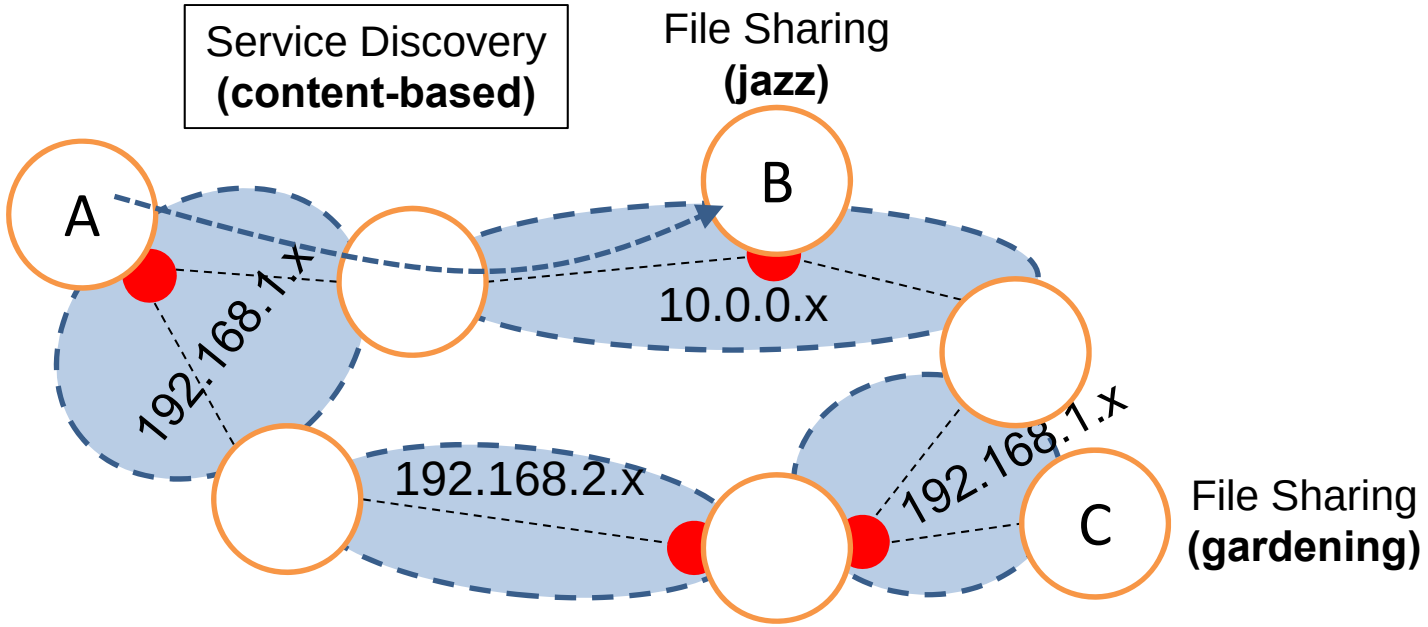
Spontaneous Multi-hop Layer

- Spontaneous networks composed of ***heterogeneous IP subnets***
- Node addresses as DSR-like IP sequences
- RAMP multi-hop service discovery based on ***syntactic service name matching***
- Node cooperation also for ***multi-hop packet dispatching***



Semantic Dispatching Layer

- Loosely coupled endpoints: ***senders specify receivers' characteristics*** in place of their identity or address
- **Intrinsically multicast/broadcast "by nature"**: senders know neither address nor identity of receivers
  - packets received only by actually interested nodes
- Node cooperation also to ***agree on formats*** to describe users' interest and to partially ***disclose users' data***



Middleware Solution for Semantic Dispatching Layer

- Middleware solution to ***transparently support semantic multicast***
  - based on ***partial*** spontaneous network knowledge
    - local user's data together with (a subset of) information about remote users
  - tradeoff among
    - ***correctness***: packets delivered only to interested nodes
    - ***efficiency***: minimizing processing and communication overhead

- ***Original transmission mechanisms for SNs***
  - semantic multicast: direct and blind
  - semantic forward

Semantic Multicast

- Senders ***specify receivers based on criteria***
  - scope of file sharing service discovery limited to users interested in jazz music
- Receivers may vary in relation to ***where*** and ***when*** criteria are checked, depending on direct/blind options (see the following)
  - senders may have a ***partial knowledge*** of receivers interests
  - receivers may have ***just joined*** the network
- Impact on packet delivery correctness and communication efficiency
  - Direct and Blind Multicast mechanisms to provide ***differentiated correctness/efficiency tradeoffs***

Direct Option

- Delivery ***criteria applied on senders***
  - sender node selects receivers ***based on its own knowledge*** of remote nodes
  - ***packets sent via unicast*** to locally retrieved nodes
  - receivers propagate packets to the application layer without any additional criteria verification
- ***No strict consistency*** of delivery semantic
  - the sender may have ***incomplete, not up-to-date knowledge*** of remote nodes based on
    - replies of previously launched queries
    - periodic broadcast of identities and public information
  - eventually false positives and false negatives
- Limited overhead
  - criteria applied ***only one time*** (at packet creation) and ***in only one location*** (on sender node)

Blind Option

- Delivery criteria applied on receivers
  - ***criteria attached to packets*** and sent via RAMP multi-hop ***broadcast***
  - receivers propagate packets to the application level only if criteria are verified
- ***Strict consistency*** of delivery semantic
  - ***receivers have full knowledge*** of their own context
  - also nodes the sender is not aware of may receive packets
- ***Additional overhead***
  - criteria delivered with packets
  - packets delivered even to not interested nodes
  - criteria verified at any potential node

Direct vs. Blind Multicast

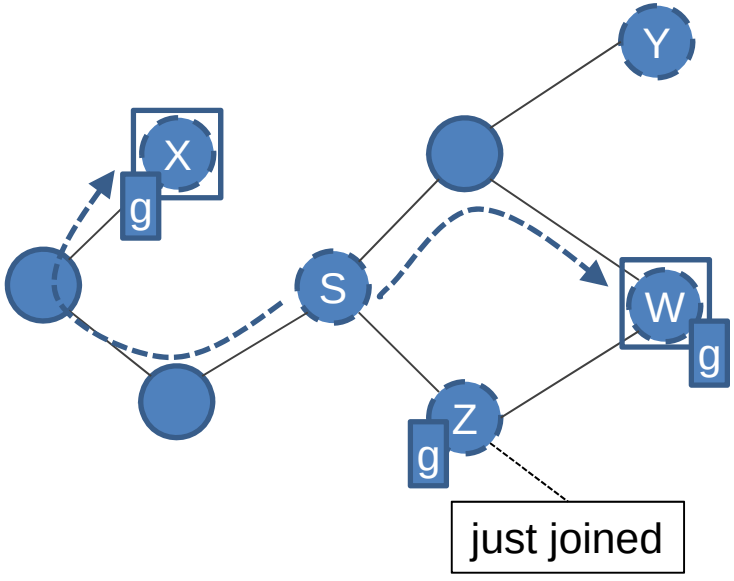
semantically-enabled node

node receiving the packet

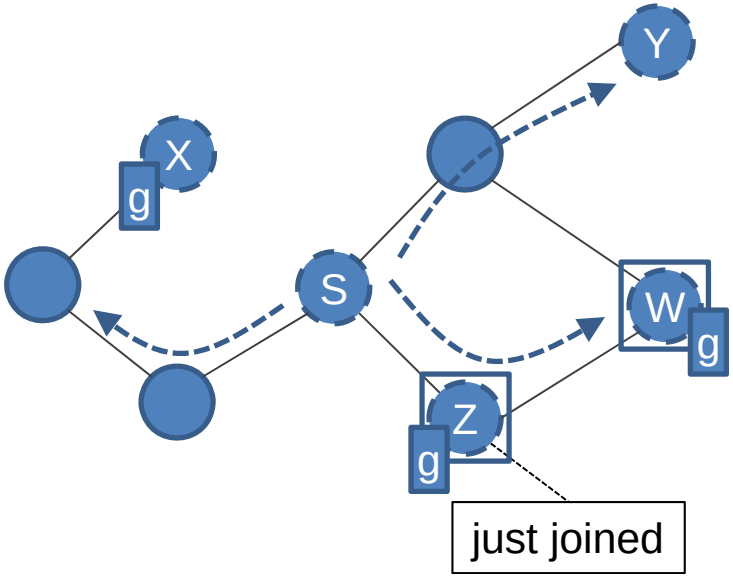
g node interested in "gardening"

packet route

Direct Multicast



Blind Multicast (TTL=2)



Semantic Forward (1)

- Receivers forward multicast packets to ***maximize data delivery coverage***
- Tradeoff between coverage and overhead
  - ***principle of locality***: the closer a receiver node, the greater the interest of the sender that the node receives the packet
    - ***Blind Multicast when close*** to the sender
    - ***Direct Multicast when far*** from the sender
  - ***decreasing overhead*** and ***correctness*** gradient
    - probabilistic approach based on sender-specified parameters

$$BP = SBP \cdot ED^{\#FW}$$

BP: Blind Probability [0,1]
SBP: Starting Blind Probability [0,1]
ED: Exponential Decaying]0,1]

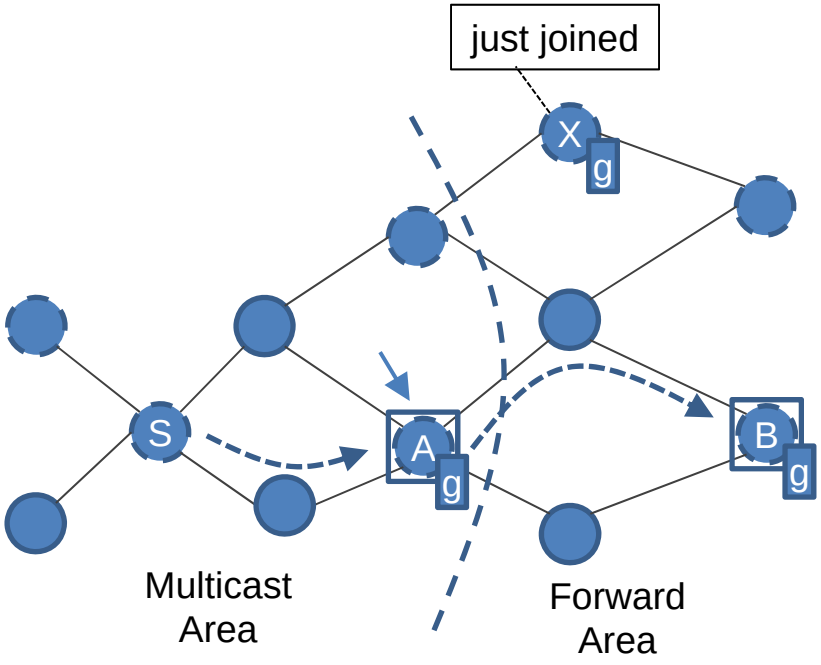
if #FW ≤ MF

FW: Forward
MF: Maximum Forwards

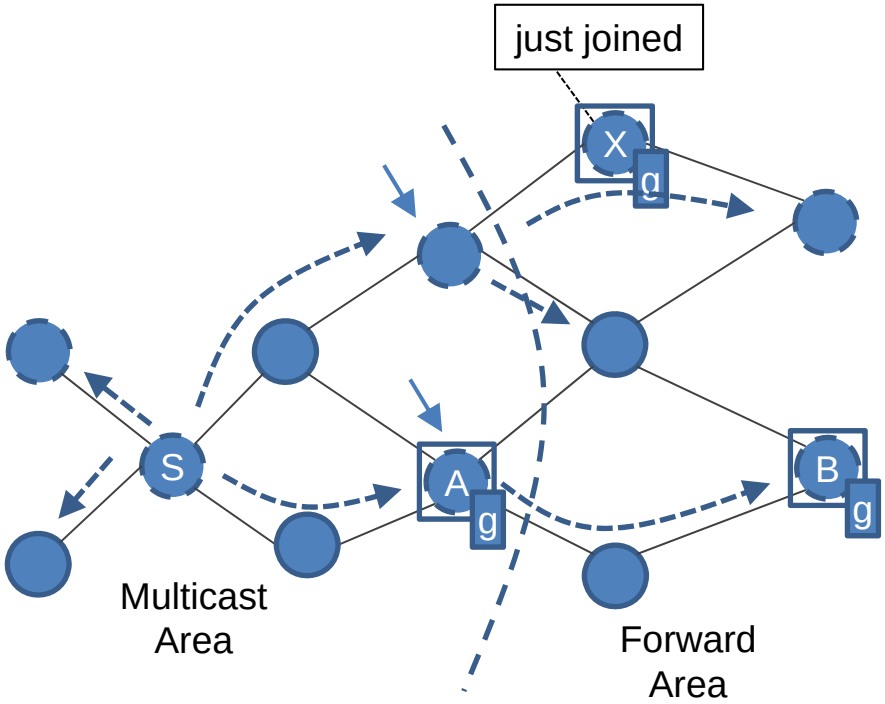
- the greater the SBP, the greater the probability of Blind Multicast
- the lower the ED, the faster our middleware switches from Blind to Direct

Semantic Forward (2)

Only Direct Multicast



Only Blind Multicast
(TTL=2)



forwarding node packet route

Data Representation (1)

- ***Semantic Web mechanisms*** to store and manage data
  - ***user characteristics*** based on Resource Descriptor Framework (RDF)
  - packet ***delivery criteria*** as SPARQL Protocol and RDF Query Language (SPARQL) queries
  - Friend Of A Friend (FOAF) as example ***vocabulary***

- ***ASK*** SPARQL queries to implement Blind and Direct Multicast ***delivery criteria***

```
ASK {
        ?ppd a foaf:PersonalProfileDocument.
        ?ppd foaf:primaryTopic ?user.
        ?ppd foaf:maker ?user.
        ?user foaf:topic_interest Music.
}
```

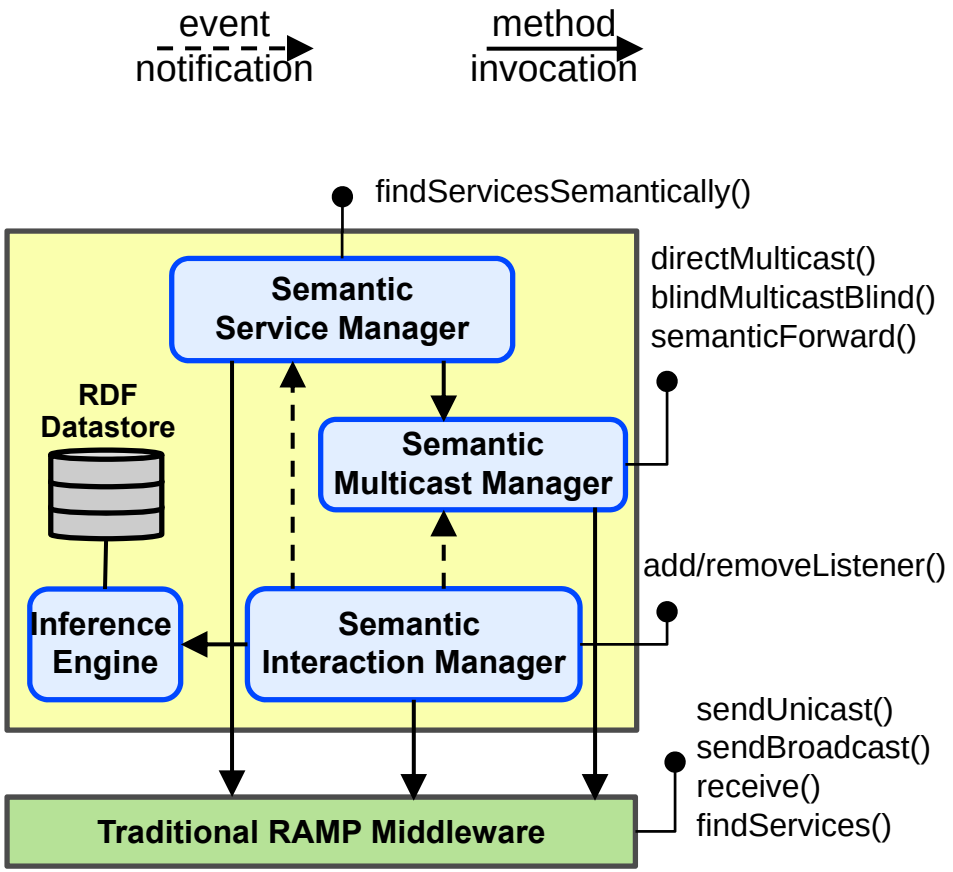
Data Representation (2)

- Different graphs for local/remote nodes
  - local triples tagged to specify **visibility rules**
    - Known People (KP)
    - KP together with people known by known people (KP+)
    - public
  - two additional Ontology Web Language (OWL) classes
    - KnownPerson and KnownByKnownPerson
- ***Construct*** SPARQL query to retrieve a ***personalized view*** of the local node, e.g., Privacy Filter

```
CONSTRUCT {
    mailto:alice@example.org ?prop ?obj.
    WHERE {
        <mailto:alice@example.org> ?prop ?obj.
        ?privacyRule ramp:onPerson <mailto:alice@example.org>.
        ?privacyRule ramp:onProperty ?prop.
        ?privacyRule ramp:permittedRole ?class.
        <mailto:bob@example.org> a ?class.
    }
}
```

Prototype Architecture

- ***RDF Datastore*** and ***Inference Engine***
  - manage FOAF documents
  - based on Jena
- ***Semantic Interaction Manager***
  - periodically sends public information via RAMP broadcast
  - receive remote multicast packets and propagate them to applications
- ***Semantic Multicast Manager***
  - supports Direct/Blind Multicast and Semantic Forward
- ***Semantic Service Manager***
  - remote service discovery based on Direct and Blind Multicast



Prototype Performance

- Comparison of ***traditional and semantic-based service discovery***
- Testbed: three IEEE 802.11b ad-hoc links and different RDF datasets
- Semantic-based service discovery ***increases latency only linearly*** in relation to both path length and dataset size
- Semantic Web techniques suitable for the target spontaneous networking scenario
- Reduced packet exchange***: knowledge of remote nodes exploited to filter out*** useless discovery traffic (see details in the paper)

		Path length (#hops)		
Service Discovery		1	2	3
Traditional		0.03 s	0.06 s	0.07 s
Semantically enabled	small dataset	0.06 s	0.13 s	0.18 s
	medium dataset	0.06 s	0.14 s	0.21 s
	large dataset	0.07 s	0.15 s	0.28 s

small/medium/large datasets contain 10/250/750 `foaf:knows` and 10/50/200 `foaf:topic_interest` relationships respectively

Conclusions and Ongoing Work

- RAMP supports ***multi-hop service-oriented*** communication in ***heterogeneous spontaneous networks***
  - easy-to-use API for service development by non-expert programmers
- Novel 3-layer multicast model
  - ***non-mutually exclusive mechanisms*** suitable for different scenarios
- Semantic multicast to ***decouple senders and receivers***
  - Blind and Direct Multicast with ***different correctness/efficiency tradeoffs***
  - ***node collaboration*** not only to dispatch packets, but also to (partially) ***disclose users' characteristics***
- Ongoing work
  - peer-to-peer ***interconnection of isolated spontaneous networks*** based on users' ***social relationships***
  - automatic ***identification of existing relationships*** from social network apps (Facebook, Twitter) based on standard APIs, whenever possible

Any Questions?

Thanks for your attention ☺

Any questions?

Prototype code and implementation insights

- http://lia.deis.unibo.it/research/RAMP/
- http://lia.deis.unibo.it/Staff/PaoloBellavista/