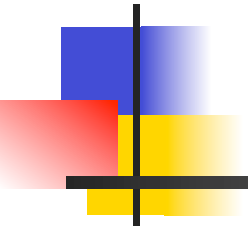
Nomadic Communications

UNIVERSITÀ DEGLI STUDI DI TRENTO



Wireless Mesh Networks

Renato Lo Cigno
LoCigno@disi.unitn.it - Tel: 2026

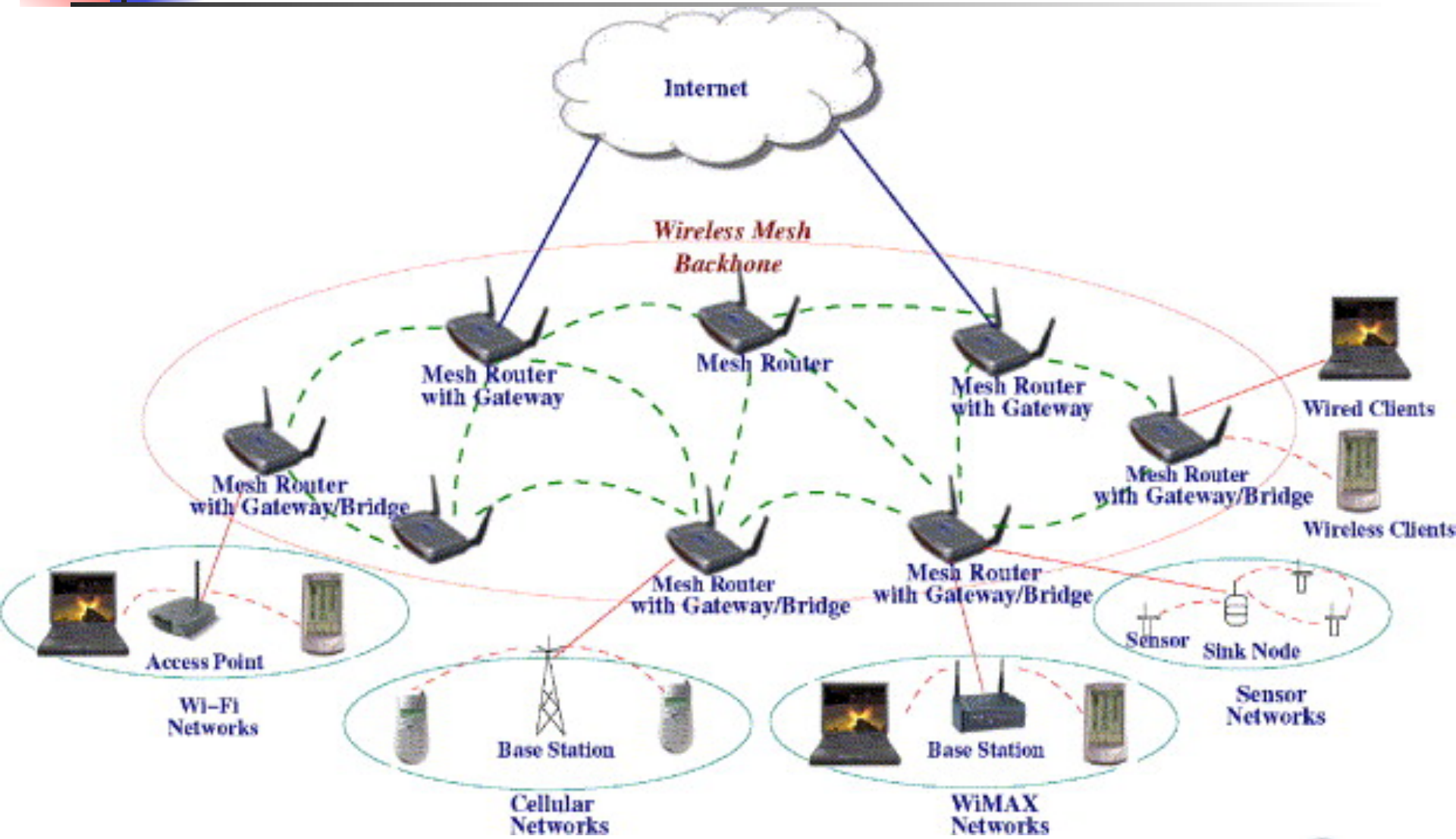
Home Page: http://isi.unitn.it/locigno/index.php/teaching-duties/nomadic-communications

Copyright

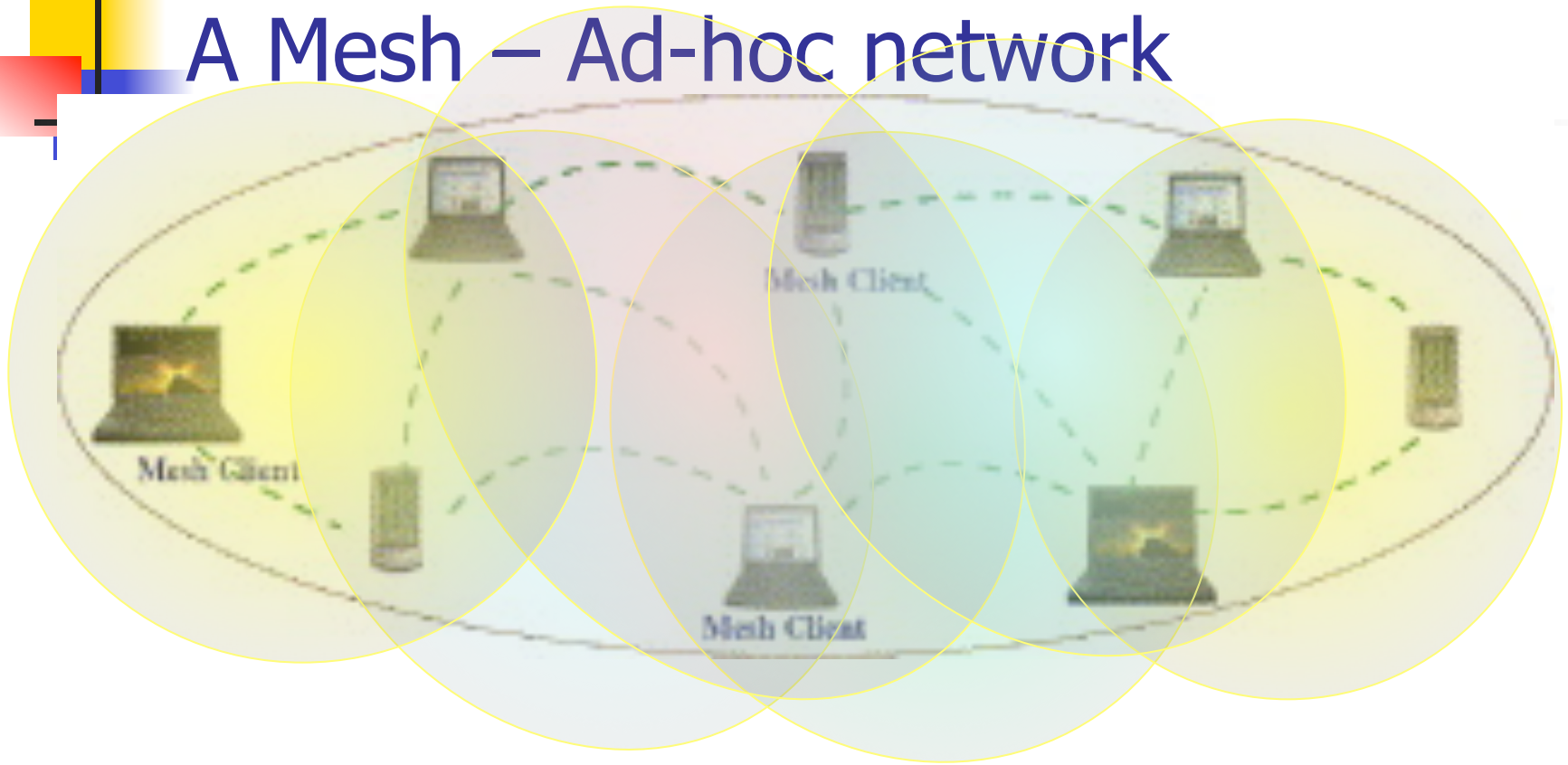
Ad-Hoc and WMN

- Ad-Hoc network
  - non permanent
  - general purpose or specific (sensors)
  - single or multi-hop, normally mobile
  - may require routing (see AODV and OLSR)
  - may have "special purposes" (military, vehicular)
- Wireless Mesh Networks (WMN)
  - more structured than Ad-Hoc
  - may be hierarchical
  - semi-permanent, some nodes are fixed
  - requires routing

WMN: a general view

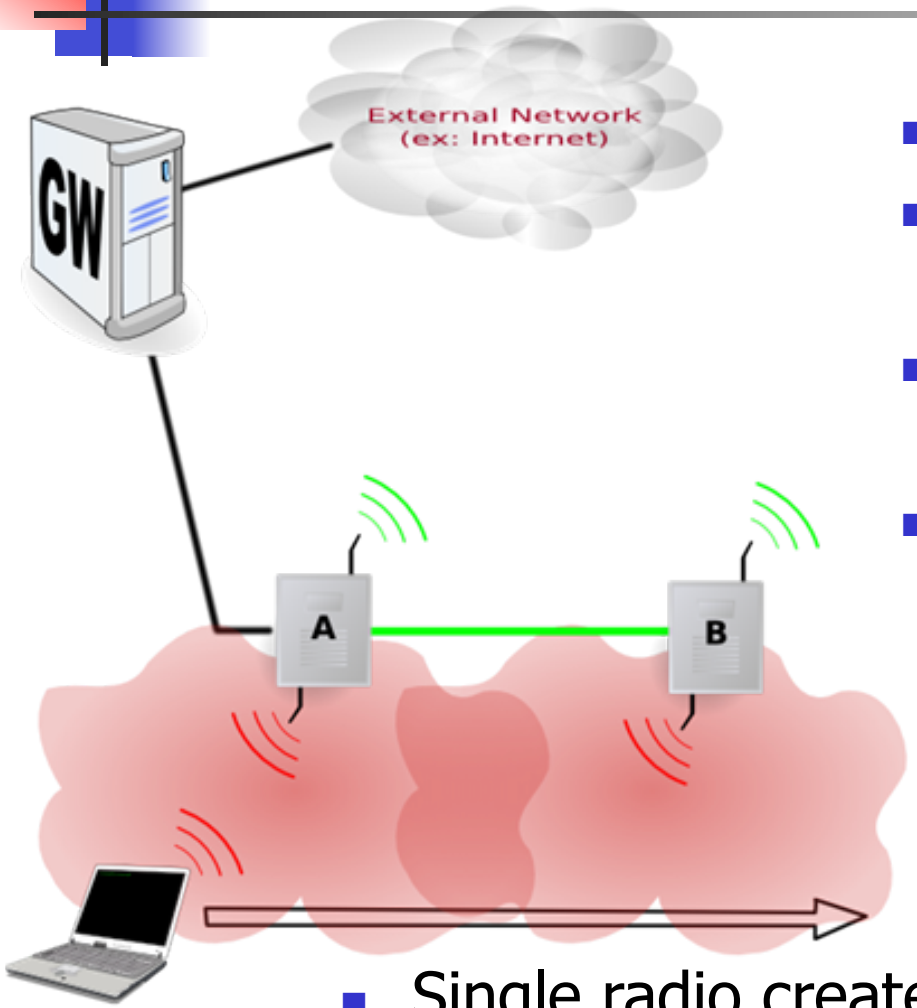


A Mesh – Ad-hoc network



- Ad-Hoc can be meshed
  - non single broadcast channel
  - multi-hop require routing

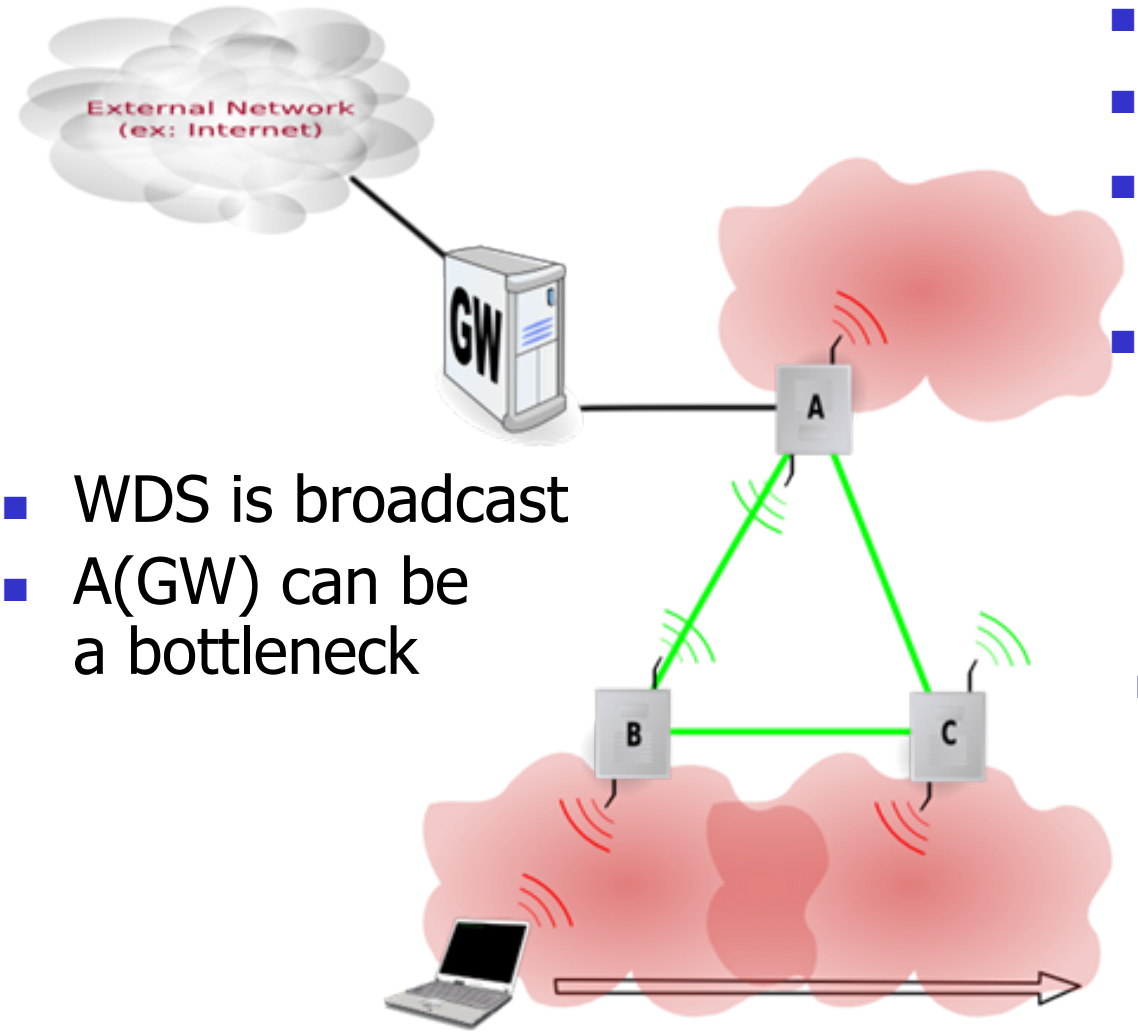
Mesh: Basic scenarios (1)



- Extended WLAN access
- Simple configuration
  - no routing
- Simple 802.11 handover support
- Double radio guarantees good performance

- Single radio creates resource conflicts
  - 3 BSS on the same channel
  - suitable for low-cost low-performance

Mesh: Basic scenarios (2)

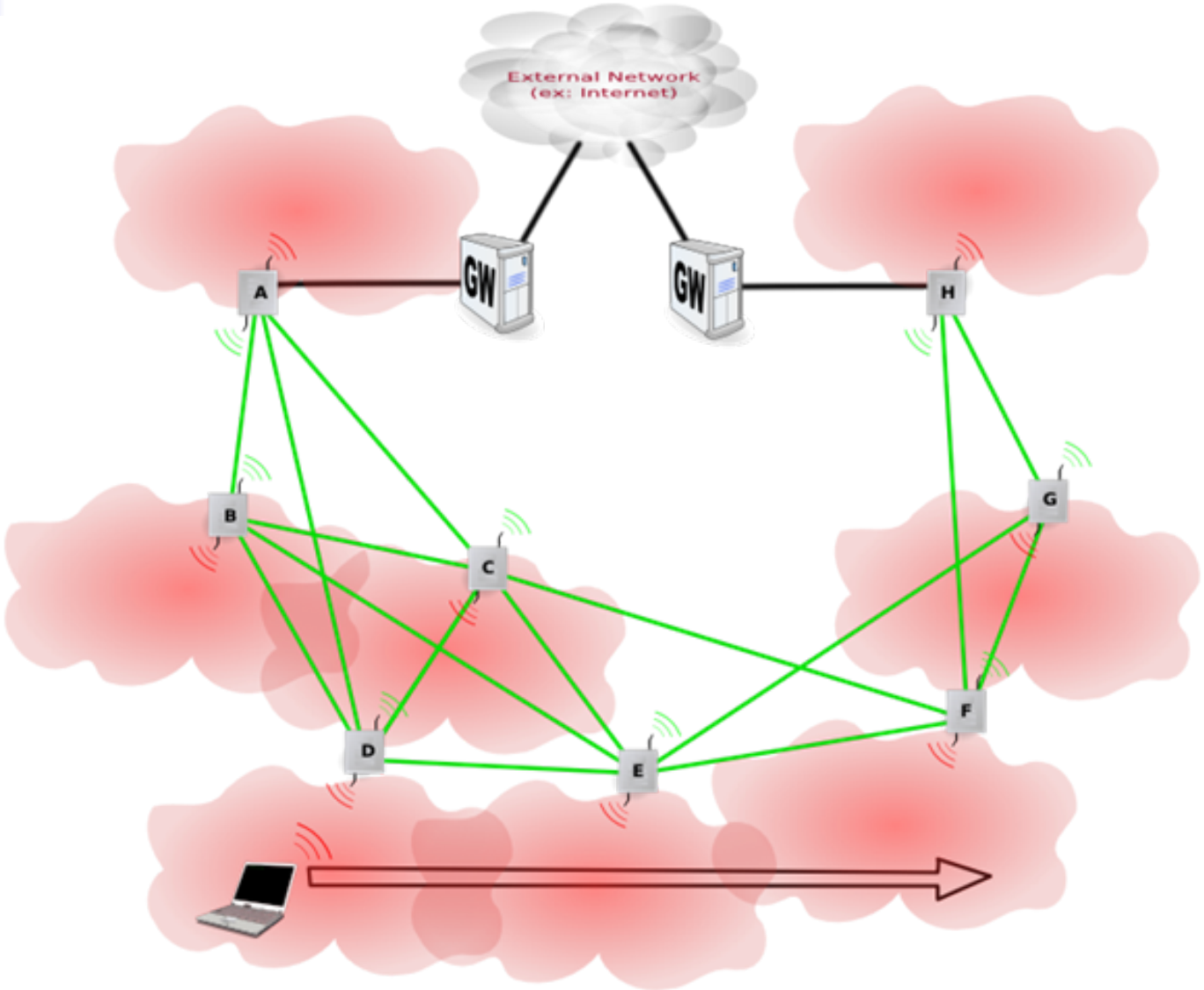


- Extended WLAN access
- Routing required
- Simple 802.11 handover support
- Double radio guarantees good performance

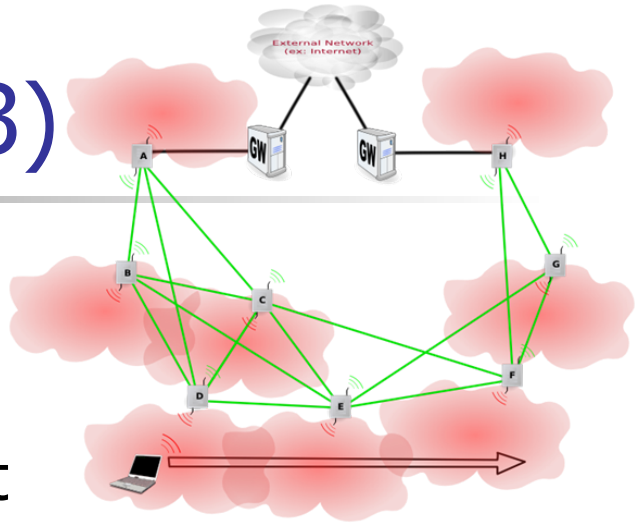
- WDS is broadcast
- A(GW) can be a bottleneck

- Single radio creates serious resource conflicts
  - n+1 BSS on the same channel

Mesh: Basic scenarios (3)



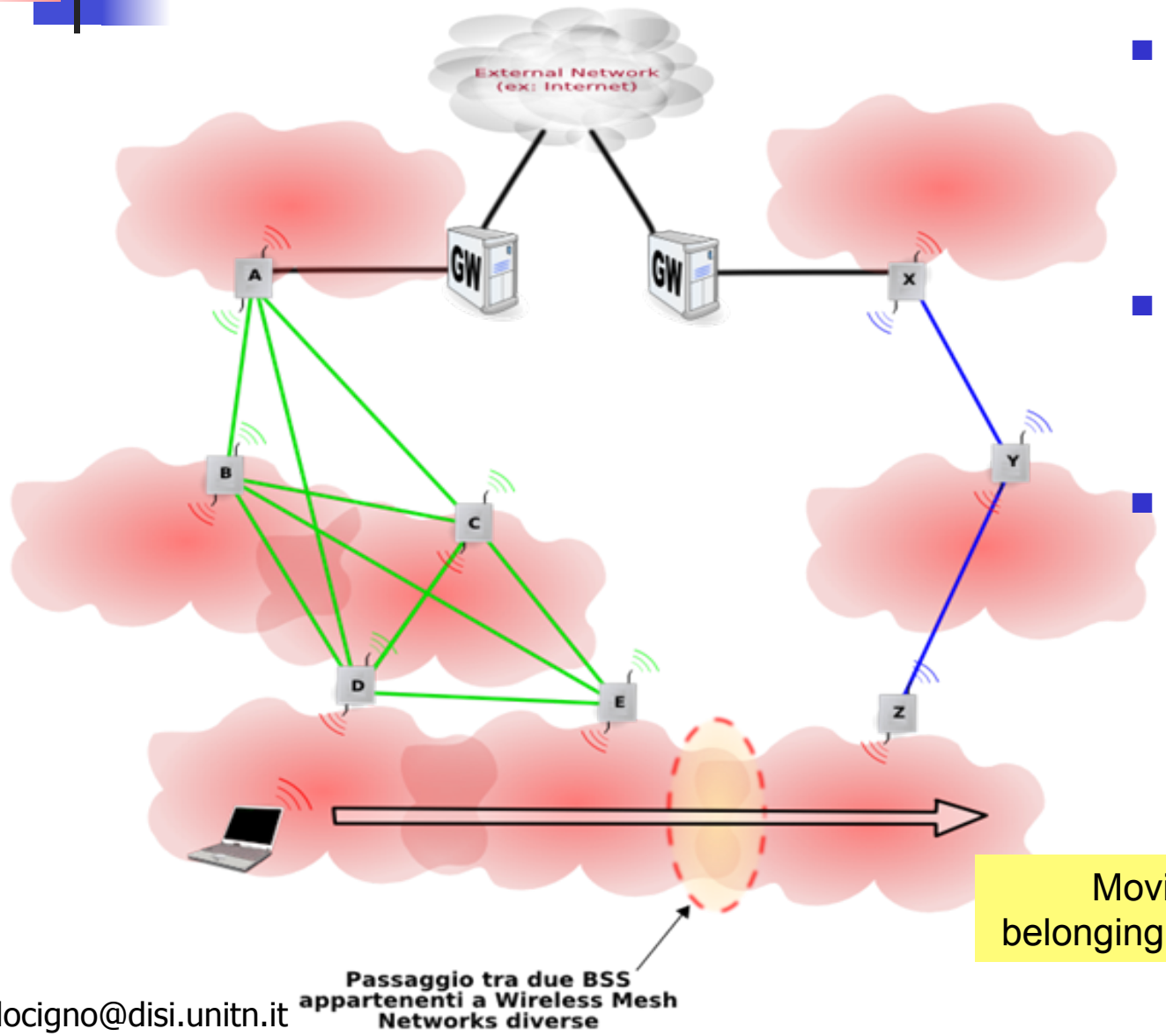
Mesh: Basic scenarios (3)



- Extended WLAN access
- Basic infrastructuring
- Single radio operation very difficult

- Multiple external gateways
  - sophisticated, flow-based routing
- Non standard handover support
  - flow based routing requires exporting the context
  - address management require coordination
- WDS may be multi-hop
  - How many channels?
- Point-to-point and broadcast channels in WDS

Mesh: Basic scenarios (3)



- Address management (DHCP) is a problem
- Flow-based routing may be impossible
- Joining/splitting of partitions is an open issue

Moving between BSS belonging to different Mesh/WDS

Mesh – Ad-Hoc: AODV

Ad-hoc On-demand Distance Vector routing – rfc3561

- DV (see RIP) protocol for next-hop based routing
- On-Demand: maintains routes only for nodes that are communicating
- Must build routes when requested
- Route Request (RREQ) are flooded through the network
- Nodes set-up reverse path pointers to the source
  - **AODV assumes symmetric links**

Mesh – Ad-Hoc: AODV

- The intended receiver sends back a Route Reply (RR)
- RR follow the reverse path set-up by intermediate nodes (unicast) establishing a shortest path route memorized by intermediate nodes
- Paths expire if not used
  - protocol & transmission overhead
  - guarantee of stability in dynamic, non reliable networks
- Usual DV problems
  - count to infinity, slow convergence, ...
  - in a dynamic environment may be too much → throughput going to zero

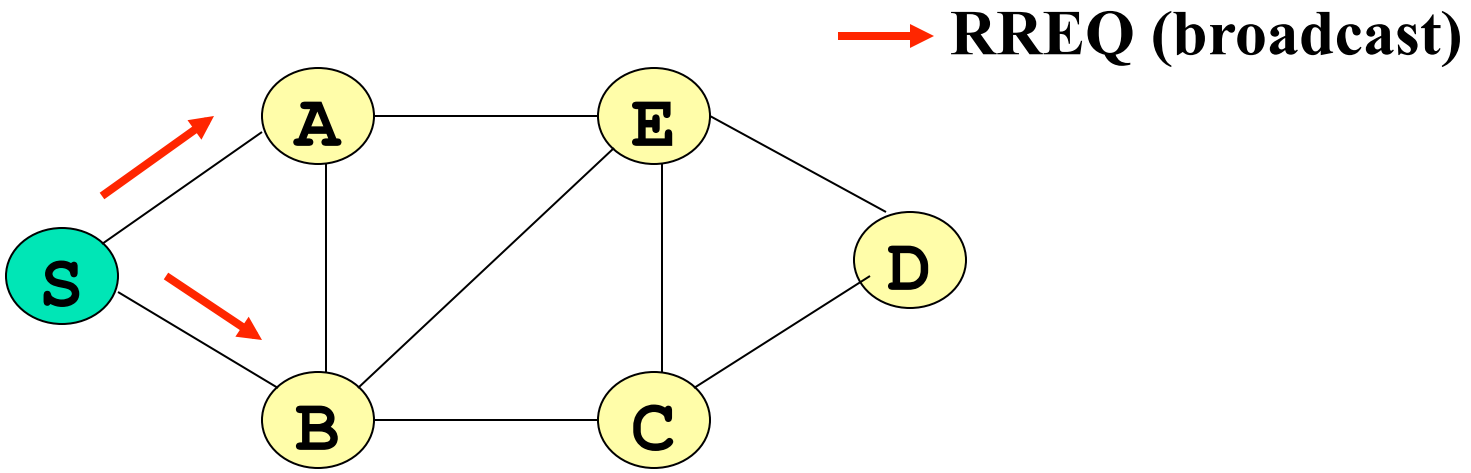
AODV Loop Freedom

- Destination sequence numbers to order routing events in time
- Ordering among *<seqno, hop count>* tuples at different nodes on a path
  - higher seqno has precedence
  - if same seqno, lower hop count has precedence
- The final selection will be the shortest path (w.r.t. some metric, not necessarily hop-count)

Mesh – Ad-Hoc: AODV

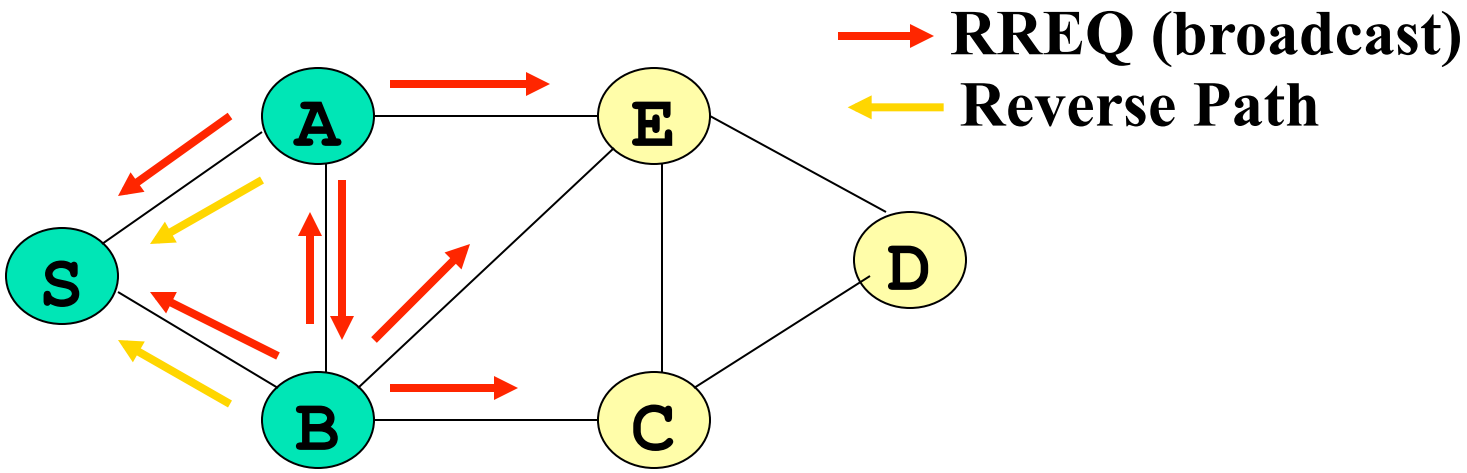
- Next-hop based (other proposals are based on source routing)
- “Flat” protocol: all nodes are equal
- Can manage only one route per s-d pair
  - can be inefficient in presence of highly variable link quality and persistence
- Good for sporadic communications
- Bad for high mobility
  - slow convergence
  - difficulty in understanding topology changes.

Basic AODV Route Discovery



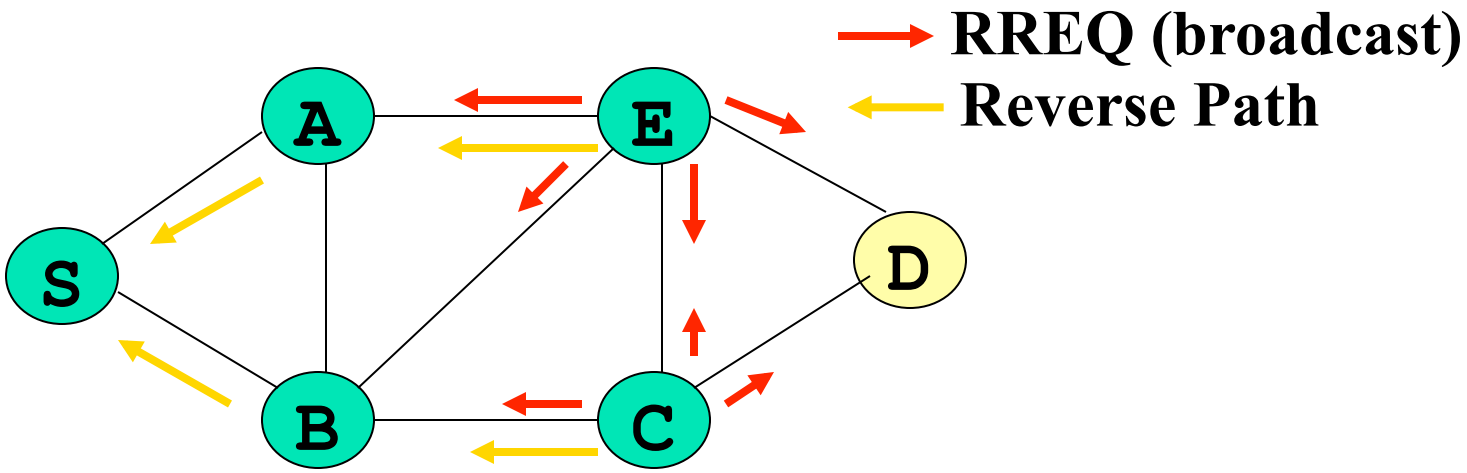
- When a route is needed, source floods a route request for the destination.

Basic AODV Route Discovery



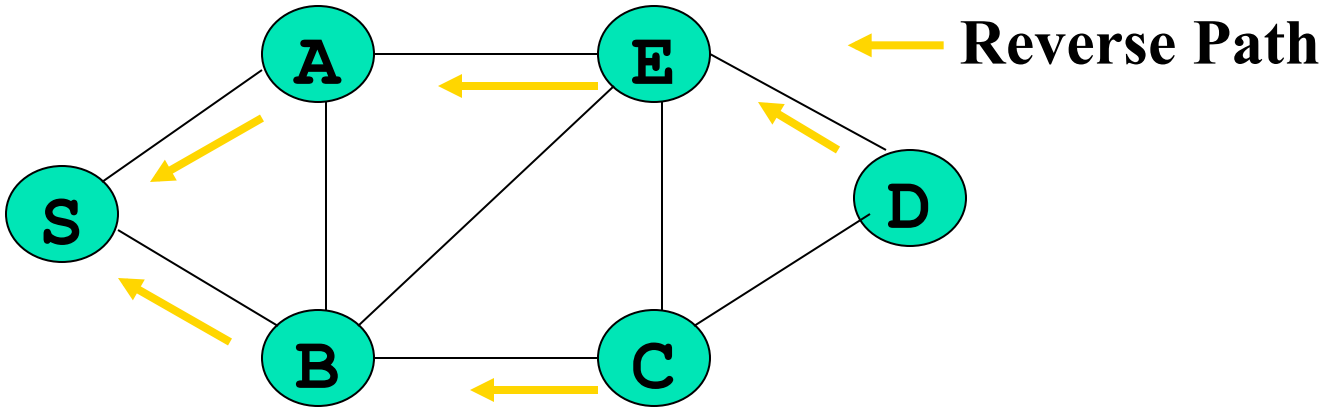
- Reverse path is formed when a node hears a non-duplicate route request.
- Each node forwards the request at most once (pure flooding).

Basic AODV Route Discovery



- Reverse path is formed when a node hears a non-duplicate route request.
- Each node forwards the request at most once (pure flooding).

Basic AODV Route Discovery



- Observation: Duplicate RREQ copies completely ignored. Therefore, potentially useful alternate reverse path info lost.

Mesh – Ad-Hoc: OLSR

Optimized Link-State Routing Protocol (rfc3626)

- Proactive, link-state routing protocol
- Based on the notion of MultiPoint Relay (MPR)
- Three main components:
  - Neighbor Sensing mechanism
  - MPR Flooding mechanism
  - topology Discovery (diffusion) mechanism.

- Auxilary features of OLSR:
  - network association - connecting OLSR to other networks

Mesh – Ad-Hoc: OLSR

Basic neighbor sensing:

- periodic exchange of HELLO messages;
- HELLO messages list neighbors + "neighbor quality“
  - HEARD - link may be asymmetric
  - SYM - link is confirmed to be symmetric
  - MPR - link is confirmed to be symmetric AND neighbor selected as MPR
- Providing:
  - topology information up to two hops
  - MPR selector information notification

Mesh – Ad-Hoc: OLSR

- Each node selects from among its neighbors an MPR set such that
  - an emitted flooding message, relayed by the MPR nodes, can be received by all nodes in the 2-hop neighborhood

- Goals:
  - reduce flooding overhead (select minimal sets)
  - provide optimal flooding distances

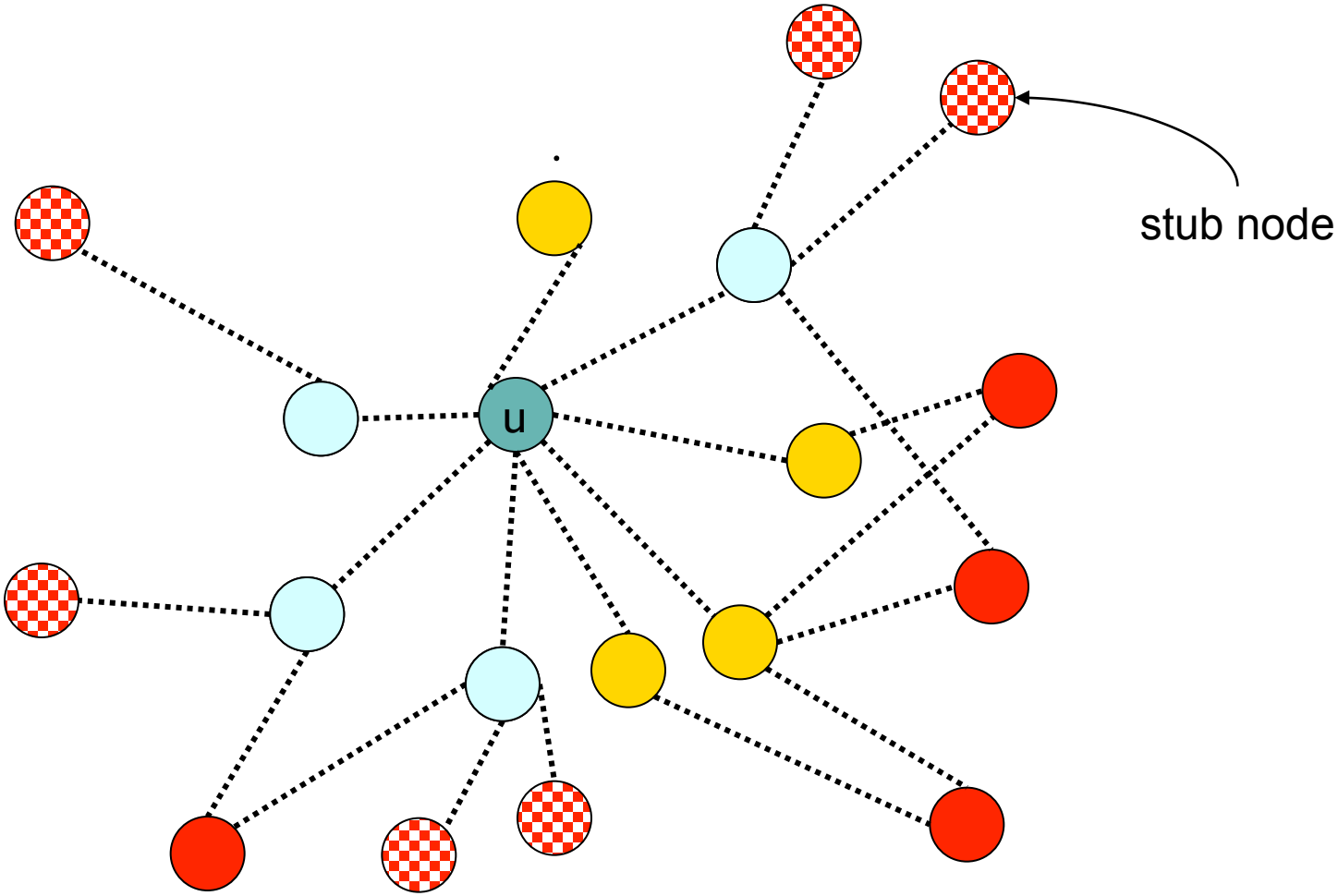
Mesh – Ad-Hoc: OLSR

- Exchanges topology information with other nodes of the network regularly
- MPRs announce their status periodically in control messages
- In route calculation, the MPRs are used to form the route from a given node to any destination in the network
- Uses MPRs to facilitate efficient flooding of control messages
- The presence of a 2-tiear topology (MPRs are sort of supernodes) makes it complex and prone to failures

MPR selection algorithm

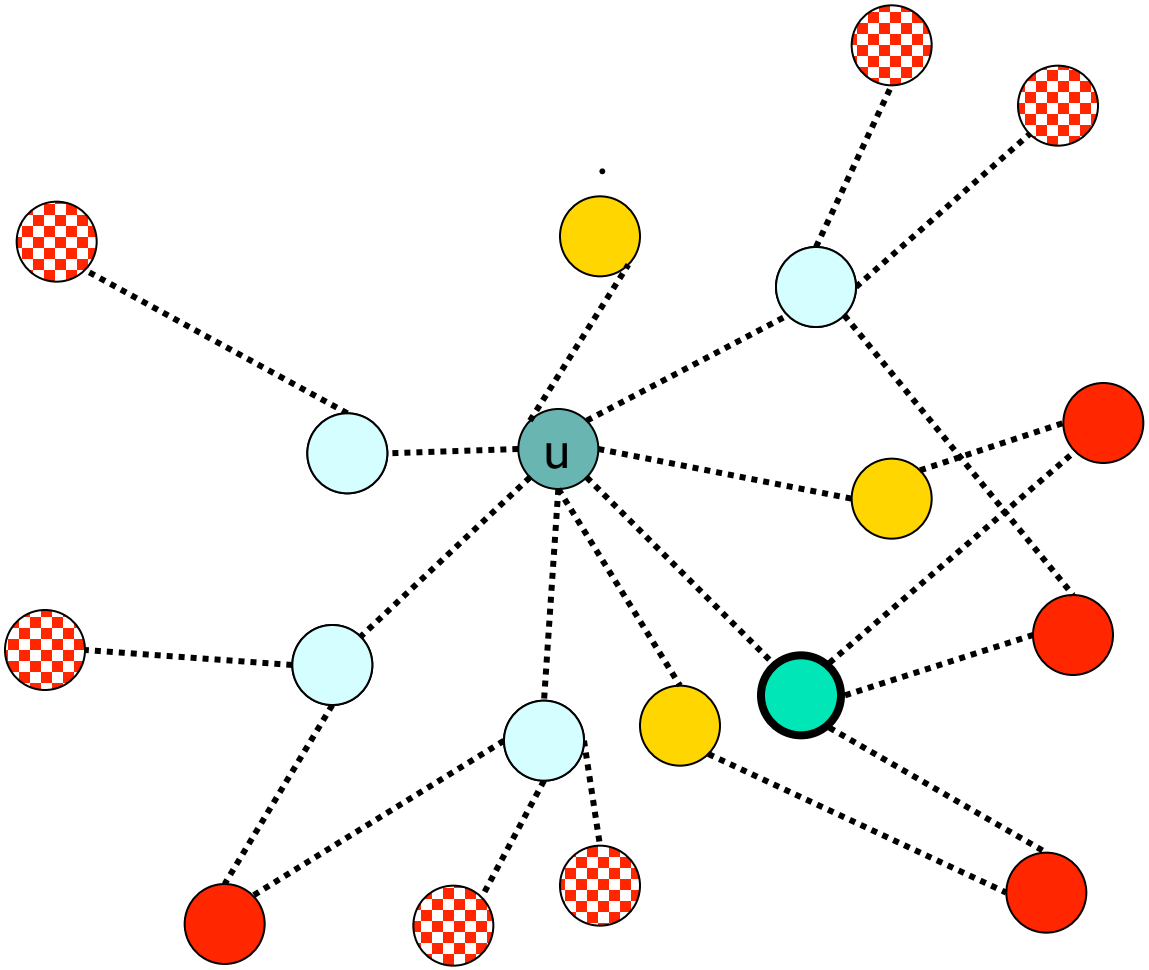
- Each node **u** has to select its set of MPRs.
- Goal : select in the 1-neighborhood of **u,** N1(**u**), a set of nodes as small as possible which covers the whole 2-neighborhood of **u**, N2(**u**)
- Done in two steps:
  - Step 1: Select nodes of N1(**u**) which cover stub nodes of N2(**u**)
    - stub nodes are those that are connected to one N1(**u**) node only
  - Step 2: Select among the nodes of N1(**u**) not selected at the first step, the node which covers the highest number of nodes in N2(**u**) not yet connected
    - Repeat Step 2 until all N2(**u**) is reached

MPR selection step 1



Select nodes (light blue) in N1(u) which cover stub nodes of N2(u)

MPR selection step 2



Select the node in N1(u) which cover the largest number of non-stub nodes in N2(u)

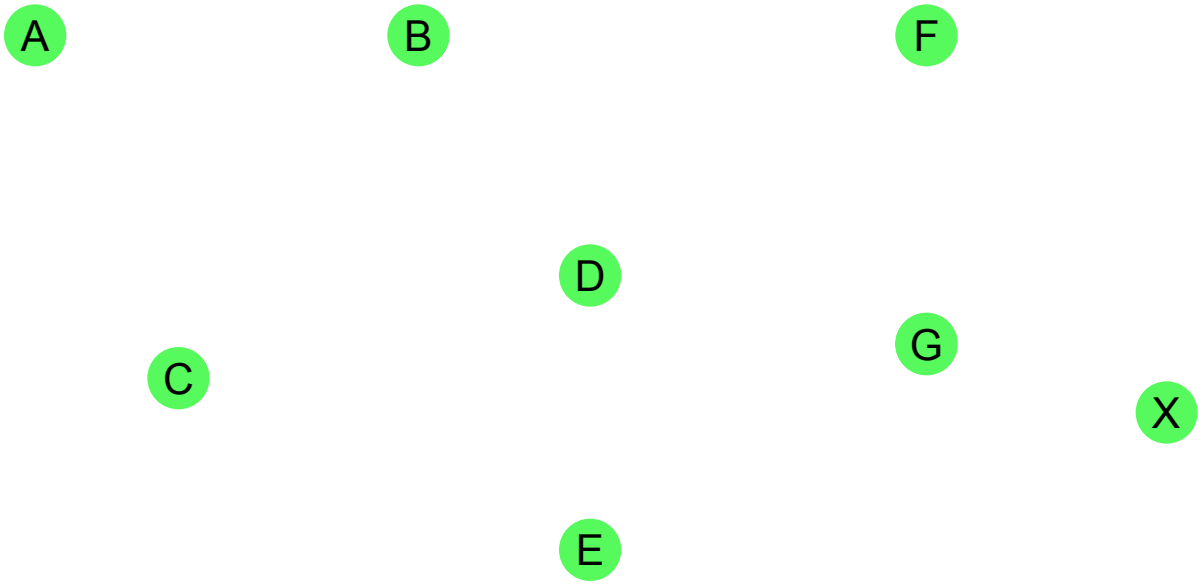
BATMAN

- Better Approach To Mesh Ad-hoc Networking
- A DV protocol using Link Qualities
- Based on periodic Broadcast of “Originator Messages”: OGM
  - Link Quality metric is the number of received OGMs
  - Path Metric is the product of link metric
  - Broadcast is always at minimum PHY rate ... difficult to distinguish high speed paths
- OGM have TTL fields to avoid too long paths
  - TTL must be tailored to the MESH dimension

BATMAN

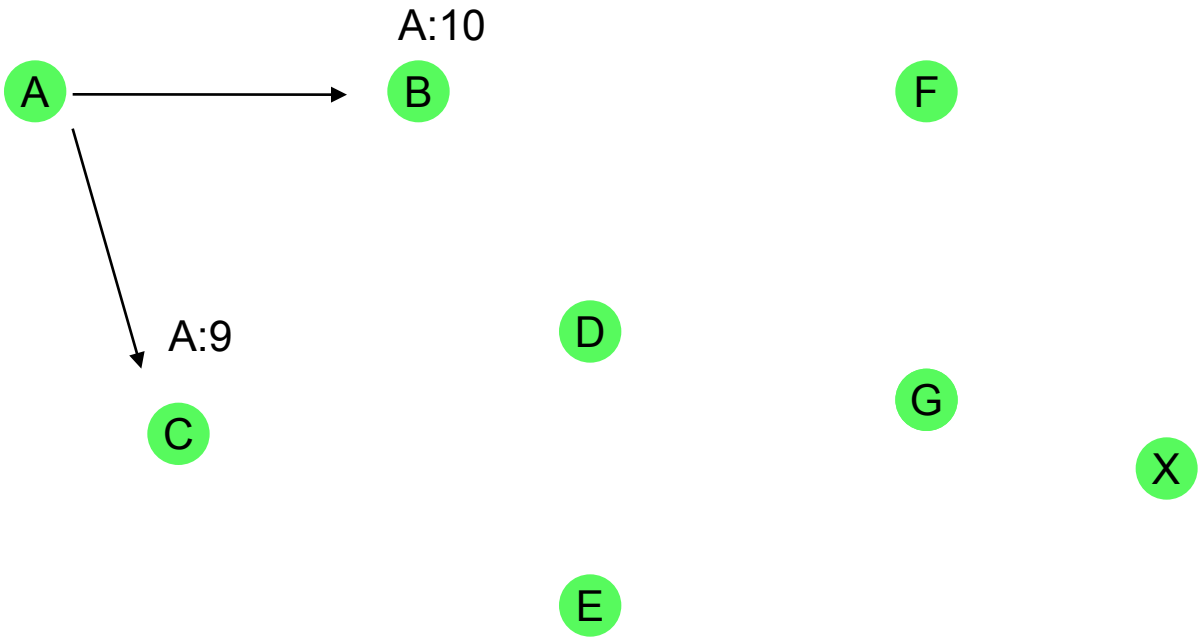
- BATMAN is a level 2.5 routing solution
- Uses MAC addresses to identify stations, avoiding the problem of changing IP addresses to deliver frames
- Not pure layer 2 since it runs in the kernel and is not integrated in NIC cards or drivers
- Relies on Layer 2 info, like link quality
- Send UDP packets and not Layer 2 frames for routing purposes
- BATMAN does not have handover enhancement support
  - Slow convergence makes connection fail
  - We proposed one (already in the distribution) with a colleague of yours 2011/2012 ☺

BATMAN



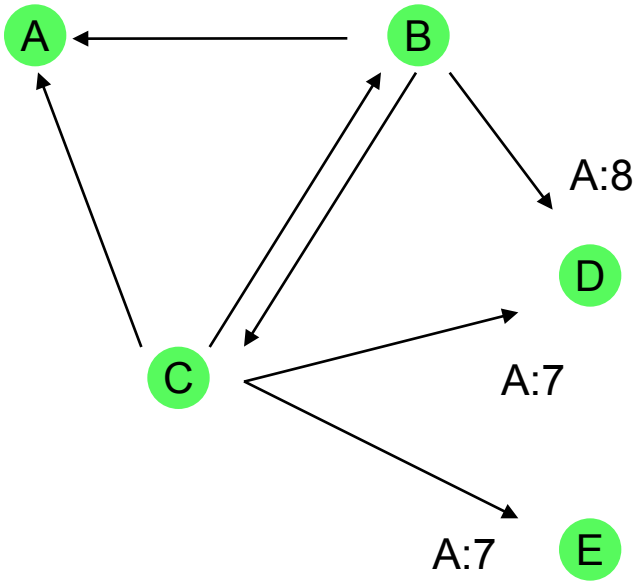
A wants to reach X

BATMAN



- Nodes broadcast originator messages (OGM's) every second
- OGM's are rebroadcast
- Other nodes measure how many OGM's are received in a fixed time window

BATMAN



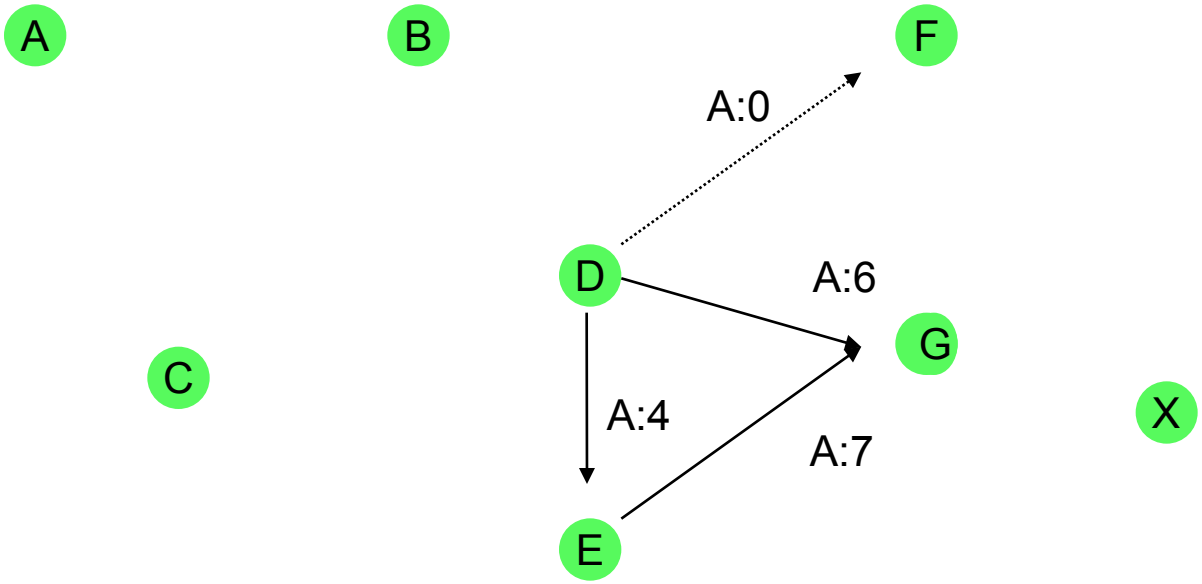
D BATMAN routing table

TO	VIA	Q
A	B	8
A	C	7

D Final routing table

TO	VIA
A	B

BATMAN



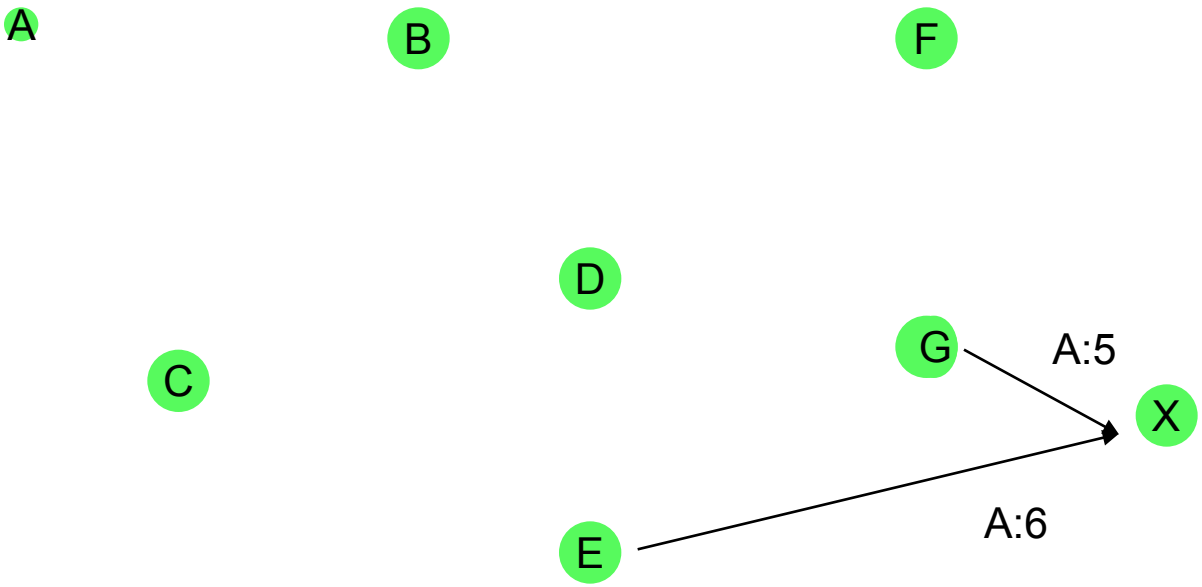
G BATMAN routing table

TO	VIA	Q
A	D	6
A	E	7

G Final routing table

TO	VIA
A	E

BATMAN



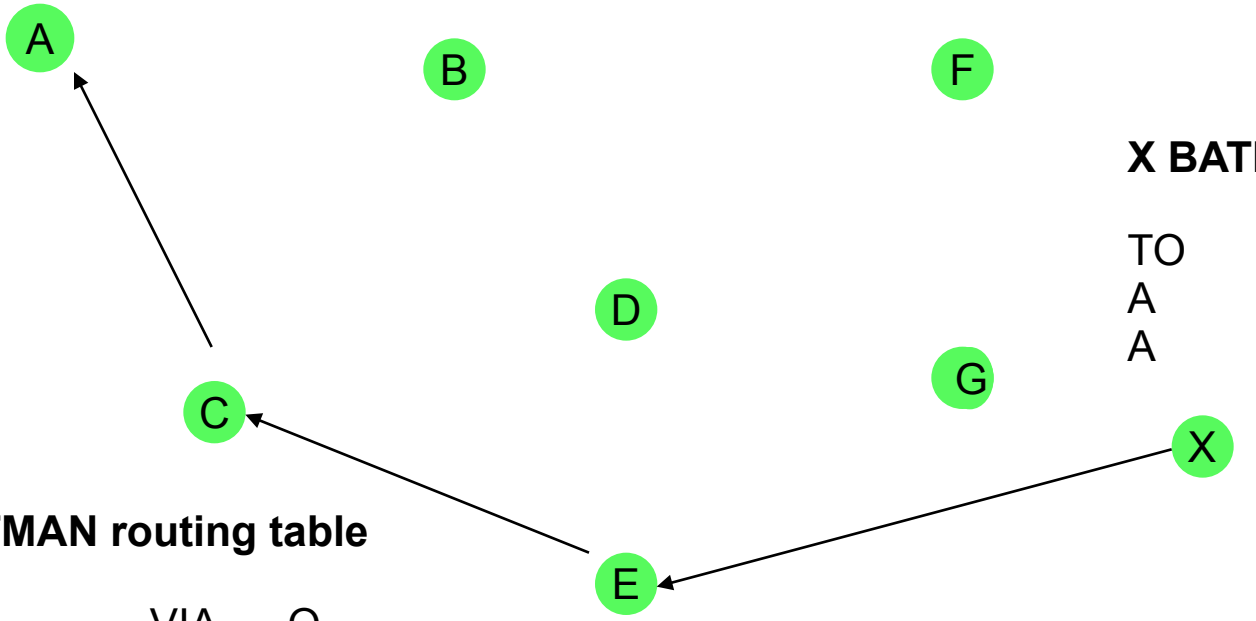
X BATMAN routing table

TO	VIA	Q
A	G	5
A	E	6

X Final routing table

TO	VIA
A	E

BATMAN



X BATMAN routing table

TO	VIA	Q
A	G	5
A	E	6

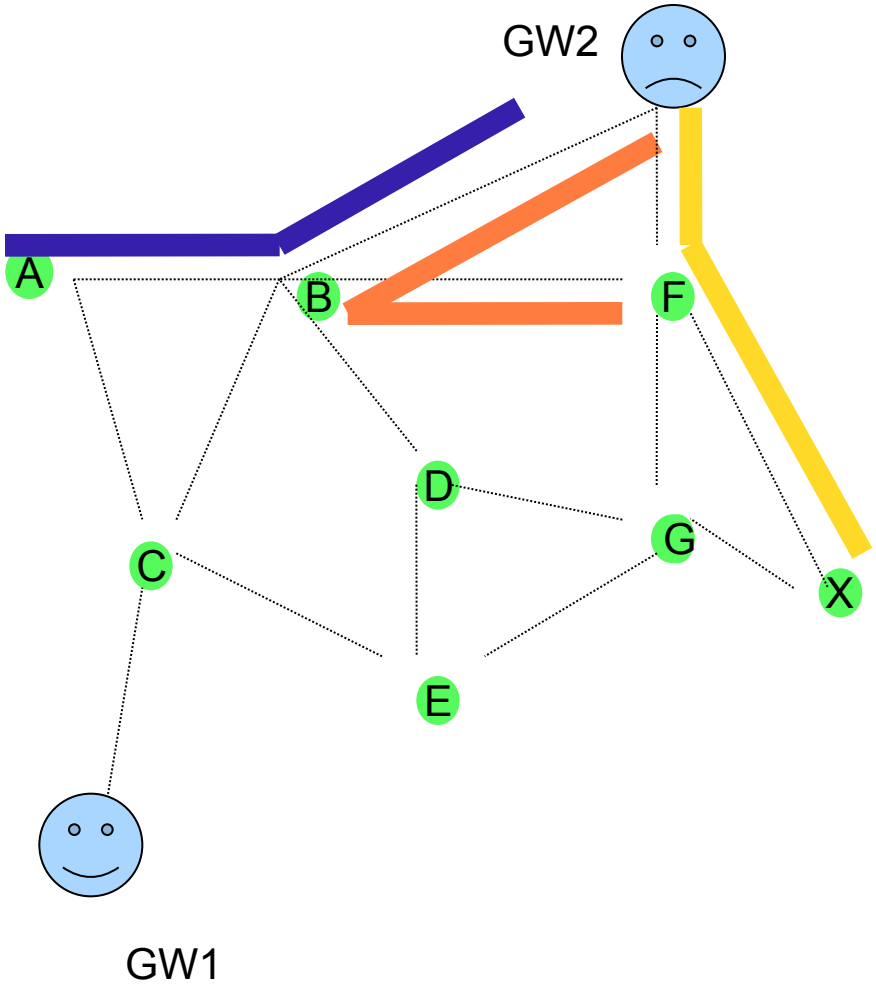
C BATMAN routing table

TO	VIA	Q
A	A	9

E BATMAN routing table

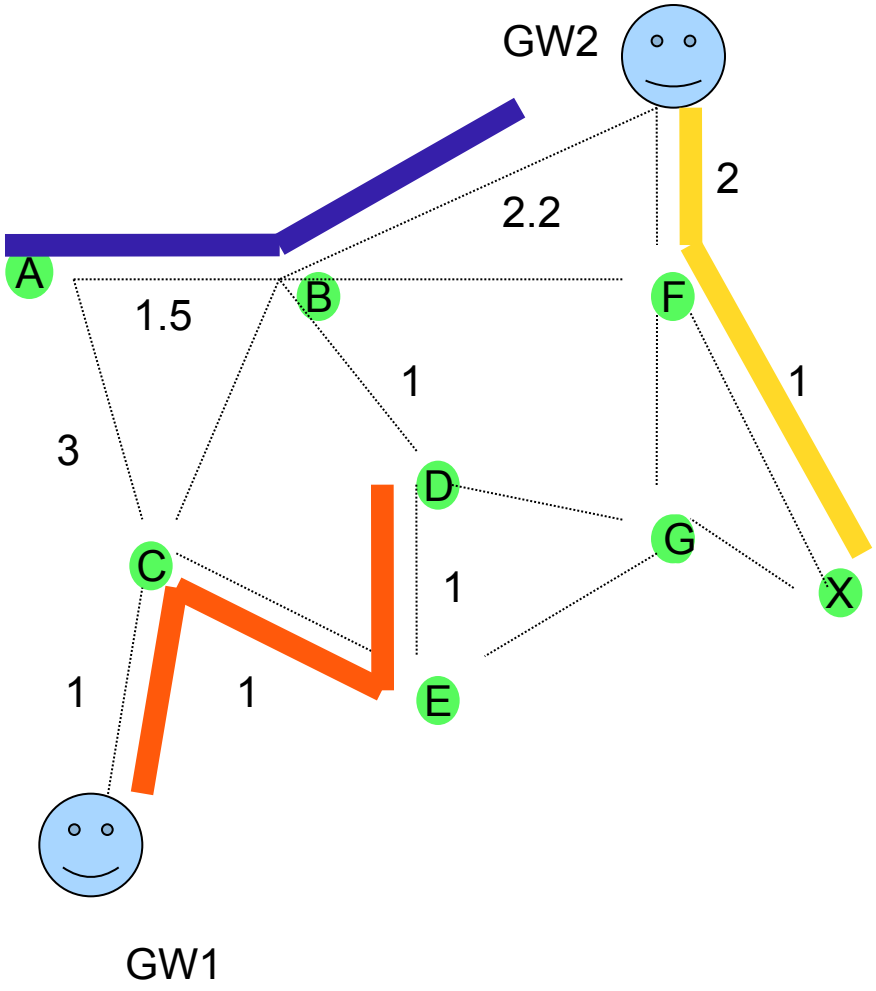
TO	VIA	Q
A	C	7
A	D	4

Current GW selection techniques



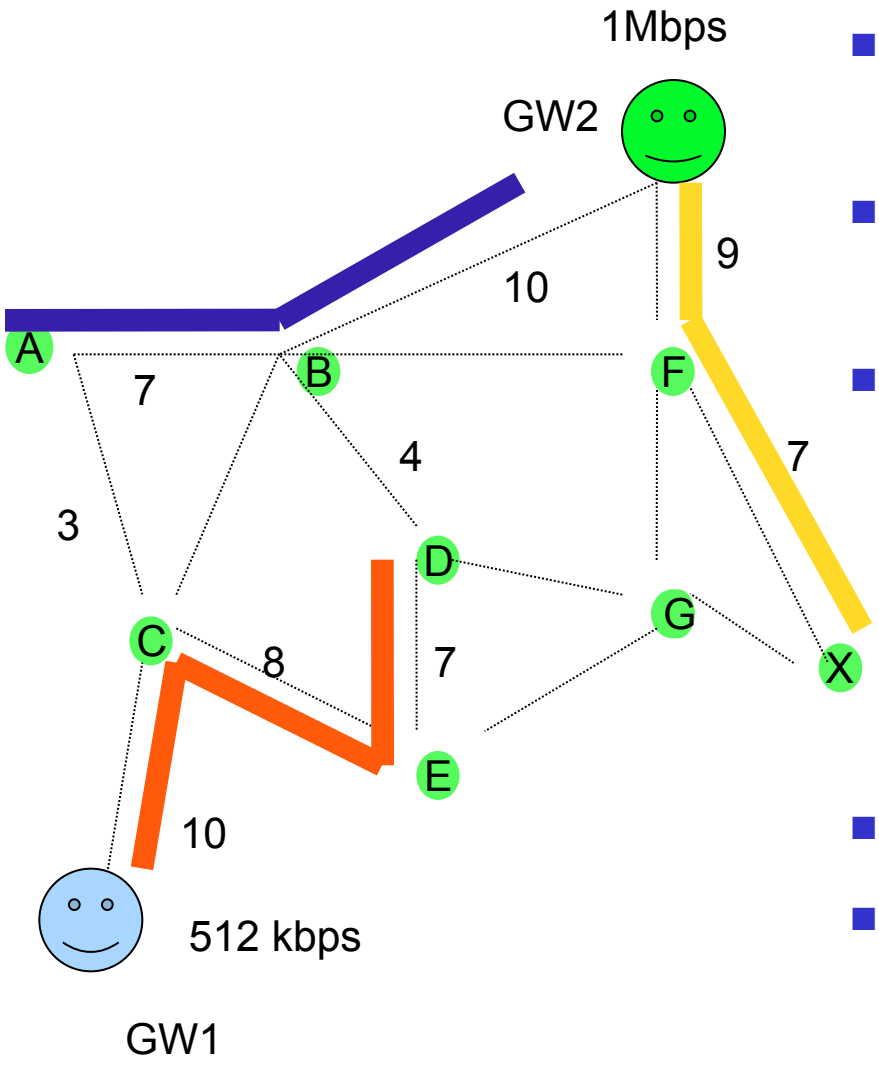
- Minimum hop count to gateways
- Used by routing protocols like AODV
- Creates single over congested gateways

Current GW selection techniques



- Best link quality to GW
- Used by
  - source routing protocols like MIT Srcr
  - Link state protocols like OLSR
- Prevents congested links to GW
- Not global optimum of GW BW usage

Current GW selection techniques



- BATMAN has advanced a little further
- GW can advertise downlink speed
- User can choose GW selection based on
  - GW with best BW
  - Stable GW (need history)
  - GW_{BW} x LQ
- Can't trust advertised GW BW
- Doesn't achieve fairness

BABEL & OTHERS

- Experimental RFC 6126
- Found in many Linux releases
- DV based on IP addresses
  - problems with handovers and mobility
- Loop free, based on ideas similar to BATMAN, AODV, DSDV (Destination Sequenced Distance Vector)
  - Destination Sequenced
- Many more proposal and enhancements to these
  - Material for an entire course, specially if also multicast is taken in the loop

Mesh Networks: 802.11s

- Working group to deliver a standard for 802.11 (& around) base Mesh Networks
- There are drafts and early releases, but not yet a definitely released standard (as of 2010)
- Tries to define a framework to support a Mesh network as a standard extended WLAN with routing that goes beyond the standard minimum spanning tree of 802.1 interconnection

Device Classes in 802.11s

- Mesh Point (MP)
  - a point able to relay messages
- Mesh AP (MAP)
  - a MP able to provide services to STAs
- Mesh Portal (MPP)
  - a MAP connected to a wired LAN
  - normally called a gateway and assumed to access the internet

Routing in 802.11s

- Hybrid Wireless Mesh Protocol (HWMP) - Mandatory
  - AODV derived link-state protocol
  - Based on trees for proaction and efficiency
  - Add on-demand features (like AODV)
- Radio Aware OLSR (RA-OLSR) – Optional
  - Radio aware metrics added to MPRs in OLSR
  - optional fish-eye routing capabilities
  - association and discovery protocols for topology discovery and buildup

Routing in 802.11s

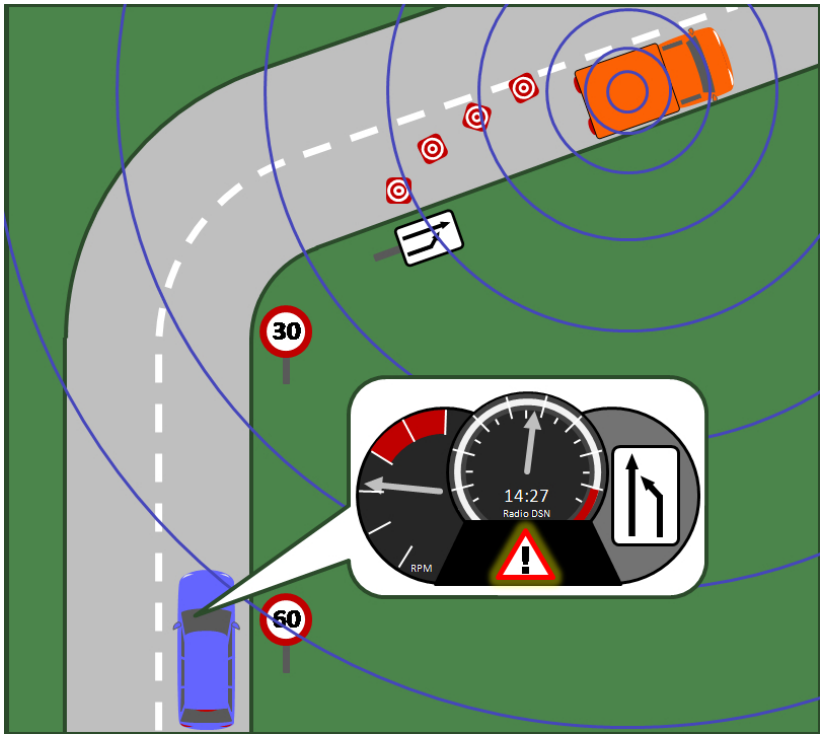
- BATMAN probably supported
- Features for multi-gateway management
- Support for Vehicular networks, where some specialized features are needed
- Use only MAC addresses for routing
- Run directly in the diverse/NIC cards

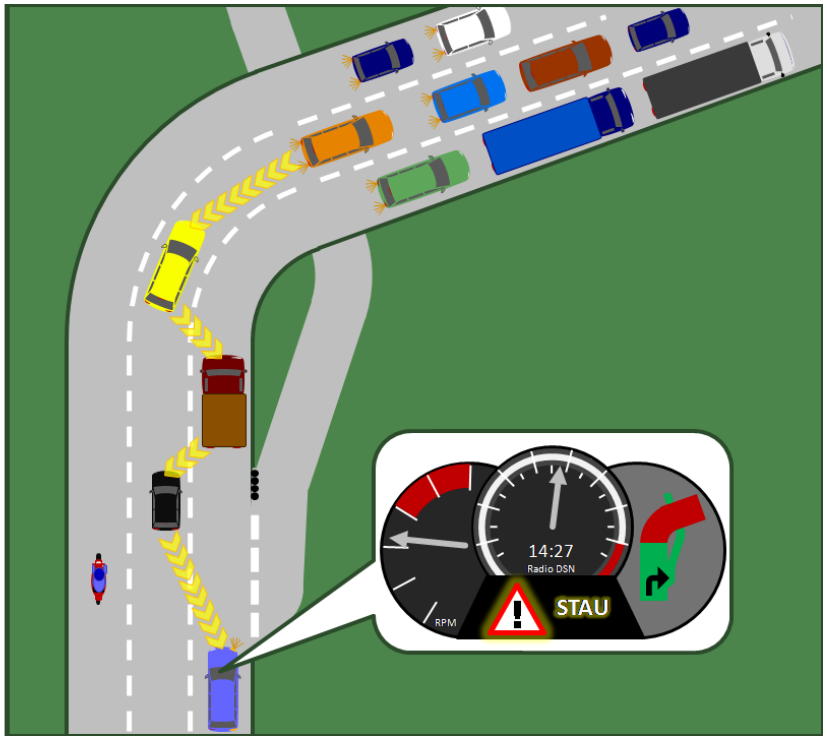
- Integration with the other 802.11 protocols ... which is the real strength!

Vehicular Networks

- **Networking-centric view to Vehicular Ad Hoc Networks**
- Focus is on wireless local area networking techniques for communication between vehicles and between vehicles and roadside units
  - Not on Inter-Vehicle Communications (IVC) based on wide area cellular networks
  - PHY/MAC: 802.11p
    - **a- derived; 5MHz BW, reduced CW, no association**
- We do not look at location techniques

Active safety





[Graphics by S. Labitzke]

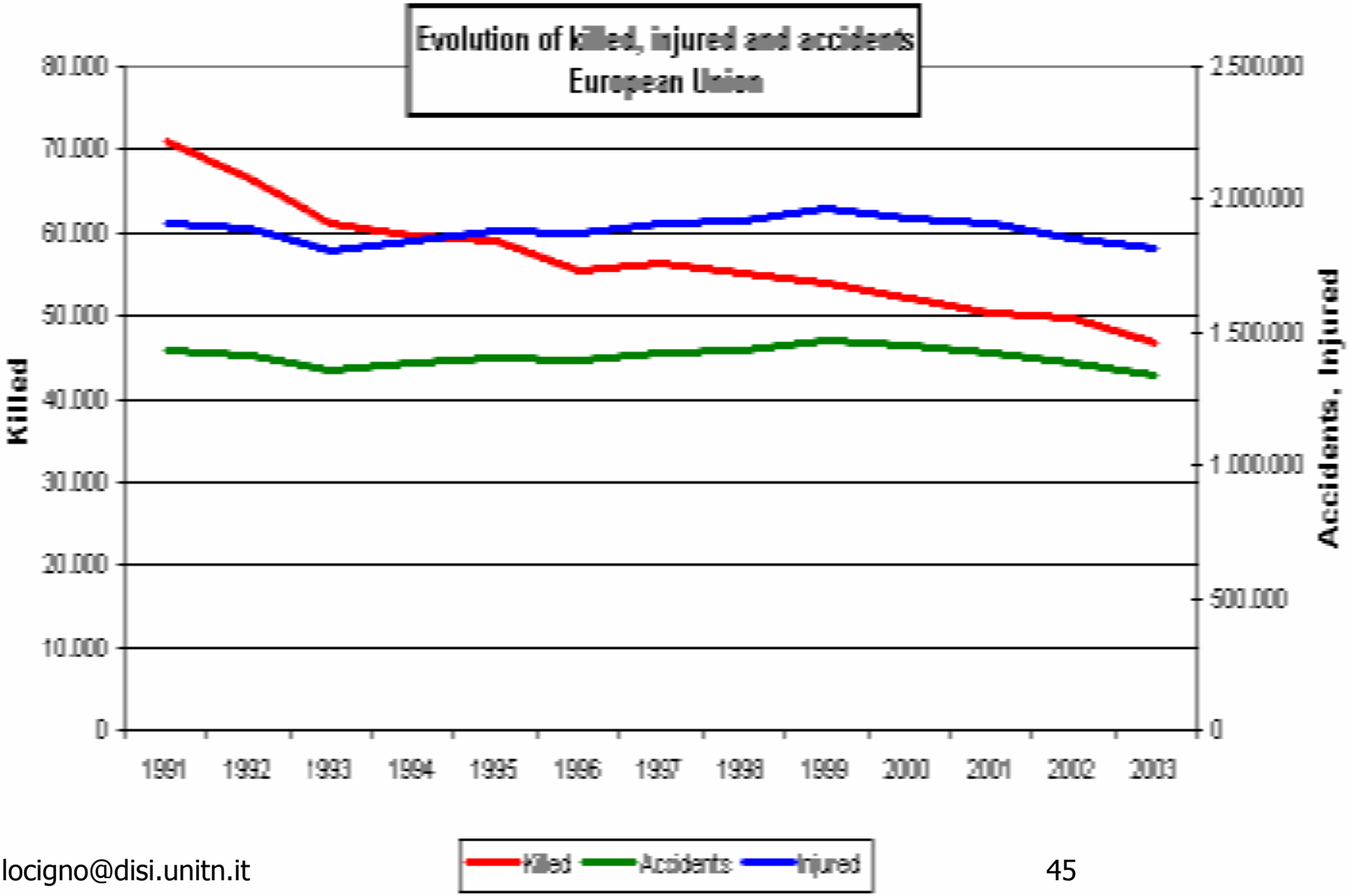
A digression on Fatalities (EU 2005)

- **Main Causes and driving errors:**
  - 95% of all road accidents involve some human error
  - In 76% of the cases the human is solely to blame
  - Misjudging, driving dynamics, weather (50%)
  - Distraction (38%)
  - **39% of passengers vehicles and 26% of trucks do not activate brakes before a collision**
  - Some 40% more do not brake effectively

- **Underlying Causes:**
  - **Alcohol**
  - **Inexperience**
  - **Tiredness**

- **Road Accidents**
  - **41.600 fatalities**
  - **1.4 million accidents involving injury**
  - **2.0 million injuries**

A digression on Fatalities (EU 2005)



Cooperative-Driving or Info-Tainment

- The main “official” push for Vanets is safety/ efficiency
- Industry (automotive) needs a revenue “golden fleece” to invest
- Industry (other) see a possible huge market for generic applications, from local info/ads to entertainment
- Technicians/scientists need to put it all together

Infrastructure and Equipment

- The average car life is 8-10 years ... with many lasting 20 or more
- Cooperative driving requires a very high penetration, say > 50%
- ... so what ...
- The chosen technology will peack in about 20 years and be still there after 40
- This is a different “pace” wrt the communication marketplace

Retrofitting
& starting from the superflous

- Communications can be put on any car as an add-on feature
  - Just like GPS navigation, most of the installed systems are not “embedded”
- Building cooperative driving on top of add-on is not feasible, but safety is much more than CoDri and InfoTainment can be appealing
  - Accidents warnings can be given to the driver, not to the breaks
  - Dangerously small distances can trigger alarms (beware of too many false alarms!)
  - ...

Working together and ... the missing leg

- We're missing the road management from the picture
- Starting from a simple information delivery systems (cheap and incremental) can convince users of the utility of retro-fitting
  - Add a communication AP every time a mobile message system is added/maintained
- When the penetration is enough increment services with the safety goal
- Cooperative Driving ... will come by itself when times are mature

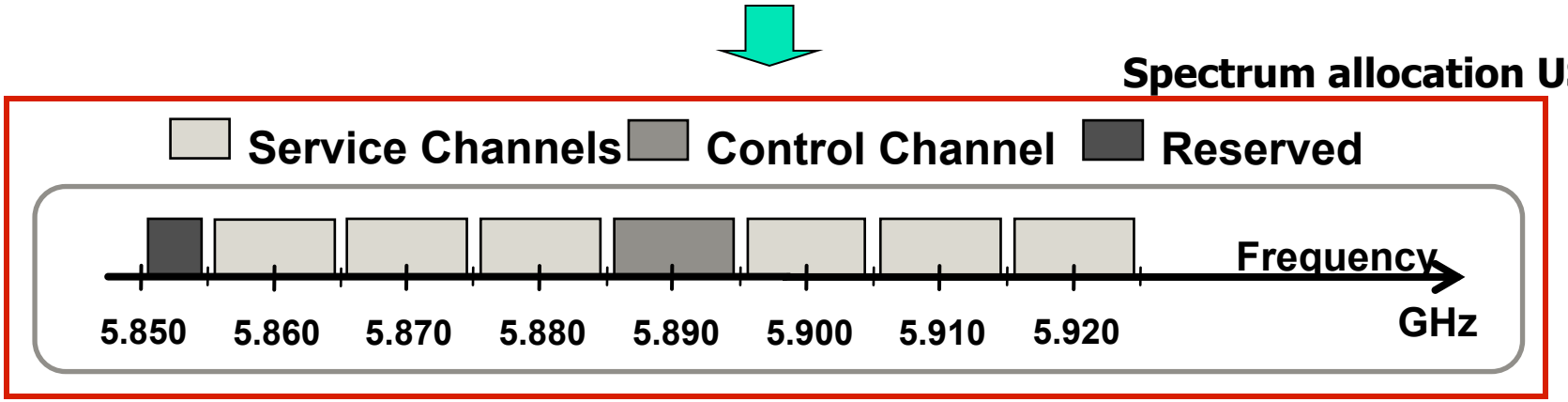
Standards

- Frequency allocation (specific for IVS) is now agreed upon in the 5.8-5.9 GHz band
  - Definitely short range (< 1000m range)
  - Licenced to avoid too much interference
  - Easy to make directional systems
- PHY is derived from OFDM WLANs
- MAC is mixed random/guaranteed access with priorities

The current IEEE WAVE standards

- **WAVE: 'Wireless Access in Vehicular Environments'**

Resource Manager: IEEE 1609.1	**Security IEEE 1609.2**
Networking Services: IEEE 1609.3	
Multichannel: IEEE 1609.4	
PHY/MAC: IEEE 802.11p	



IEEE P1609

- **Wireless Access in Vehicular Environments (WAVE)**

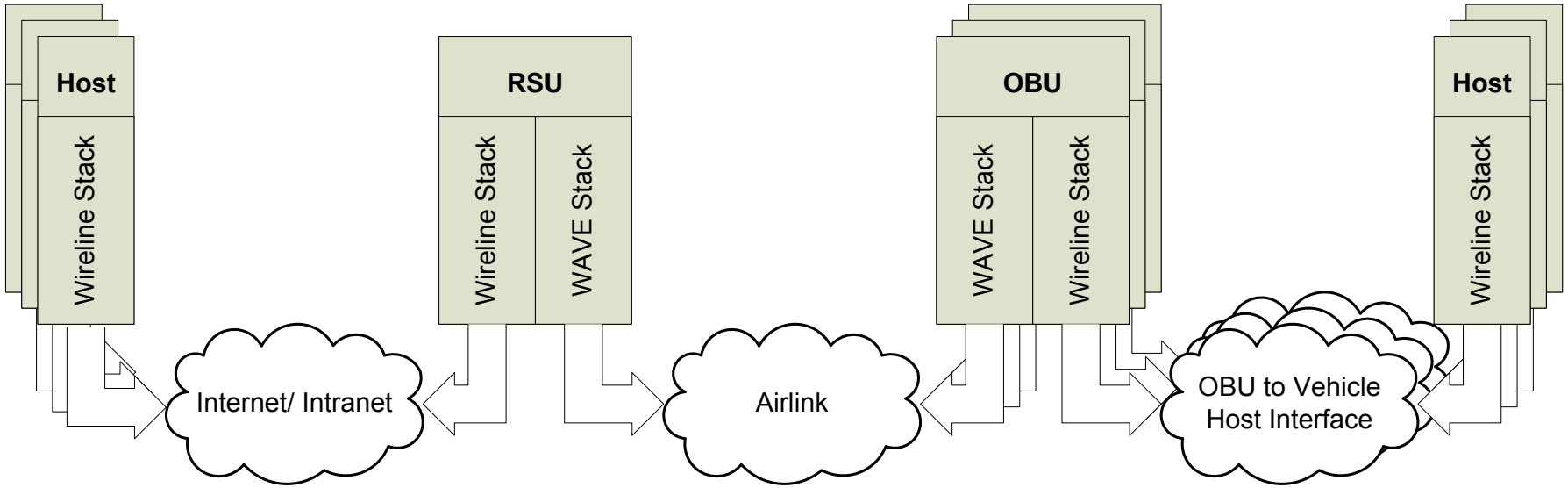
- **IEEE P1609.1 - Resource Manager**
  - services and interfaces for resource management
  - Describes key components
  - Defines data flows and resources
  - Defines command message formats and data storage formats
  - Specifies the types of devices that may be supported by the On Board Unit (OBU)
- **IEEE P1609.2 - Security Services for Applications and Management Messages**
  - Defines secure message formats and processing
  - Defines the circumstances and purposes/contents for using secure message exchanges
  - Specify mandatory processing based for specific exchanges

IEEE P1609

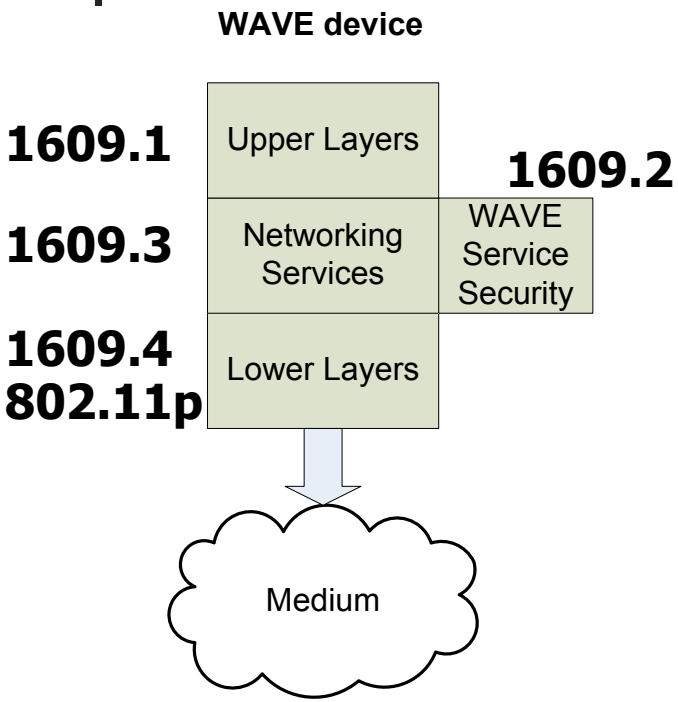
- **IEEE P1609.3 - Networking Services**
  - Network and transport layer services, including addressing and routing
  - Defines Wave Short Messages (WAVE-specific alternative to IPv6)
  - Defines the Management Information Base (MIB) for the WAVE protocol stack
- **IEEE P1609.4 - Multi-Channel Operations**
  - Enhancements to the IEEE 802.11 MAC to support WAVE operations

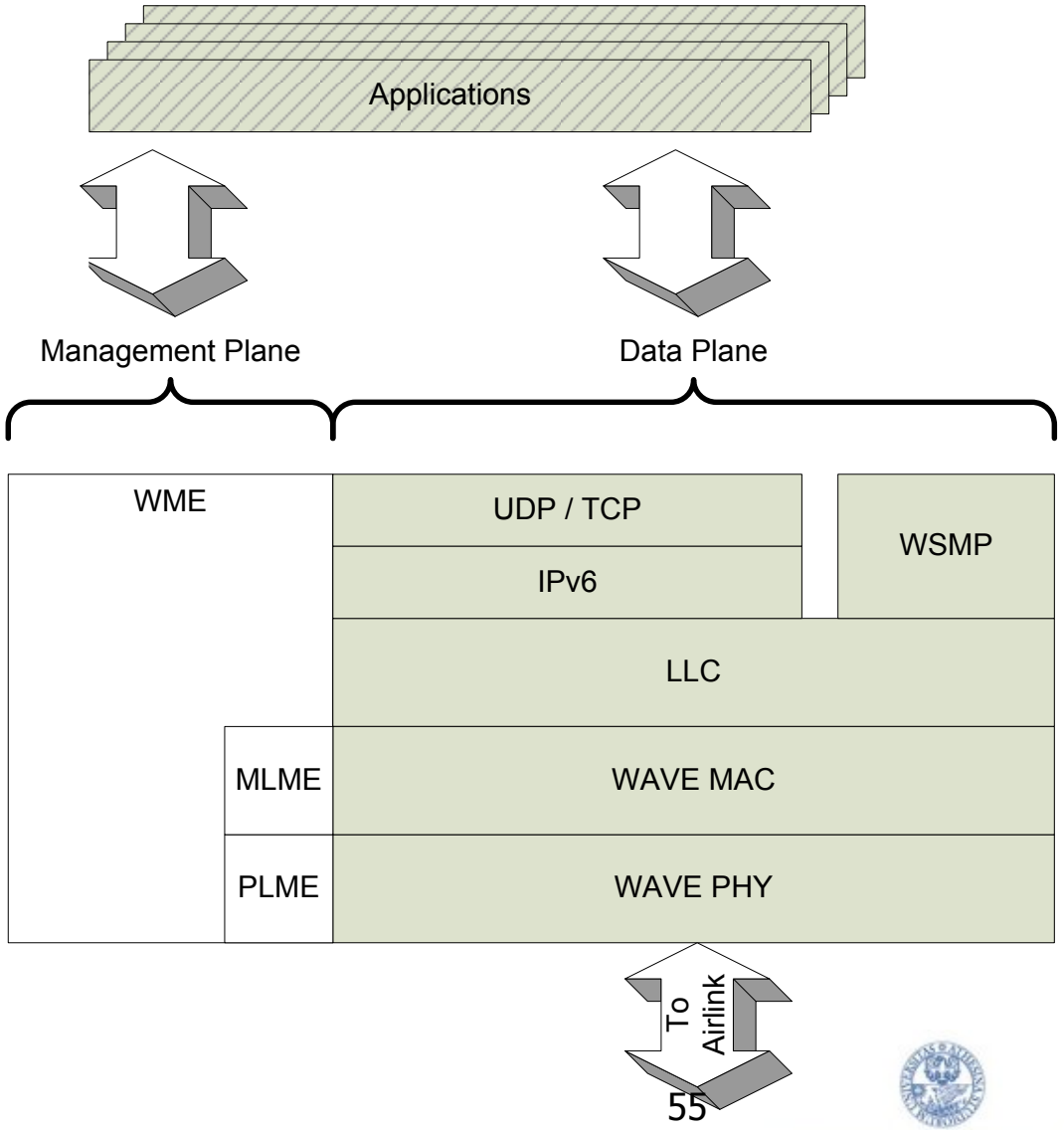
1609: system architecture

- Encompasses both C-t-C and I-t-C communications
- Defines also a subnet on the vehicle for info distribution and management



1609: protocol architecture

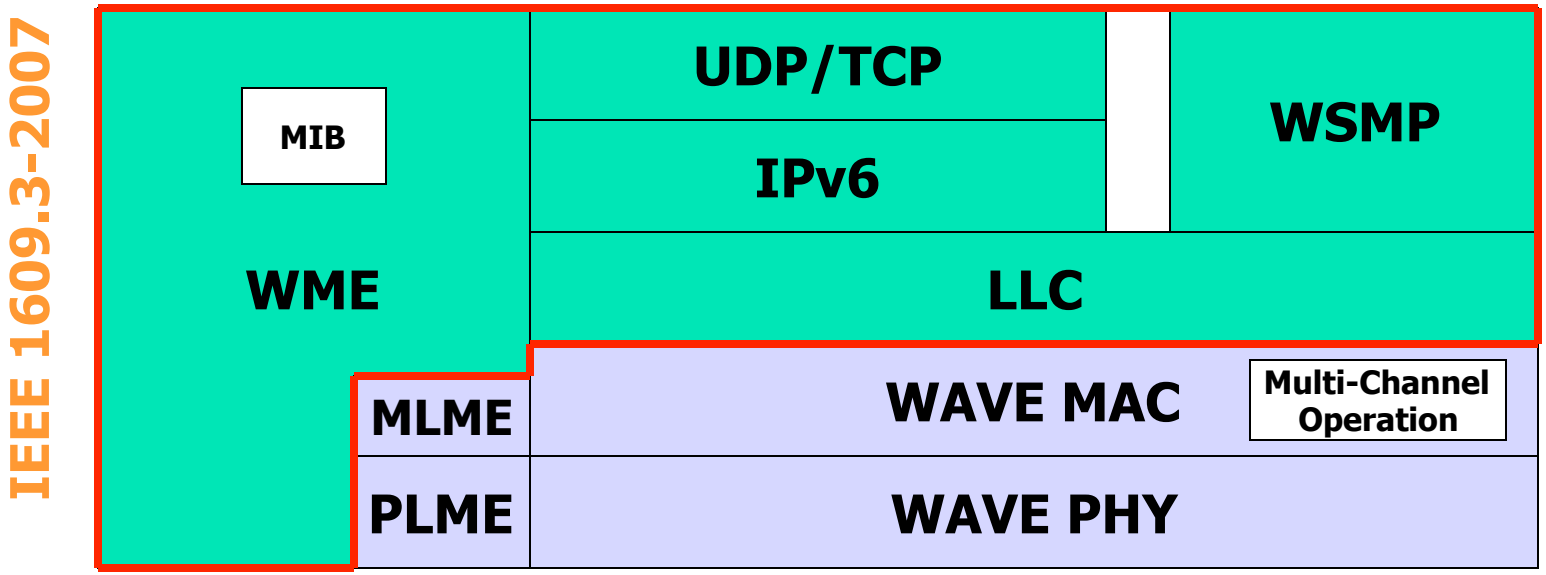




IEEE 802.11p

- Define 802.11 modes for
  - Rapidly changing PHY
  - Very short-duration communications exchanges
- Provide the minimum set of specifications to ensure interoperability
- Support transactions shorter (in time) than the minimum possible with infrastructure or ad hoc 802.11 networks
- Defines WAVE signaling and interface controlled by the MAC
- Describes functions and services required by WAVE-conformant stations

WAVE protocol stack



- WAVE: Wireless Access in Vehicular Environments
- WSMP: Wave Short Message Protocol
- WME: Wave Management Entity

WAVE Short Message Protocol

- WSM-WaveShortMessage.request

  Parameters of primitive contain 'ChannelInfo':
  - ChannelNumber
  - Adaptable
  - DataRate
  - TxPwr_Level
- Permits applications to control these transmit parameters for each individual frame
- WSM-WaveShortMessage.indication

WSM header format:

1	1	1	1	1	4	2	variable
WSM Version	Security Type	Channel Number	Date Rate	TxPwr_ Level	PSI	WSM Length	WSM Data

locigno@disi.unitn.it

Middleware: message example