Overall goals

- Separate the management plane from data plane
  - Logical virtualization of resources

- Provide visibility into network properties
  - Tie it in with authentication in access

- Include client participation in network management
  - Leverage crowd-sourcing in understanding network performance
Management-data separation

- Management plane should not depend on availability of data plane

- Separate routing computation for the management plane
  - Routing for connectivity and availability alone and not efficiency
  - Can view this as backup path(s) for management functions
  - Only authorized management traffic allowed to use these routing entries
  - Required that the second routing method be independent of the data plane routing method

- Power of two random choices
Management-data separation

- Possible routing solutions for management plane
  - For core networks, use some static routing methods, e.g., a hypercube on GUIDs
  - For DTN components, use some geographic routing techniques, e.g., GPSR

- Possible to leverage additional capabilities
  - A cellular card in each router that is independently addressable
  - Does not matter if the path is low bandwidth and high latency

- Hence, each node has two locators --- a data path locator and a management path locator

- Management traffic can be sent on the data path, but not vice versa
Network visibility

- Each entity declares a set of resources that it measures and retains that is accessible externally
  - Use XML style semantics to declare it
  - Object type (e.g., Cache content), Object value (e.g., stored content in the cache), Timestamps, Certificate of Authenticity, and Users allowed to get/set

- A get, set, and alarm interface
  - Get object @ GUID, requires credentials (or a chain of credentials)
  - Set object @ GUID with value, requires credentials
  - Set alarm, etc.

- Who is allowed access to different objects is a policy decision to be made by administrators
Some example queries

- GetGUIDs: Which GUIDs visible at a location in the past
- GetCache: What is stored in the local cache of the router
- GetGUIDQueries: Which GUIDs were accessed by someone from GNRS
- GetPathInfo: Avail bw, Latency, and Loss properties of a path
An Example

- Video server (VS) to controller:
  - alarm(path / properties)

- Controller to router:
  - alarm(path / properties)

- Video server can set some video transmission properties based on this information
Client-assisted management

- Collect measurements from diverse clients
- Use these measurements to understand network properties

Challenges
- How much data to collect?
- How can we trust this data?
- Can we aggregate this data?
WiScape stands for the Wireless landScape
Madison, WI – 155 sq.km. (more than 2 yrs of data)
Persistent dominance

<table>
<thead>
<tr>
<th></th>
<th>NetA</th>
<th>NetB</th>
<th>NetC</th>
<th>None</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>26%</td>
<td>13%</td>
<td>13%</td>
<td>48%</td>
</tr>
</tbody>
</table>

20 Km road stretch over 3 months (2 times a day)
Persistent dominance

Data between Chicago to Madison
Methodology

Aggregate in space? (zones) → Yes → What zone size should be used?
→ 250m radius

Aggregate in time? (epoch) → Yes → What time epoch should be used?
→ Zone specific O(10s of minutes)

Can we client-source data? → Yes → How many client-sourced samples?
→ Zone specific

- 250m zone radius
- For each zone: calculate time epoch
- For each zone: calculate sample quantity
Observation: Coarse grained monitoring feasible

Summary stats can be done with client participation

Can help detect flash crowds, locations with persistent bad performance
Ex: Flash crowd

Game day Camp Randall (Madison, WI)
Detect non-WiFi interferers using WiFi clients [IMC 2011]

What API needed?
What scale of measurements needed?

Goals:
Who else is in the air?
What is their impact on intended (WiFi) transmitters?
Where are they located?
Researchers Combat Wi-Fi RF Interference

University of Wisconsin researchers have developed a cost-effective solution for solving increasing Wi-Fi RF interference issues.

As many people know, wireless networks are rife with RF (Radio Frequency) interference. This is not only an issue with large-scale business Wi-Fi, but within homes as well. Non-Wi-Fi devices such as cordless phones, Bluetooth headsets, some audio and video transmitters, various Zigbee devices and more, can all interfere with the performance of Wi-Fi.

Now, researchers from the University of Wisconsin have come up with a low-cost and effective solution for detecting and remediating RF interference.

The solution, known as Airshark, is a system that detects multiple non-Wi-Fi RF devices in real-time, allowing for immediate remediation of wireless devices to take steps to mitigate interference from non-Wi-Fi devices.

Dumb cards detect and avoid rival signals from other protocols

Airshark is an experimental application bringing detect-and-avoid frequency-hopping to previously dumb Wi-Fi cards, without the addition of any new hardware.

Created by three boffins at the University of Wisconsin, Airshark uses the limited intelligence of existing Wi-Fi cards to spot other protocols hanging out in the 2.4GHz band. That includes Zigbee, Bluetooth and microwave ovens as well as everyone else, potentially making a Wi-Fi router switch channels to somewhere a little less crowded.