# Networking in a Heterogeneous, Intermittent World

Kevin Fall

Principal Engineer

Intel Research, Berkeley

*kfall@intel.com*

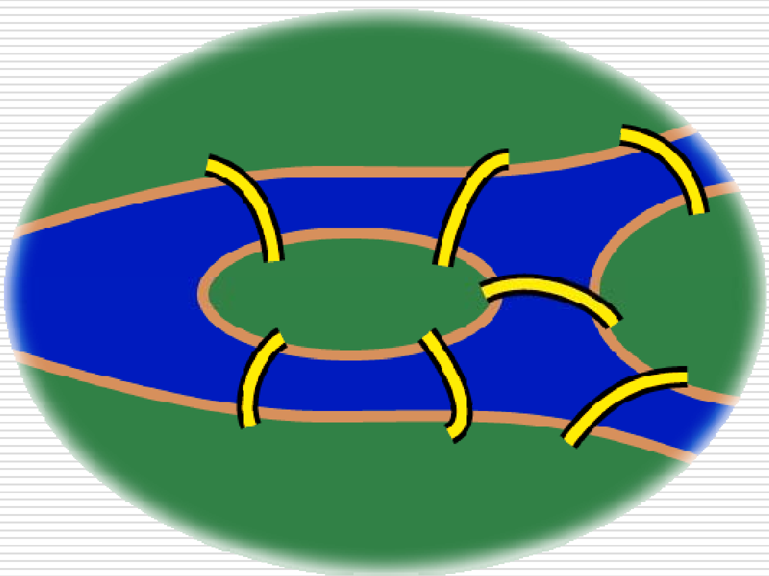
*June 28, 2006 / WoWMoM 2006 / Buffalo, NY*

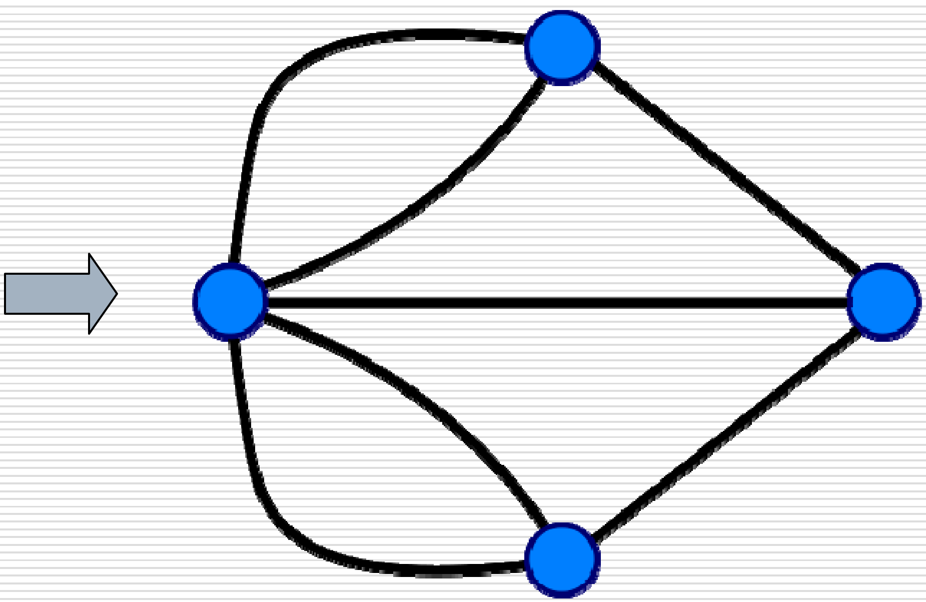
# Data Networks

- Graph Theory
- The Internet
- Challenged Networks
- Delay Tolerant Networking
- Multimedia & Futures

# Graph Theory

- Euler's Seven Bridges of Königsberg (1736)

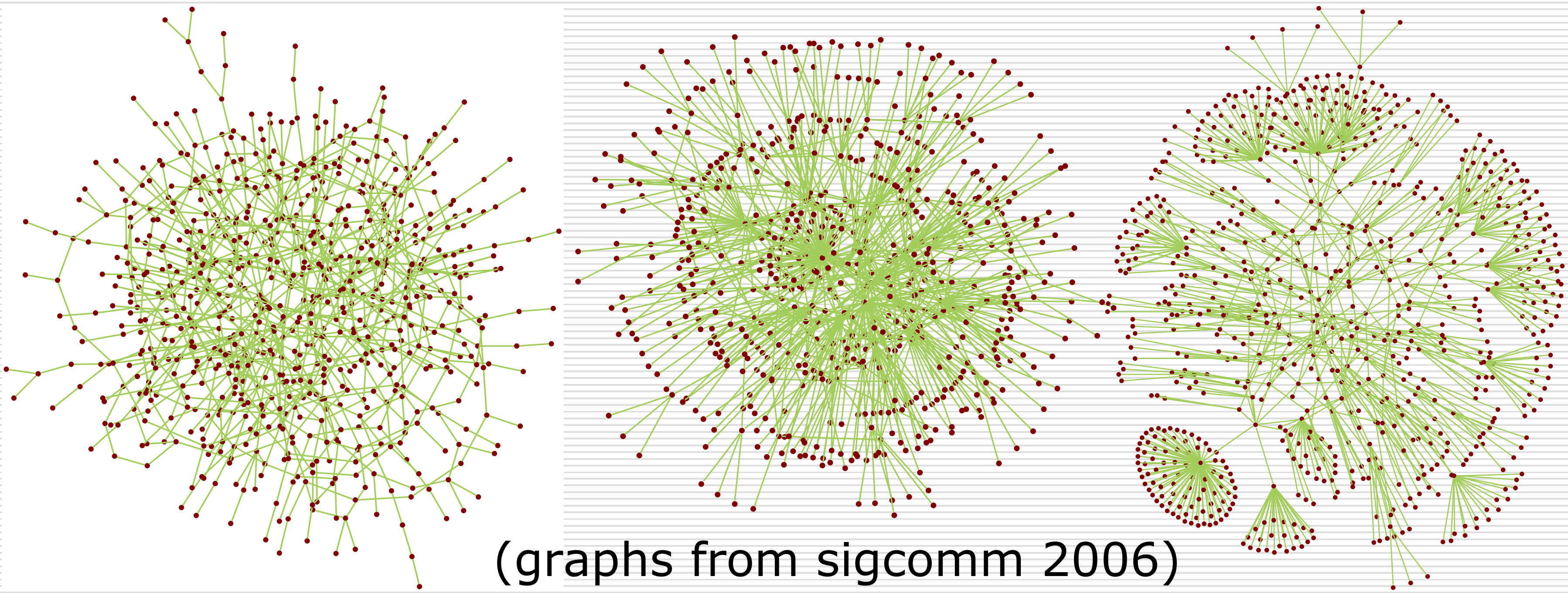




Implications: beginnings of graph theory, topology

# Network Topology

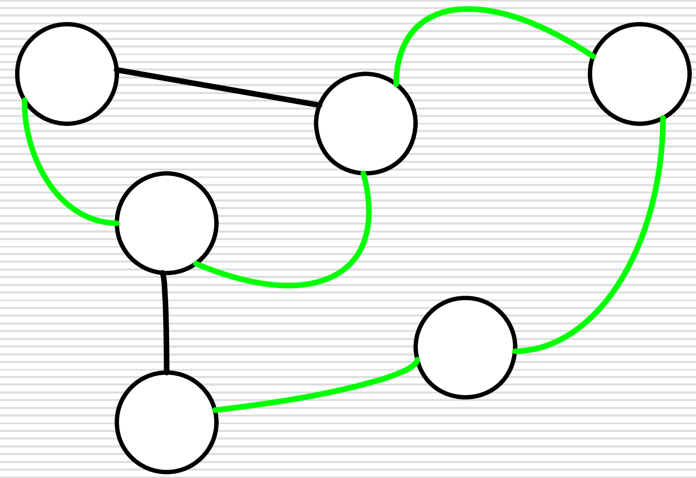
- The study of connectivity (global) and continuity (local) properties

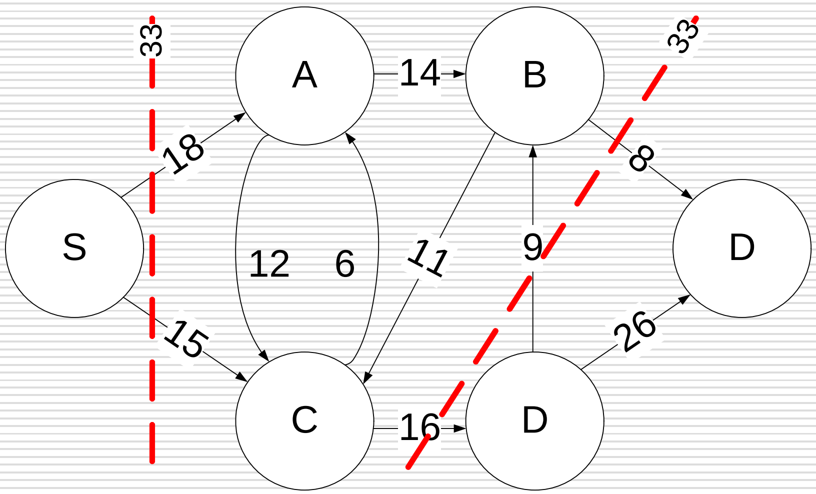


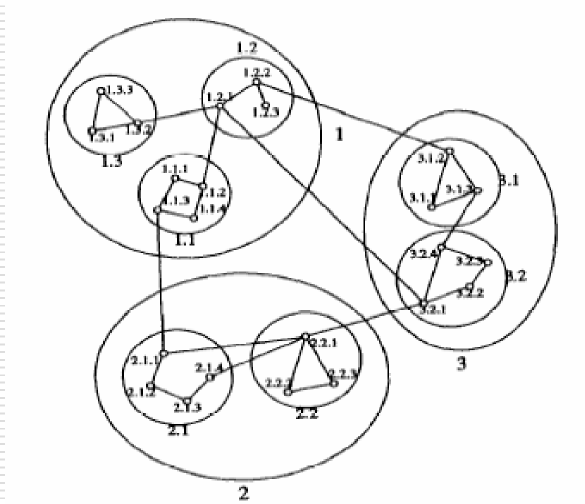
(graphs from sigcomm 2006)

# Classical Algorithms (on networks)

- Spanning Trees
- Max-Flow/Min-Cut
- Shortest Path(s)





# Internet Architecture

- Topology
  - Fully-connected (general) routing graph
    - Baran's, "On Distributed Communication Networks" (1962) – not star or 'decentralized'
  - Mostly shortest-path routes (classic algorithms)
  - Node labels (hierarchical label assignments)
    - Klienrock/Kamoun (1977)
- Data Plane
  - Store/forward of interoperable packets
    - Kleinrock, Baran, Davies, Cerf, Kahn
  - End-to-end reliability – dumb network
- Management & Security
  - Management at the application layer
  - Security and accounting secondary (at ends)

# Internet Assumptions (in practice)

- Topology graph may change a bit, but remains connected [even in MANETs]
- Node labels remain topologically-related
- E2E path has modest delay at most
  - Control loops on O(one RTT)
- E2E path doesn’t have really big, small, or asymmetric bandwidth
- Not much re-ordering
- End stations don’t cheat
- End stations are more reliable than routers
- Paths not very lossy (< 1%)
- In-network storage is limited / short-term

# Observations

- Classic graph theory used in networking assumes simple static connected graphs (w/out a vertex capacity)
- Most distributed algorithms start there and react to change by re-establishing a static graph model to work on

- Internet builds on these, adds packets, hierarchical node labels and protocols like TCP/IP…and has served us fantastically well

- But the world has changed somewhat

# The New World

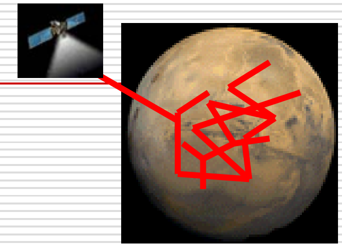
- Wired infrastructure is highly reliable and fast

    (in developed areas of the world)
- Memory is relatively cheap
- Not everybody plays nice
- Bombs aren’t the biggest net threat
- Wireless data links promotes more
    - node mobility
    - use cases for data networks
    - (difficulty in using TCP/IP)

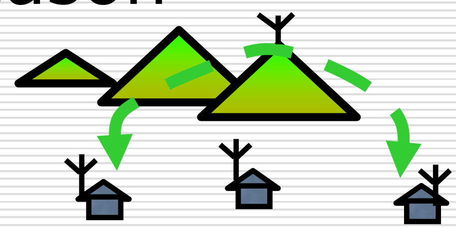
# Consequences

- For wired networks in developed areas
  - Scalability and manageability
    - more important than optimality
    - requires understanding of structure & uses
  - Little to no need for QoS-like features
  - Need for whole-net defense approaches
- For the others* ← **Focus Here**
  - operation on mobile and out-of-range or disconnected nodes needs work
  - ways to integrate legacy networks needed
  - ways to handle obstacles (technical, political) in ‘transitional economies’

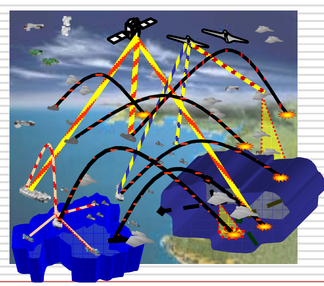
# **Challenged** Networks

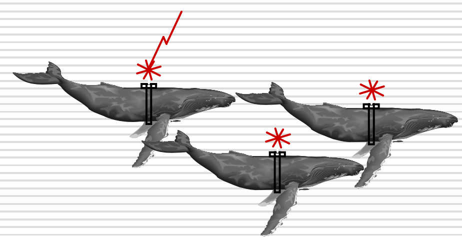


- Unusual
  - Containing features or requirements an Internet-style network architect would find surprising or difficult to reason about

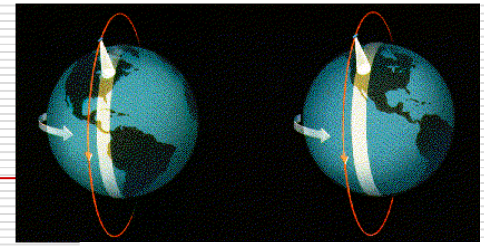


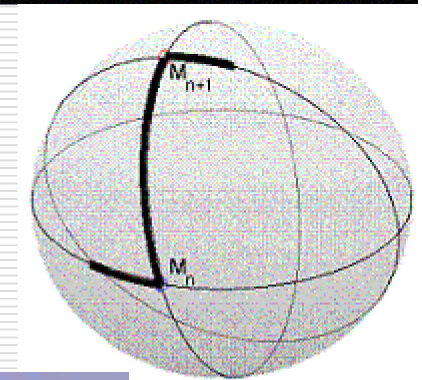
- Potentially disrupted
  - An operating environment making communications difficult



# Characteristics







- Random/predictable connectivity
- Big delays, low bandwidth
  - satellites (GEO, LEO / polar)
  - exotic links
    - deep space comms
    - underwater acoustic comms

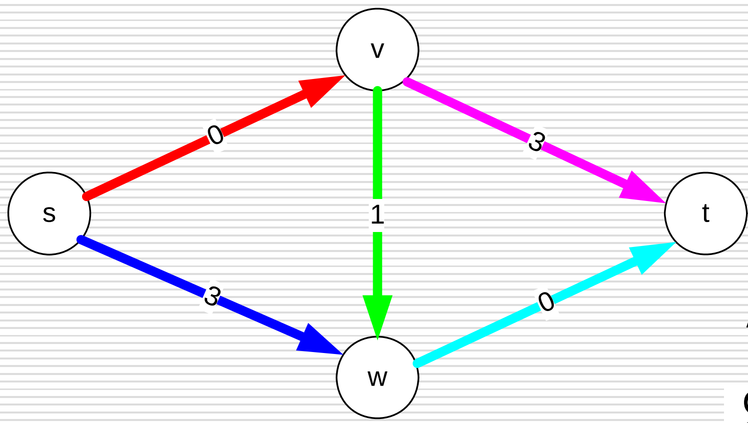
- Big delays, high bandwidth
  - Buses, mail trucks, patrol vehicles, zebras, etc.

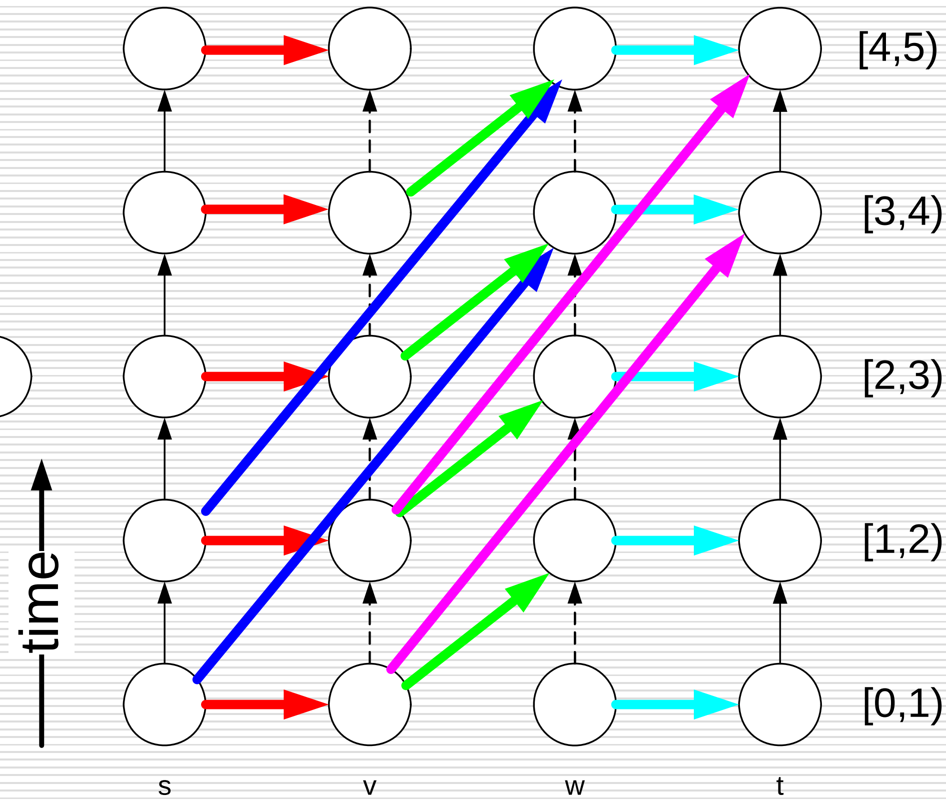
# Internet for Challenged Networks?

- Is the underlying graph theory adequate?
- Is the topology/routing approach ok?
- Is the data plane model still good?
- What happens when one or more of the Internet assumptions don’t hold (strongly)?
- Do:
  - Applications break or have intolerable performance?
  - Communications become impossible?
  - Elements of the system become less secure?

# Evolving Graph Theory

- Time-evolving graphs (flows over time/ dynamic flows)




*Time-Expanded Graphs [FF58]*
*pseudopolynomial size of input*

# Evolving Topology & Addressing

- IP uses fixed 32(128) bit addresses assigned based on topology location
  - couples location with identification
  - not inherently secured
  - aggregable ~ “scalable” [KK77]
- Name-based and flat routing
  - helps separate ID from topology
  - can be linked with application uses
  - non-aggregable (but still “scalable”)
    - see results in DHT schemes + compact routing

# Evolving the Data Plane

- Datagrams are a poor fit with
  - connection-oriented / cloud network
  - small frames (sensornets, atm)
  - poor links and unusable network storage
- Application Data Units (ADU's)
  - tailored to the application's desires
  - might be stored / retransmitted by network
  - convenient security unit

# Evolving End-to-End

- Recall ‘fate sharing’ (Clark):
  - *it is acceptable to lose the state information associated with an entity if, at the same time, the entity itself is lost*
- But state (e.g. for reliability & security) doesn’t need to be in <u>the</u> endpoint for the duration of a dialog
- The network can participate (and hold state), but it’s unwise to distribute critical state

# What to Do?

- Some problems surmountable using existing Internet/TCP/IP model
  - ‘cover up’ the link problems using performance enhancing proxies (PEPs)
  - Mostly used near “edges”
  - Brittle wrt asymmetric routing, security
- But some environments never have an e2e path (or a low-loss e2e path)
- Yet we want our applications to work

# Delay-Tolerant Networking

- Major Goals
  - Support interoperability across ‘radically heterogeneous’ networks
  - Tolerate large delays and major disruptions
- While maintaining
  - Flexibility and extensibility in support of innovation
  - Decent performance for networks with low loss/delay/errors

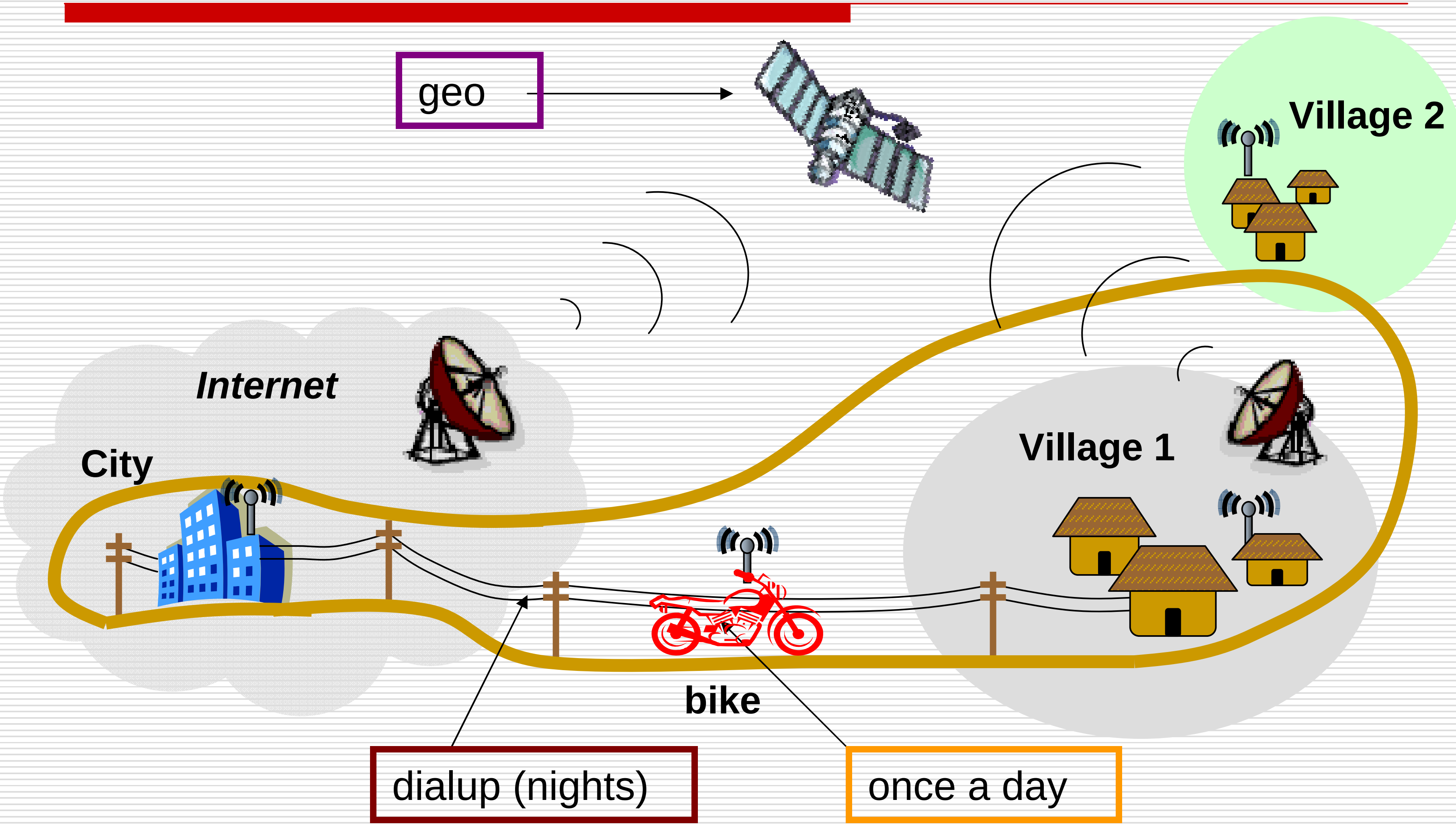
# DTN Architecture Components

- Naming
  - generalized URI (many address families)
  - late binding (mapping) to location
- Application Data Units
  - variable-sized messages (with options)
  - can be signed, fragmented, timestamped
- Store and Forward Operation
  - ‘plug-in’ routing algorithm framework
  - persistent storage for store-and-forward
- Per-(overlay)-hop & E2E security

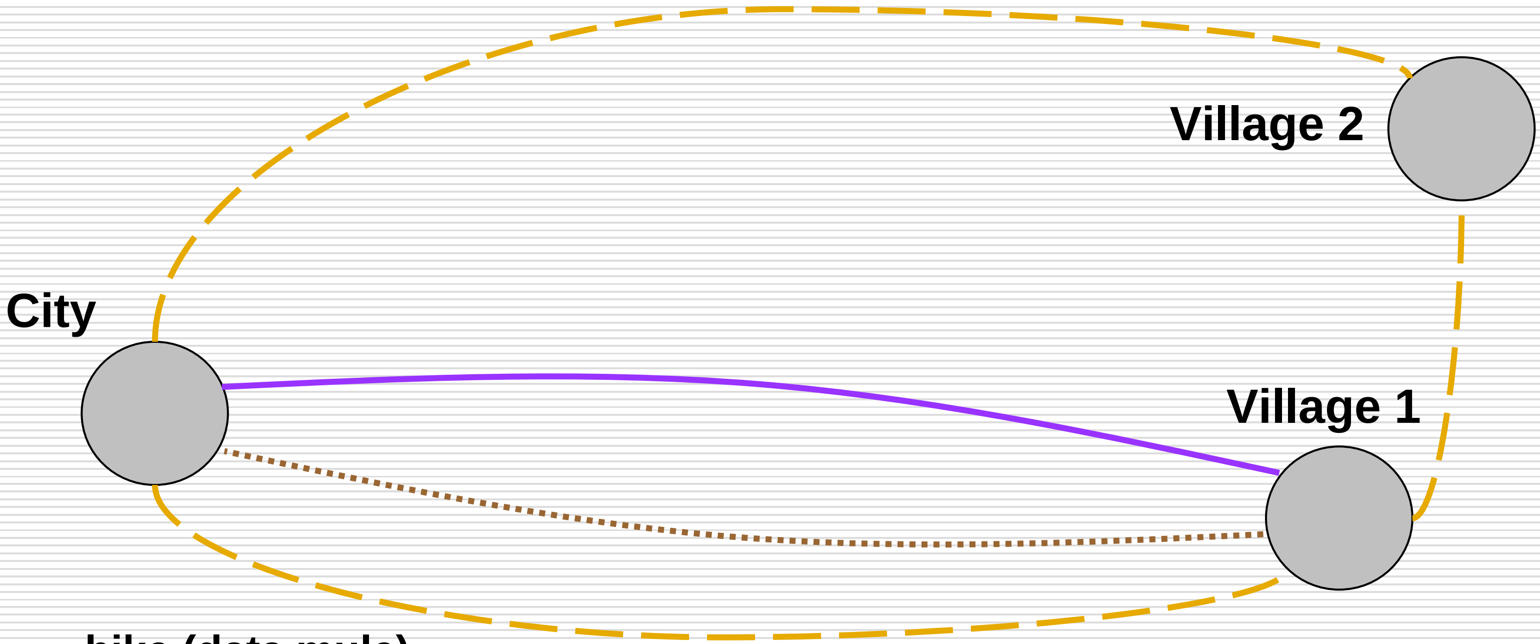
# DTN Routing

- Topology is a time-varying multigraph among DTN overlay nodes
  - can place DTN nodes at critical points
  - scheduled, predicted, and opportunistic routes
  - long-term storage during outages
- Fragmenting ADU's
  - use all links available to achieve result
  - Proactive: optimize contact volume
  - Reactive: resume where you failed

# Example Routing Problem



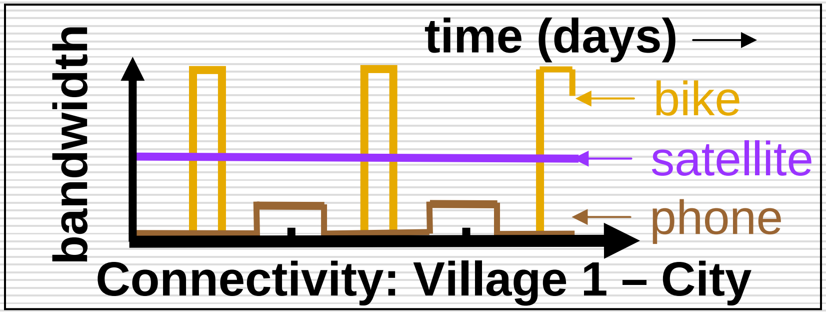
# Example Graph Abstraction




**bike (data mule)**
intermittent / scheduled / high capacity

**geo satellite**
continuous / medium/low capacity

**dial-up link**
scheduled / low capacity



# DTN Application Model

- DTN API for sending/receiving ADUs
  - agent handles bundle processing
  - asynchronous sends
  - asynchronous receipts with callbacks
- Callbacks
  - persistent registrations (~ socket bindings that span reboots)
  - can re-invoke original program or do something else
- Options for: error/ACK reporting

# DTN, Mobility & Multimedia [1]

- Multimedia ≠ ‘real time’
  - except: chat, sports, concerts, (news)
  - the rest is food for Tivo®
- Need good quality on playback
  - can’t compress or lose too much
    - except perhaps chat / channel surfing / pip
  - need timestamps and synchronization
- Consequences
  - interactivity is real driver for low latency
  - reliable transport and caching are key

# DTN, Mobility & Multimedia [2]

- DTN: reach the user any way possible
  - its about the content, not distribution
  - multigraph abstraction → multiple pipes
  - fragmentation → use multiple pipes well
  - addressing → use other legacy networks
- DTN: deliver ADUs reliably if asked to
  - variable-sized message
  - can be signed, fragmented, timestamped
  - can be secured until consumed
    - (and maybe after)

# Conclusions

- New approaches for new use cases
  - graph theory → consider time, structure
  - addressing → name / content-based
  - data plane → tolerate disruption & capitalize on storage in network
  - security → enforced not only at ends
- User experience
  - much content ok in 'real-enough' time
  - if played back at high quality

# Thanks

- Delay Tolerant Networking Research Group (DTNRG)
  - http://www.dtnrg.org
  - dtn-interest@mailman.dtnrg.org
- Organizations (partial list)
  - UC Berkeley, Intel, DARPA, NSF, MITRE, JPL, Woods Hole Oceanographic Institution (WHOI), Waterloo

*kfall@intel.com ; kfall@whoi.edu*

# Some Relevant Links

- DTNRG:
  - http://www.dtnrg.org
- DARPA DTN Program:
  - http://www.darpa.mil/ATO/solicit/DTN/index.htm
- Dieselnet:
  - http://prisms.cs.umass.edu/diesel/
- Tetherless Computing Architecture:
  - http://mindstream.watsmore.net/
- EDIFY Research Group:
  - http://edify.cse.lehigh.edu/
- Technology and Infrastructure for Emerging Regions:
  - http://tier.cs.berkeley.edu/
- Drive-Thru Internet
  - http://www.drive-thru-internet.org/