INF5090

Advanced Topics in Distributed Systems

Autonomic Networking

Andreas Mauthe
andreas@comp.lancs.ac.uk
InfoLab21, D13
Lancaster University
UK

Contents

- Objectives
  - To become familiar with the ideas behind autonomic networking
  - To learn about important basic principles
  - To understand the related issues

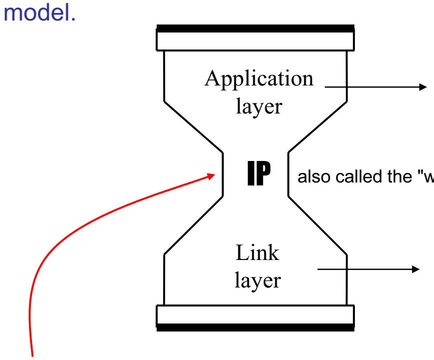
Motivation (I)

Paradigm shift in Communication – is it required and if yes why?

- The Internet suffers from architectural stress:
  - Not ready to *integrate and manage* the envisaged huge numbers of dynamically attached devices (wireless revolution, mobility, personal area networks etc)
  - Lacks integrated *monitoring and security* mechanisms
- Issues
  - No functional scalability
  - Increasing costs of network management
  - New services
  - Security threads
- Goal
  - Specifications, architectures and techniques that make networks more scalable, adaptive and autonomic networks

Motivation (II)

- Variability in the Internet is above and below IP: it's the "hour-glass" model.



www, email, ftp, ssh, DNS, peer-to-peer (eMule, BitTorrent) VoIP (Skype), VoD, grid, …

also called the "waist" of the Internet

Ethernet, wifi (802.11), ATM, SONET/SDH, FrameRelay, modem, ADSL, Cable, …

Changing/updating the Internet core (e.g., IPv6, Multicast, MIP, QoS, …) is difficult or impossible ! ⇨ Disruptive approach inevitable, this calls for new architecture

Challenges and Approach

Design and develop
New Networking Principles

- From static networks to flexible compartments
- From hierarchical routing to routing between network compartments
- Allow for interconnecting heterogeneous networks
- Ease network management – add autonomic features

- Develop a new Network Node Design
- From static layers to flexible compartments
- Multiple node compartments run in parallel
- From static layers to functional composition
  - Use functions when needed
- Include node and network monitoring

LANCASTER UNIVERSITY
Computing Department

Characteristics of Autonomic Networks

- Autonomicity of network nodes
  - Features: auto-configuring, operation independence, self-managing
- Scalability
  - Physical scalability
    - In terms of number of network nodes and communication entities
  - Functional scalability
    - Providing adequate functionality for different network types
      - e.g. small wireless ad-hoc networks to global high-speed networks
- Adaptability
  - Network conditions
    - Changes in workload, resource availability, etc.
  - Exceptional circumstances
    - Failures, attacks, etc.
- Simplicity
  - In development and deployment

Autonomous networks and their components should require little or no direct intervention during set-up and runtime but still provide a stable, reliable and secure communication infrastructure adapted to the environment they operate in and the requirements of the applications

vs.

Autonomic networks are autonomous networks with the ability to learn and adapt to changes in the environment

LANCASTER UNIVERSITY
Computing Department

The self-x Attributes

- Fundamental autonomic networking principles are expressed in various self-x attributes
  - Self-organising
    - Network nodes organise themselves to form a community
      - Dynamic role assignment
      - Joint decision making
  - Self-managing
    - Network nodes manage their behaviour according to context and rules
    - Self-configuring
      - First step of self-management within an autonomous network
  - Self-optimising
    - Network nodes
      - Adaptation of node behaviour to regular network conditions
    - Network
      - Global optimisation through joint decision making
  - Self-monitoring
    - Network nodes monitor their own state and the network state
      - Autonomous information sensing and processing
      - Observation of neighbour behaviour
  - Self-healing
    - Networks can recover from node failures through re-organisation
    - Nodes can recover through re-configuration
  - Self-protection
    - Resilience against attacks and male-behaviour

LANCASTER UNIVERSITY
Computing Department

Distributed Decision Making

- Decisions are taken in the network
  - Forwarding, multicast, filtering, translating, etc.
  - What kind of decisions can be taken in the Network?
    - How about …
      - Blocking packets, flow priorisation, encryption, etc.
- Who is taking decisions?
  - Network nodes
    - Collectively or individual
  - Based on
    - Situation/ context awareness
      - (local) knowledge about the situation
      - Constantly evolving
      - Levels:
        - » Perception: perceiving (critical) factors in the environment
        - » Inference: understanding what those factors imply
        - » Prediction: predicting system future state
    - Information exchange with other nodes
    - Policies
      - Locally executed
      - Must result in co-ordinated behaviour
- Issues
  - Discovering misbehaviour
  - Reacting to misbehaviour
  - ➔ Trust and collaboration
  - ➔ Policy representation and compliance

LANCASTER UNIVERSITY
Computing Department

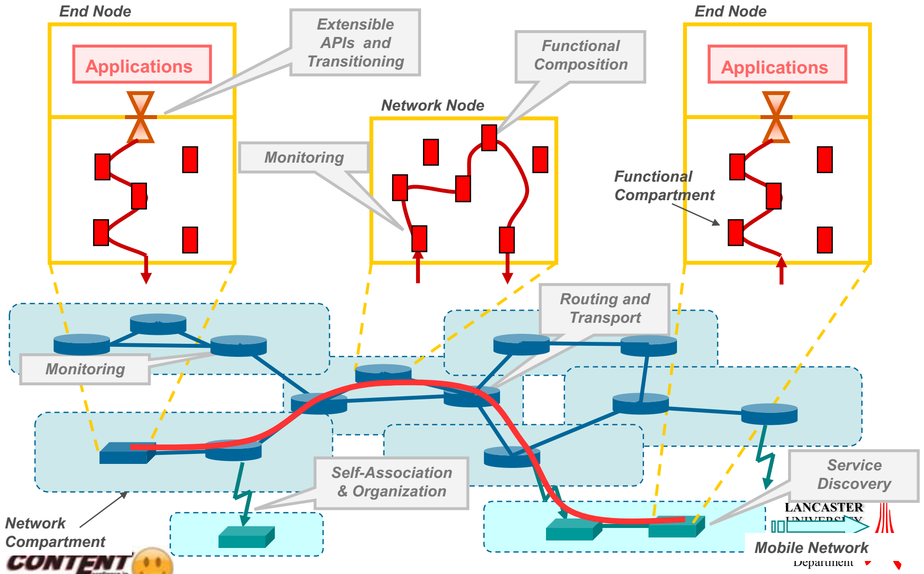
Information Gathering: Measurements

- Distributed information gathering
  - Integral part of network nodes
  - Selective perception
- Data selection
  - Locally at network nodes
    - Selection and aggregation as early as possible
    - Configurable according to network and communication type
    - Measurement methods:
      - Active vs. passive
      - Off-line vs. in-line [Pezaros et al 2004]
      - Using IETF measurement protocols?!?
        - » IP Measurement Protocol (**IPMP**), One-Way Active Measurement Protocol (**OWAMP**), etc.
  - Capturing relevant data
- Information exchange
  - Controlled by decision process
    - Dynamic perception rules
  - Using IETF exchange protocols?!?
    - **IP F**low **I**nformation e**X**port, PSAMP, etc.

LANCASTER UNIVERSITY
Computing Department

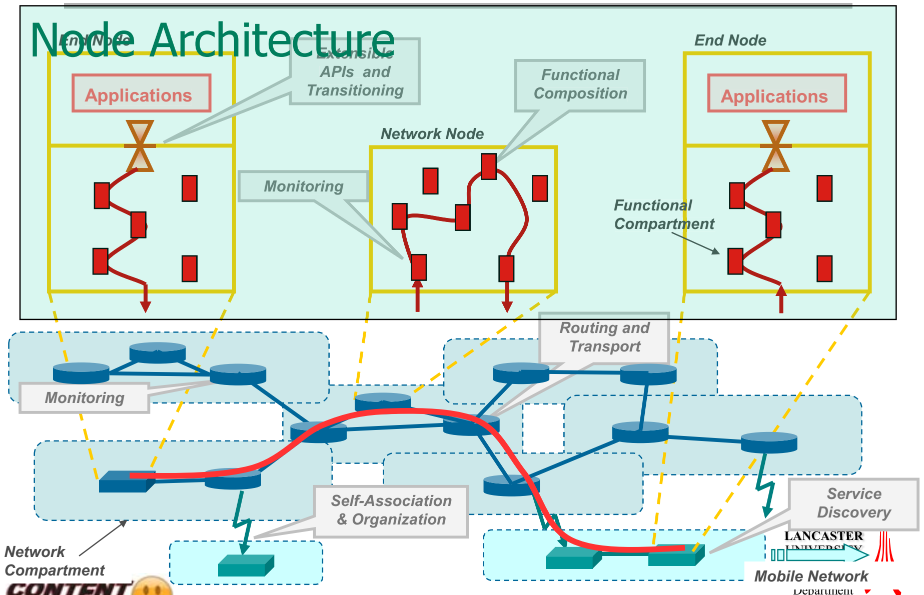
The ANA Project View -

Autonomic Networking Architecture



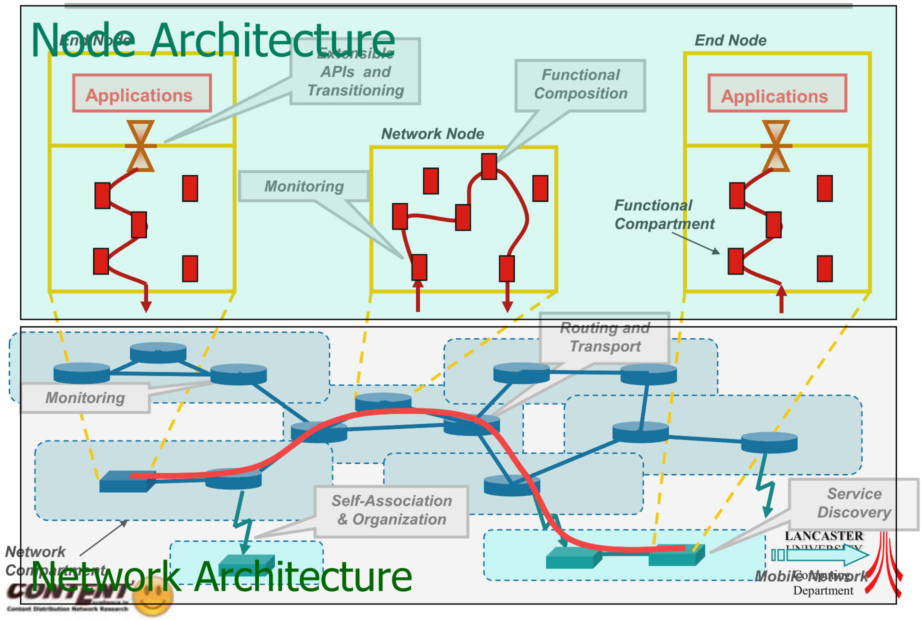
The ANA Project View -

Autonomic Networking Architecture



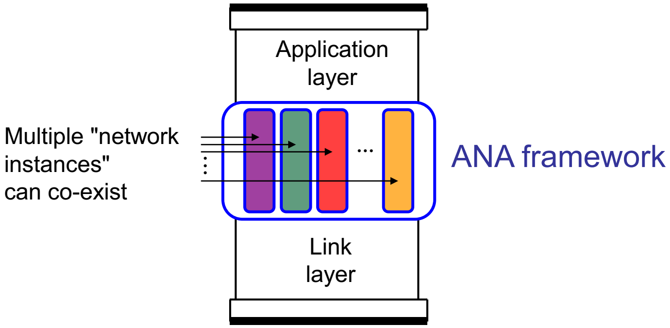
The ANA Project View -

Autonomic Networking Architecture

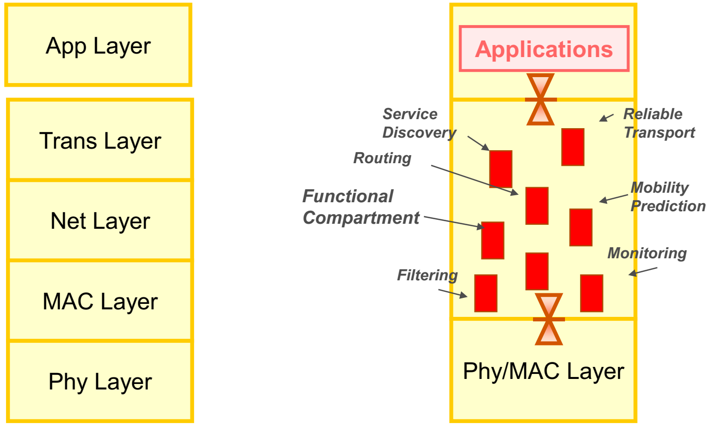


ANA <> "one-size-fits-all"

- ANA does not want to propose another "one-size-fits-all network waist".
  - ANA is a **meta-architecture** to host, interconnect, and federate multiple heterogeneous networks.

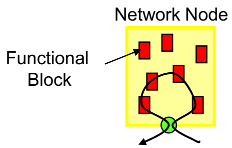


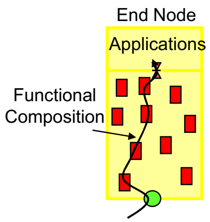
From Layers to Functional Compartments



Functional Composition

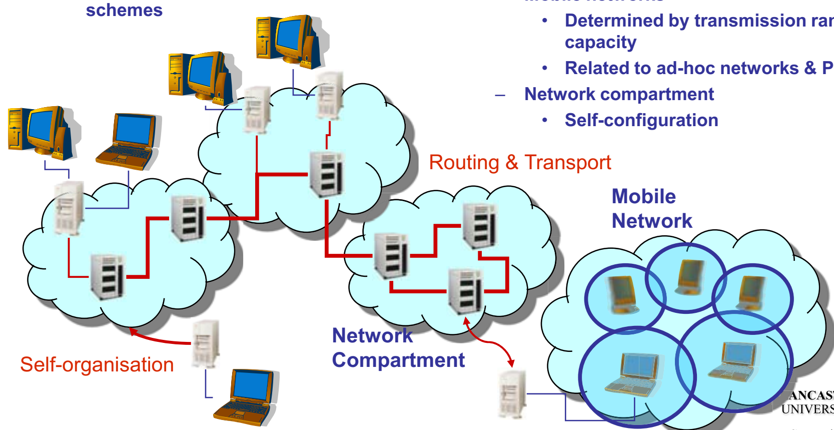
- Organisation of network functions:
  - Composition of stack, interaction models, protocol layer fusion, interlayer control loops, cross layer optimisations
  - Dynamic run-time deployment of network functions components (active networks)
- Self-Learning
  - No basic AI research in ANA, but application of machine learning to improve self-awareness of networks
  - Applications in ANA: mining monitored data (anomaly detection), learning failures causes, detecting and discriminating traffic anomalies, predicting mobility





Autonomic Networking: Network View

- **Routing & Transport**
  - **Distributed decision making**
    - **Self-configuration of forwarding tables**
  - **Routes locally determined**
  - **New concepts based on new addressing schemes**
- **Self-organisation**
  - **Mobile networks**
    - **Determined by transmission range and capacity**
    - **Related to ad-hoc networks & P2P**
  - **Network compartment**
    - **Self-configuration**



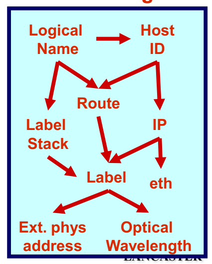
Network Issues: Set-up and configuration

- Dynamic association of network nodes
  - Self-association
    - Address allocation/ acquisition
    - Routing configuration
    - Service registration,…
  - Self-configuration
    - Role assignment and functional complexity
      - Dynamically loading networking functions
- Dynamically re-configuring & optimisation
  - Depending on network and neighbour state
    - Passive observation of neighbour behaviour
    - Node workflow auditing
- Network administration/ management
  - Distributed decision making
    - Policy based, high-level rule & goal driven
  - Information sensing and processing
    - Data gathering, selection and analysis
    - Using network context and cross layer information
- Network configuration depending on purpose and situation

Network Issues: Routing & Transport

- Routing
  - Forwarding of packets from sender to receiver
    - Within and across multiple network compartments
      - Different network types,
      - Different routing and forwarding strategies
    - ➔ Traditional and generic schemes are not sufficient
  - Additional requirements
    - Resilient and survivable communication in challenging environments
      - Episodically connected links
      - Mobile nodes
    - Dynamic and context aware routing and forwarding
- Addressing and identifying
  - Different classes of identifiers
    - Depending on scope and involved networks
- End-to-End Transport
  - Goal: minimisation of setup time and communication overhead
  - ➔ Merging and integration of functionality
    - e.g., path setup with other network procedures

AN Naming



Future?!?

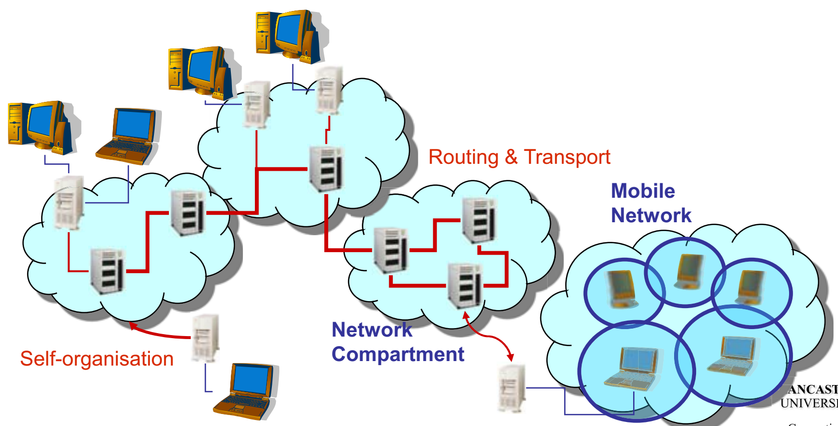
LANCASTER UNIVERSITY
Computing Department

Network Layer

- Network formation
  - Establishment of network structure and connectivity
    - Negotiation & auto-configuration of (fault tolerant) protocol components, protocol functions & network services
    - Self-organisation into resilient survivable networks
      - ➔ Infrastructures, protocols and signalling must be:
        - Secure and attack resistant
        - Authenticated if necessary
        - Adaptive and suitable for the deployment environment
    - Existing infrastructure should be used but not relied upon
      - Name servers, PKI, etc.
- Network maintenance
  - Autonomic operations to maintain the network
    - Self-managed
    - Self-diagnosis, & repair
    - Continuous re-optimisation

LANCASTER UNIVERSITY
Computing Department

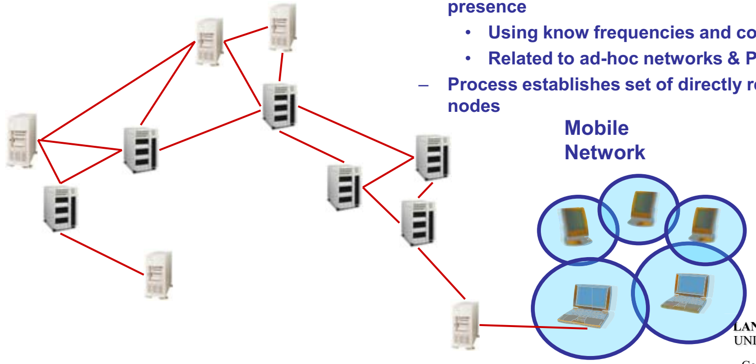
Autonomic Networking: Self-Organisation (I)



LANCASTER UNIVERSITY
Computing Department

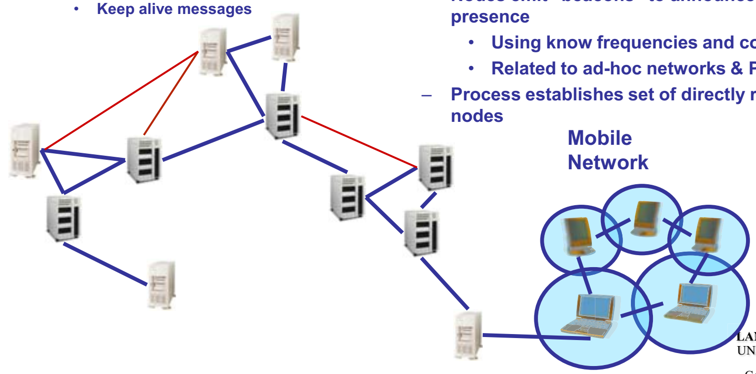
Network Organisation: Self-Organisation (II)

- Self-organisation
  - Nodes emit “beacons” to announce their presence
    - Using know frequencies and codes
    - Related to ad-hoc networks & P2P
  - Process establishes set of directly reachable nodes



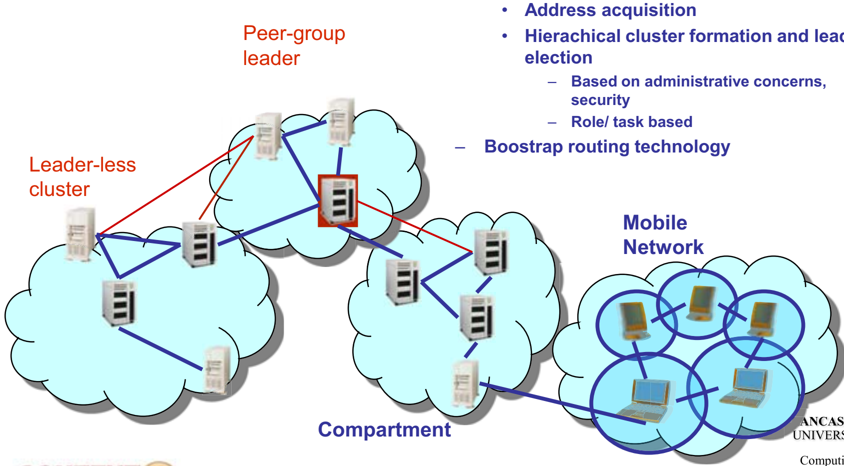
Network Organisation: Self-Organisation (III)

- Link formation
  - Pair-wise negotiation of link formation
    - Interested nodes answer beacons
    - Exchange identification, node and link characteristics
      - Layer 2 connectivity structure
  - Maintainance
    - Keep alive messages
- Self-organisation
  - Nodes emit “beacons” to announce their presence
    - Using know frequencies and codes
    - Related to ad-hoc networks & P2P
  - Process establishes set of directly reachable nodes



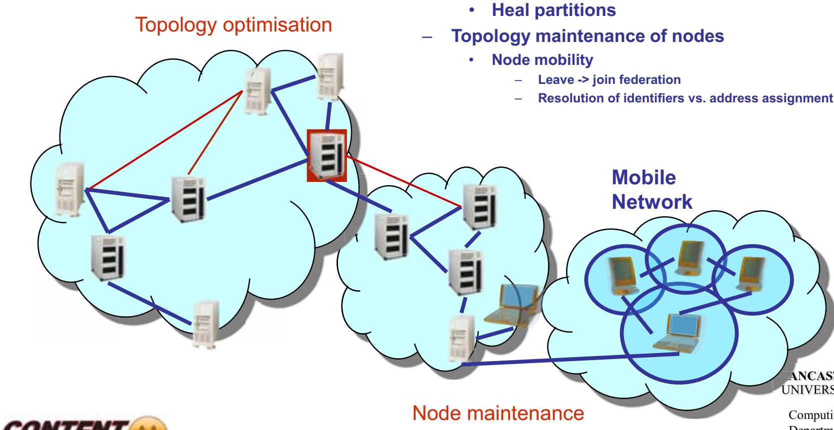
Network Organisation: Self-Organisation (IV)

- Self-organisation & Federation
  - Communication nodes organise into federations
    - Address acquisition
    - Hierachical cluster formation and leader election
      - Based on administrative concerns, security
      - Role/ task based
  - Boostrap routing technology



Network Organisation: Self-Organisation (V)

- Topology optimisation and maintenance
  - Topology maintenance of federation
    - Split/ merge
      - Due to group mobility and dynamic coalitions
    - Heal partitions
  - Topology maintenance of nodes
    - Node mobility
      - Leave -> join federation
      - Resolution of identifiers vs. address assignment



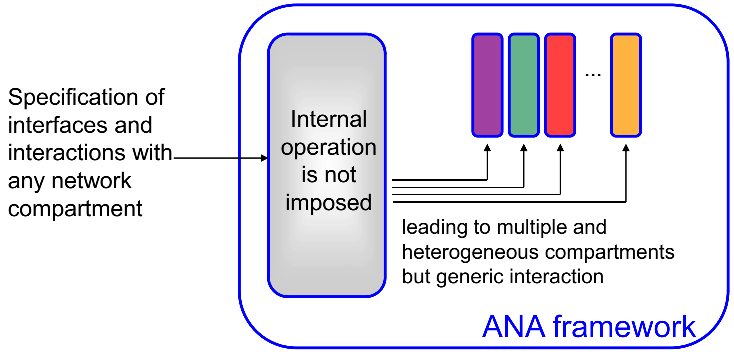
Autonomic Networking Abstractions

- Abstractions (to be detailed in the following slides*):
  - Compartment
  - Information Channel (IC)
  - Information Dispatch Point (IDP)
  - Functional Block (FB)

* The slides are part of the ANA Blueprint

Compartment

- Compartment = wrapper for networks
  - Does not specify the internal structure but how compartments interact



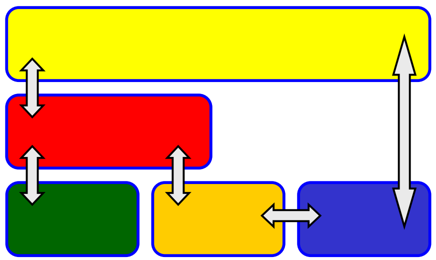
Compartment: The basic Networking Unit

- A (network) compartment implements the operational rules and administrative policies for a given communication context. It defines:
  - How to join and leave a compartment: **member** registration, trust model, authentication, etc.
  - **How to reach** (communicate with) another member: peer resolution, addressing, routing, etc.
  - The compartment-wide policies: **interaction rules** with "external world", the compartment **boundaries** (administrative or technical), peerings with other compartments, etc.

Compartments decompose communication systems and networks into smaller and easier manageable units.

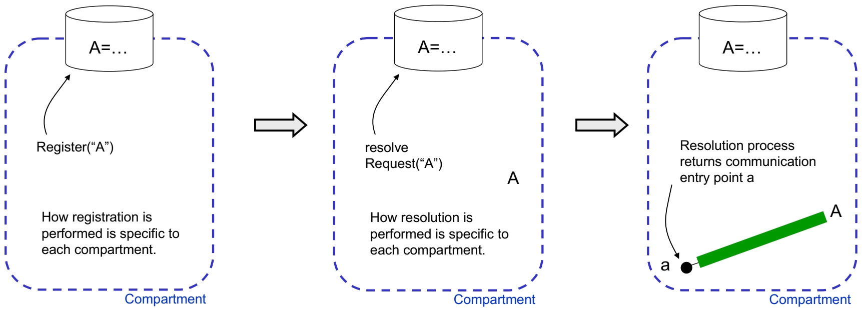
Compartment Abstraction

- The compartment abstraction serves as the unit for the federation of networks into global-scale communication systems.
  - Compartments can be overlaid, i.e. compartments can use the communication services of other compartments (and vice versa).



Compartment Functionality

- Registration and resolution are key functionalities of compartments.
  - Each compartment defines a conceptual membership database.
  - Registration: explicit joining and exposing is required ("default-off" model).
  - Resolution: explicit request before sending ("no sending in the void").

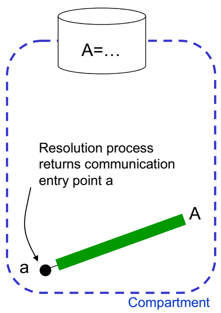


Naming and Addressing

- Addressing and naming are left to compartments.
  - Each compartment is free to use any addressing and naming schemes (or is free to not use addresses, for example in sensor networks).
- The main advantages are:
  - No need to manage a unique global addressing scheme.
  - No need to impose a unique way to resolve names.
  - ANA is open to future addressing and naming schemes.
- The main drawbacks are:
  - Global routing becomes something similar to searching. (if communicating parties are not all members of a given compartment).

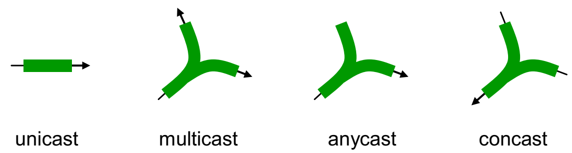
Local Labels for Handling (global) Addresses

- "Resolution of members" results in a local label
  - Addresses (if any) and names (if any) limited as input for resolution
  - Applications send data to labels (which stands for a communication entry point)
- Properties of local labels:
  - Size of labels can change from device to device
  - Labels' lifetime = communication lifetime (like sockets)
  - No need to manage a unique global addressing scheme
  - ANA is open to future addressing and naming schemes (via resolution)



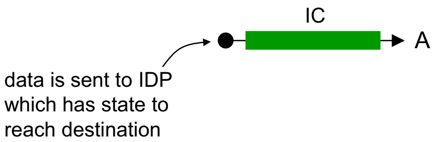
Information Channels (ICs)

- Resolution process returns access to an "information channel" that can be used to reach the target member(s).
  - Various types of information channels.



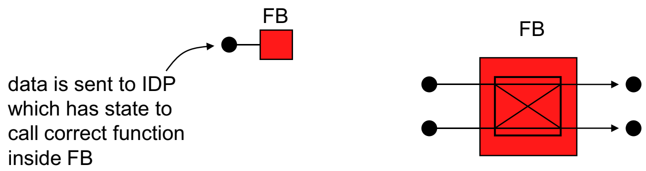
Information Dispatch Points (IDPs)

- Startpoints instead of endpoints
  - Communication is always towards a startpoint, or information dispatch point (IDP)
    - Ability to bind to destinations in an address agnostic way.
    - This is important to support many flavors of compartments that can use different types of addresses and names.
    - Useful decoupling between identifiers and means to address them.

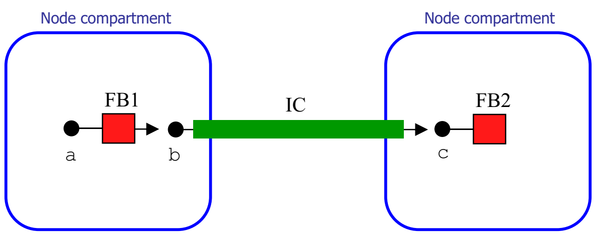


Functional Blocks (FBs)

- Code and state that can process data packets.
  - Protocols and algorithms are represented as FBs.
  - Access to FBs is also via information dispatch points (IDPs).
  - FBs can have multiple input and output IDPs.
  - FB internally selects output IDP(s) to which data is sent.

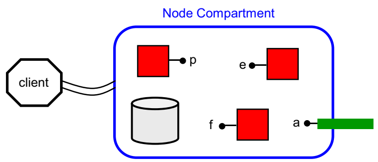


How ICs, FBs, and IDPs fit together



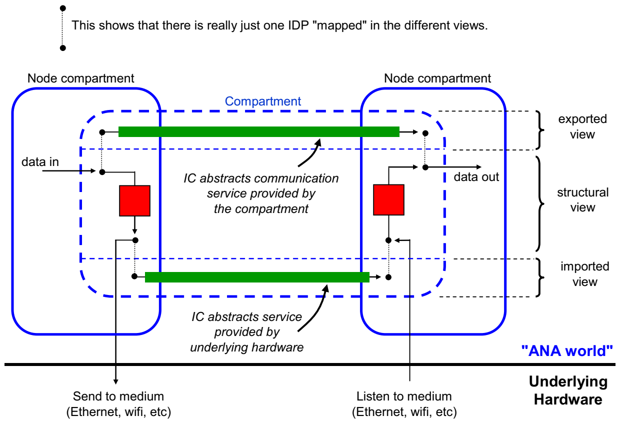
Modeling Nodes as Compartments

- Organise a node's functionalities as (compartment) members:
  - Member database: catalog of available functions
  - Resolution step to access a given function
    - Also implements access control.
  - Resolution instantiates functional blocks (FBs)
  - The node compartment hosts/executes FBs and IDPs
- Applications first attach to the node compartment:
  The node compartment is the "startpoint" of any communication.



Different « Views" for a Compartment

- A network compartment has different views, for different usage.

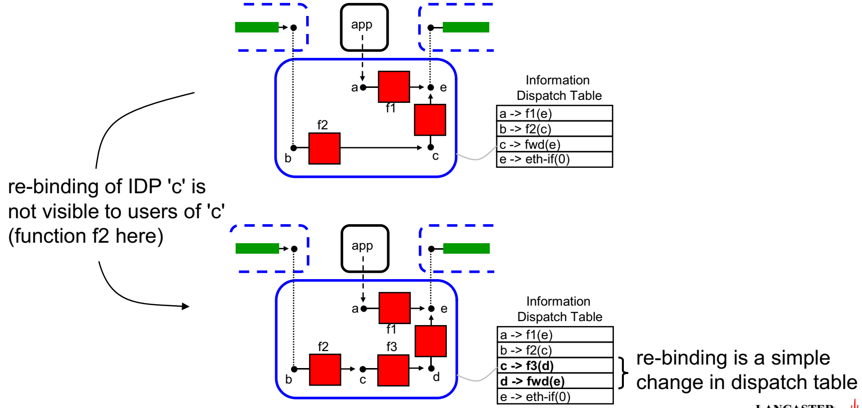


LANCASTER UNIVERSITY
Computing Department

CONTENT

Functional Composition (I)

- "Chains" of functions are setup on-demand in a dynamic way.
  - Packet dispatching in ANA is based on IDPs.



LANCASTER UNIVERSITY
Computing Department

CONTENT

Functional Composition (II)

- Motivation
  - Varying roles of network nodes
  - Changing network conditions
  - Varying end-to-end paths
  - Different application requirements
- Idea
  - Customisation of communication structures
    - On-demand creation and removal of custom communication structures
- Approaches
  - Cross-layer interaction vs. …
    - Cross over-/underlay optimisation
    - Using cross-layer information and parameterisation
      - "Knobs" and "Dials"
    - Establishing interlayer control loops
  - Modular network heaps
    - Pool of protocol functions
      - Network and transport
    - Composed according to the requirements of specific communication instances
      - Ad-hoc and on-demand
    - Searchable functionality
      - Search and association of modules according to requirements

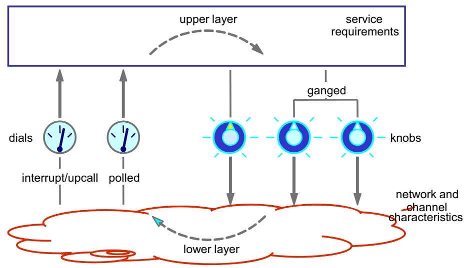
LANCASTER UNIVERSITY
Computing Department

CONTENT

Cross-layer/Component Information Sharing

- Issues
  - Current transport protocols assume
    - Strongly connected and stable, symmetric end-to-end paths via a reliable medium
  - e.g. TCP
    - Combined flow, error and congestion control
    - Unable to discriminate channel loss from congestion
      - Channel loss is treated like congestion by throttling the source
        - » Wrong action in uncongested networks with weak links
- Information and control flow required
  - Feedback through "dials" between the functional components/ layers
  - Influencing lower component/ layer functionality by "knobs"
  - e.g. error control based on loss characteristics

➔ **Cross Layer Design needs special attention to avoid unwanted interactions between the closed loop systems across the layers**

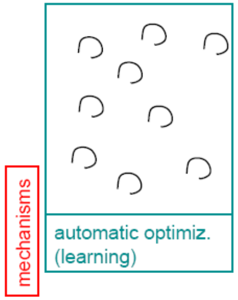


LANCASTER UNIVERSITY
Computing Department

CONTENT

Modular Network Heaps

- Structure
  - Basic mechanisms
    - Communication primitives
    - Network primitives
    - Multiple mechanisms
      - Alternative approaches
  - Control loops
    - To constantly optimise communication se
    - Automated control loop management
      - Possibly using machine learning
    - Customisation should be possible

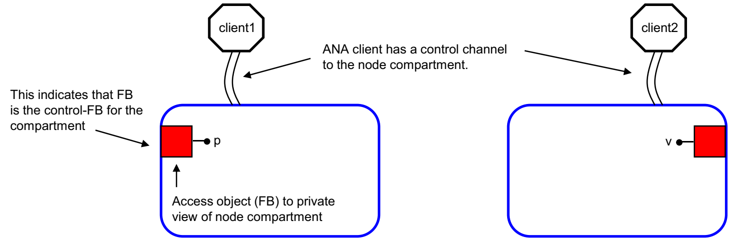


➔ Choices is important for development/ evolution
  - Therefore: always provide at least two ways of doing a job
    - 2 or more paths
    - 2 or more addresses
    - 2 or more transport protocols

  one is primitive, one is sophisticated, one is a shortcut . . .

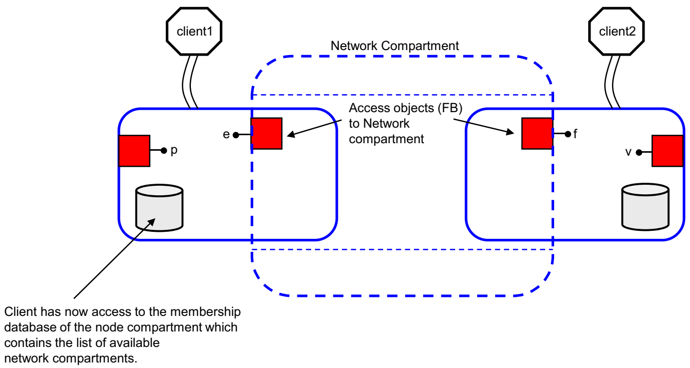
Example of Communication Setup

- Interaction with the node compartment is via a special kind of FB called an "access object (AO)".
  - For example, register and resolve requests are sent to the AO.



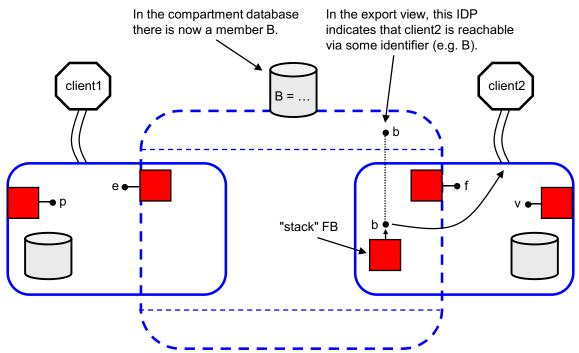
Example of Communication Setup (II)

- Clients get access to the network compartment access objects.



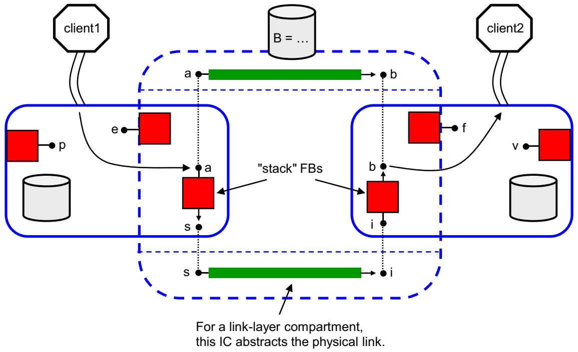
Example of Communication Setup (III)

- Client2 registers (via the IDP 'f') an identifier "B" with network compartment.
  - Conceptually, this creates an entry in the membership database.



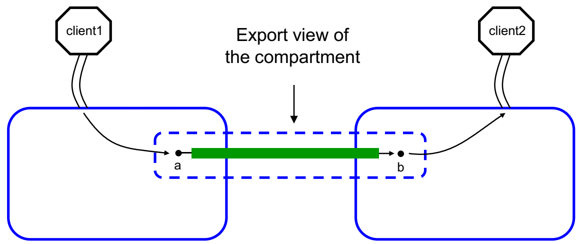
Example of Communication Setup (IV)

- Client1 resolves (via the IDP 'e') the identifier "B" and receives startpoint IDP 'a'.

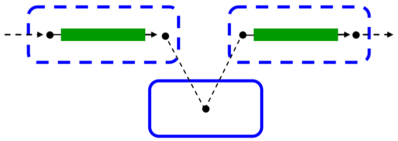


Example of Communication Setup (V)

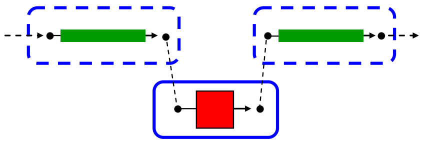
- Typically, client1 only sees exported view (unless compartment exposes internal operation).



Forwarding … Some Examples.

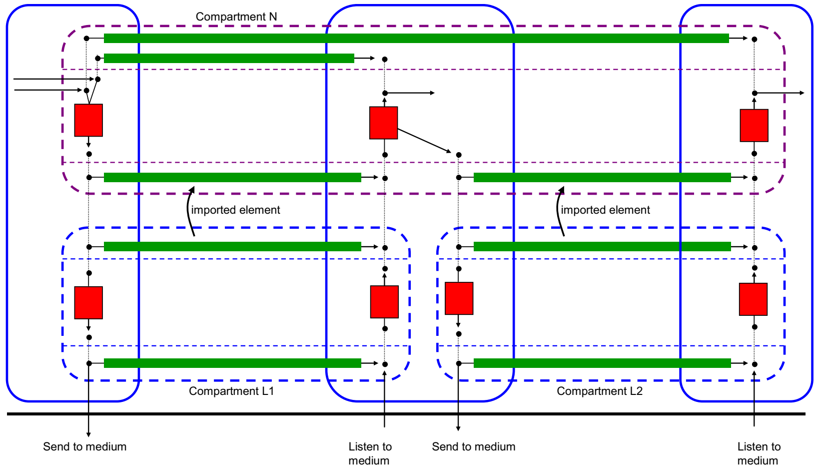


Bridging



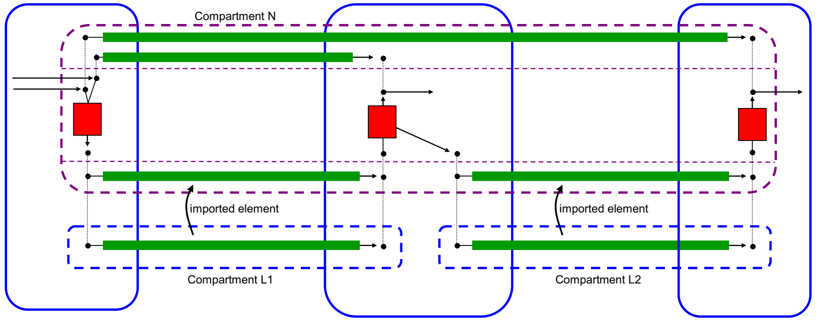
+ intermediate processing

Overlay Scenario with Compartments



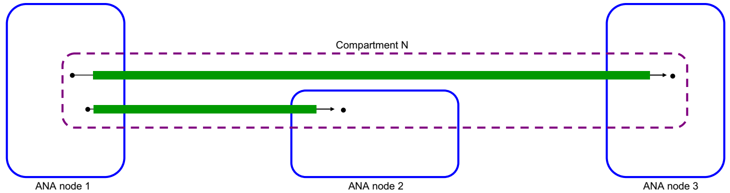
Overlay Scenario with Compartments (II)

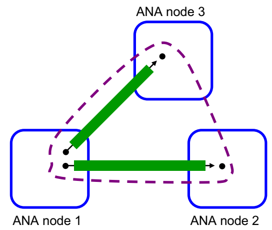
- Same figure but only with exported views of L* compartments



Overlay Scenario with Compartments (III)

- Figure just showing export view of compartment N.



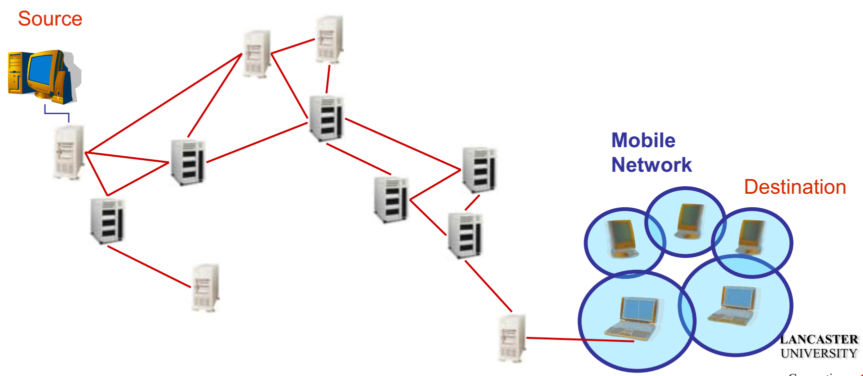


Which could also be drawn like that
(just showing the export view).

Communication Aspects

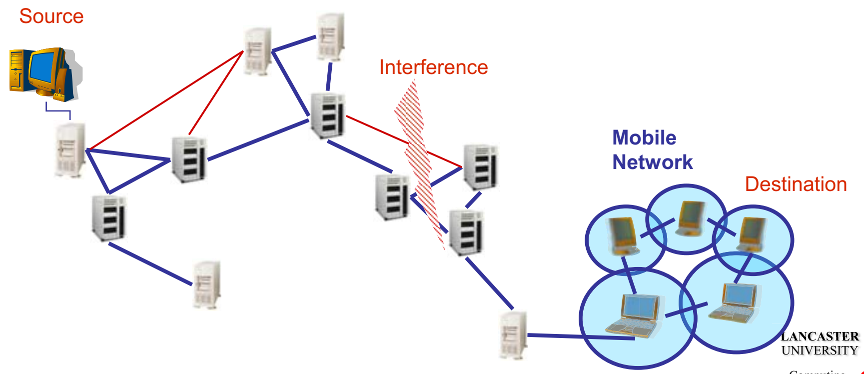
- Network layer functions
  - Identification: permanent host and/or service identifiers for communicating entities
  - Addressing: (temporal) network layer topology identifiers for communicating entities
  - Forwarding
  - Routing
  - Signalling: network layer control traffic
  - Traffic management: management of traffic and congestion
- Forwarding vs. routing
  - Forwarding: transfer of packets hop-by-hop
    - Using link-layer services
    - Network layer (node) determines next hop
      - Per packet decision (based on forwarding table)
  - Routing: determining/ establishing path to forward packets on
    - Routing algorithm independent of forwarding
      - Forwarding table populated according to routing algorithm
      - Routing generally per flow/ connection
    - Routing algorithm assume stable states
      - Complete end-to-end path must exist at one point
      - Link outages are treated as faults that must be repaired

Communication along unstable Paths (I)



Communication along unstable Paths (II)

- With interface or suspected eavesdropping
  - Routing cannot converge on a source to destination path

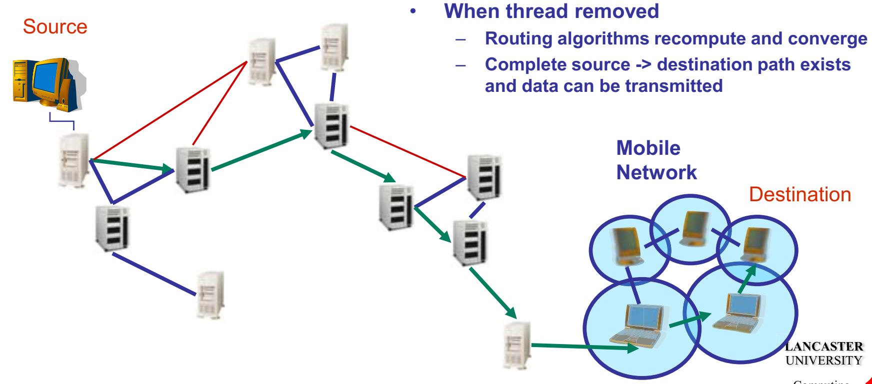


CONTENT

LANCASTER UNIVERSITY
Computing Department

Communication along unstable Paths (III)

- With interface or suspected eavesdropping
  - Routing cannot converge on a source to destination path
- When thread removed
  - Routing algorithms recompute and converge
  - Complete source -> destination path exists and data can be transmitted

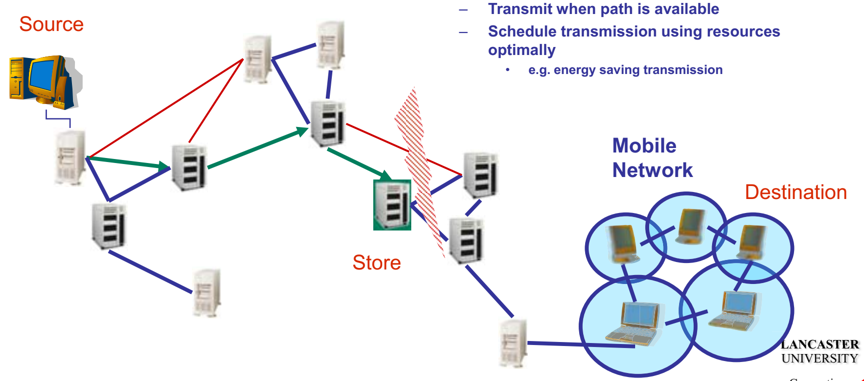


CONTENT

LANCASTER UNIVERSITY
Computing Department

Communication along unstable Paths (IV)

- Routing convergence
  - Unstable and episodic connectivity has to be assumed
- Survivable communication
  - Assume eventual connectivity
  - Store-and-forward when necessary
  - Transmit when path is available
  - Schedule transmission using resources optimally
    - e.g. energy saving transmission

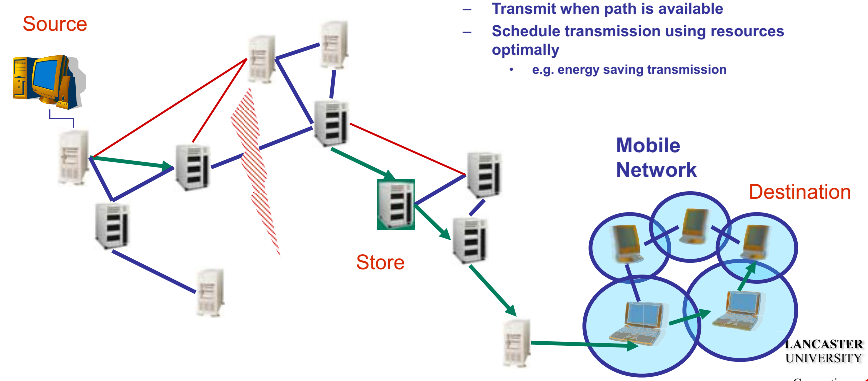


CONTENT

LANCASTER UNIVERSITY
Computing Department

Communication along unstable Paths (IV)

- Routing convergence
  - Unstable and episodic connectivity has to be assumed
- Survivable communication
  - Assume eventual connectivity
  - Store-and-forward when necessary
  - Transmit when path is available
  - Schedule transmission using resources optimally
    - e.g. energy saving transmission

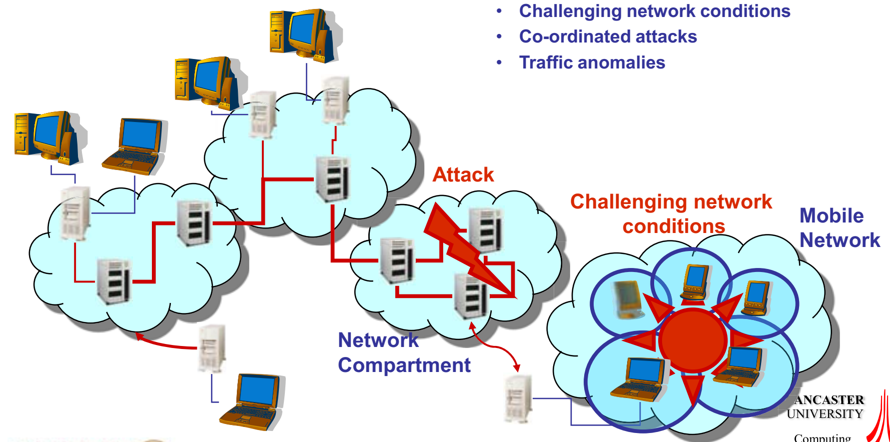


CONTENT

LANCASTER UNIVERSITY
Computing Department

Autonomic Networking: Resilience

- Resilience
  - The ability to tolerate
    - Challenging network conditions
    - Co-ordinated attacks
    - Traffic anomalies



LANCASTER UNIVERSITY Computing Department

CONTENT

Definition

- What is Resilience?

  Resilience is the capability of the network to maintain and acceptable level of service in the face of challenges to normal operation (including legitimate but unusual traffic)
  http://www.comp.lancs.ac.uk/resilience/

  - This includes:
    - Unusual but legitimate traffic load (e.g. flash crowds)
    - High-mobility of nodes and sub-networks
    - Weak, asymmetric, and episodic connectivity of wireless channels
    - Unpredictably long delay paths either due to length (e.g. satellite) or as a result of episodic connectivity
    - Attacks against the network hardware, software, or protocol infrastructure (from recreational crackers, industrial espionage, terrorism, or warfare)
    - Large-scale natural disasters (e.g. hurricanes, earthquakes, ice storms, tsunami, floods)
    - Failures due to mis-configuration or operational errors
    - Natural faults of network components

LANCASTER UNIVERSITY Computing Department

CONTENT

Characteristics of Resilient Networks

- Services provided to the application need to …
  - provide the ability for users and applications to access information when needed, e.g.:
    - Web browsing, distributed database access, sensor monitoring, situational awareness
  - maintain end-to-end communication association, e.g.:
    - collaborative session, video conference, teleconference, etc.
  - support operation of distributed processing and networked storage, e.g.:
    - Ability for distributed processes to communicate with one another
    - Ability for processes to read and write networked storage
- Resilient network services must …
  - remain accessible whenever possible
  - degrade gracefully when necessary
  - ensure correctness of operation, even if performance is degraded
  - rapidly and automatically recover from degradation
- Resilient networks are engineered to …
  - resist challenges to normal operation
  - recognise when challenges and attacks occur and isolate their effects
  - ensure resilience in the face of dependence of other infrastructure such as the power grid
  - rapidly and *autonomically* recover to normal operation
  - refine future behaviour to better resist, recognise, and recover

LANCASTER UNIVERSITY Computing Department

CONTENT

Relationships to other Concepts

- *Survivability*:
  - Is the capability of the system to fulfil the mission in a timely manner, even in the presence of attacks or failures
  - *Fault tolerance*
    - Is the ability of a system or component to continue normal operation despite the presence of hardware or software faults
      - Fault tolerant systems are engineered only to tolerate isolated random natural failures.
      - Fault tolerance is necessary but not sufficient for survivability
- *Disruption tolerance*
  - Is the ability for end-to-end applications to operate even when network connectivity is not strong (weak, episodic, or asymmetric) and the network is unable to provide stable end-to-end paths.

➔ Survivability and disruption tolerance are necessary but not sufficient for resilience?!?

LANCASTER UNIVERSITY Computing Department

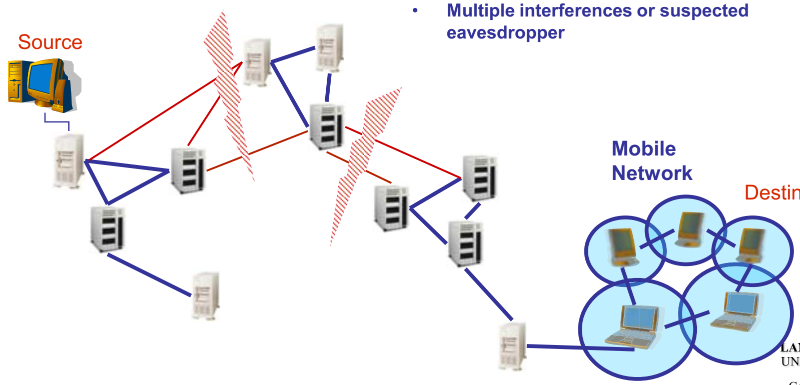
CONTENT

Survivable Communication

- Exploitation on (local) knowledge
  - Opportunistic behaviour
    - Transfer data when links are available and nodes are reachable
    - Epidemic routing protocols
    - To consider:
      - Probability of delivery: scoped and schedule routing if possible
        - » Reduce load while maintaining probability of delivery
        - » Reduce load while maintaining 'goodput'
  - Exert control
    - On node and subnetwork movements
    - Protocol and parameter choices
- Adjust data transfer to environment
  - Cut-through when stable path is available
    - Traditional physical layer techniques
    - Low-latency for capable nodes
  - Store-and-forward
    - Immediate transfer when links become available
  - Store-and-forward with scheduled transfer
    - Wait until link becomes available
    - Controlled transfer
  - Store-and-haul data

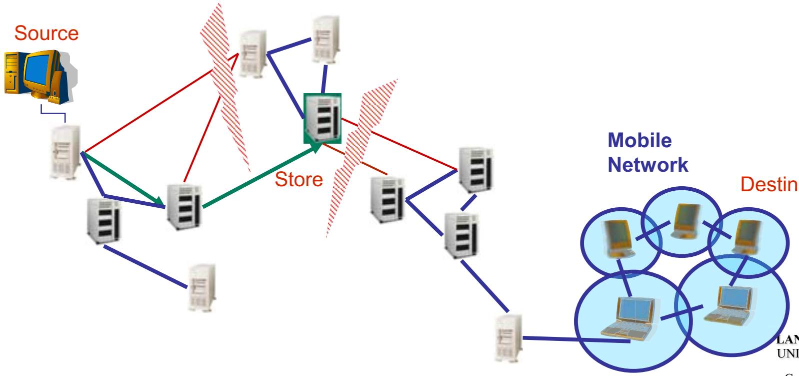
Exploiting Mobility (I)

- **Position nodes exploit mobile nodes**
  - **Exert control on movements of other nodes**
  - **Mobile nodes can carry data as they move**
    - **Store-and-haul data without radiating transmission**
    - **Transit areas of no channel connectivity**
- **Multiple interferences or suspected eavesdropper**



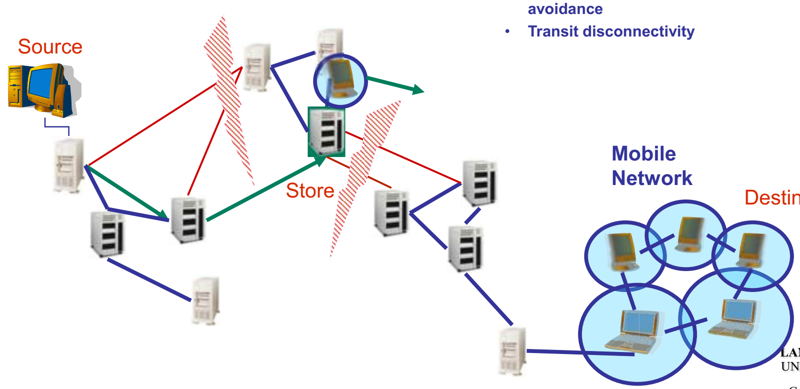
Exploiting Mobility (II)

- **Multiple interferences or suspected eavesdropper**
  - **Solution I: move node or steer antenna around interference**



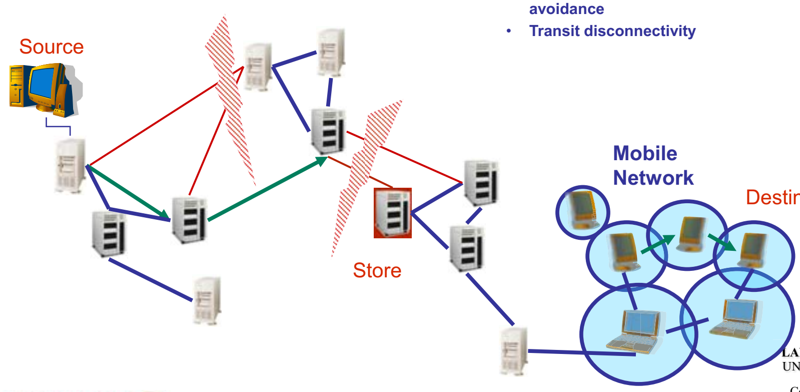
Exploiting Mobility (III)

- **Multiple interferences or suspected eavesdropper**
  - **Solution I: move node or steer antenna around interference**
  - **Solution II: Mobile nodes haul data**
    - **Interference and adversary node avoidance**
    - **Transit disconnectivity**



Exploiting Mobility (III)

- **Multiple interferences or suspected eavesdropper**
  - **Solution I: move node or steer antenna around interference**
  - **Solution II: Mobile nodes haul data**
    - **Interference and adversary node avoidance**
    - **Transit disconnectivity**



LANCASTER UNIVERSITY
Computing Department

Example: PROPHET Protocol

Probabilistic ROuting Protocol using History of Encounters and Transitivity.

- Assumptions
  - Predictable movement patterns
    - Nodes that visit locations repeatedly are likely to do so in the future
  - Bandwidth and storage space are limited resources
- Idea
  - Each node `a` maintains a delivery predictability metric
    - $P_{(a,b)} \in [0,1]$, for all other network nodes `b`
      ➔ represents the probability of two nodes being linked
    - Metric is used when deciding about the forwarding of messages
  - When two nodes meet they
    - Exchange summary vector and delivery predictability vector
    - Update the internal delivery predictabilities
    - Exchange actual messages
      - Based on forwarding strategy

LANCASTER UNIVERSITY
Computing Department

PROPHET Probability Metric

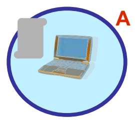
- Update of probability metric on node encounter
  - $P_{(a,b)} = P_{(a,b)old} + (1 - P_{(a,b)old}) * P_{init}$
    - is initialisation constant, $(0,1]$
- Value decreases with age
  - $P_{(a,b)} = P_{(a,b)old} * \gamma^k$
    - γ is the aging constant,
    - k = number of time units since the metric was aged last
- Transitivity property
  - $P_{(a,c)} = P_{(a,c)old} + (1 - P_{(a,c)old}) * P_{(a,b)} * P_{(b,c)} * \beta$
    - β is a scaling constant the determines the impact on the delivery predictability, $[0,1]$
    - Idea
      - If node `a` frequently encounters node `b` and node be frequently encounters node `c` than node `b` is a good node to forward messages fro node `c` to
- Forwarding strategy
  - Forward message to all encountered node with higher P-value for any given destination

A. Lindgren, A. Doria, Olov Schelén: “Probabilistic Routing in Intermittently Connected Networks”, Proceedings of 1st international Workshop on Service Assurance with Partial and Intermittent Ressources (SAPIR2004), August 2004

LANCASTER UNIVERSITY
Computing Department

Prophet Example (III)

Message for D

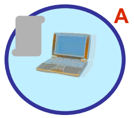


	A	B	C	D
A		Low	Low	Low
B	Low		Low	Low
C	Low	Low		High
D	Low	Low	High	

LANCASTER UNIVERSITY
Computing Department

Prophet Example (II)

Message for D

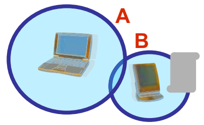


	A	B	C	D
A		Low	Low	Low
B	Low		High	Low
C	Low	High		High
D	Low	Medium	High	

CONTENT
LANCASTER UNIVERSITY
Computing Department

Prophet Example (III)

Message for D



	A	B	C	D
A		High	Low	Low
B	High		High	Low
C	Medium	High		High
D	Low++	Medium	High	

CONTENT
LANCASTER UNIVERSITY
Computing Department

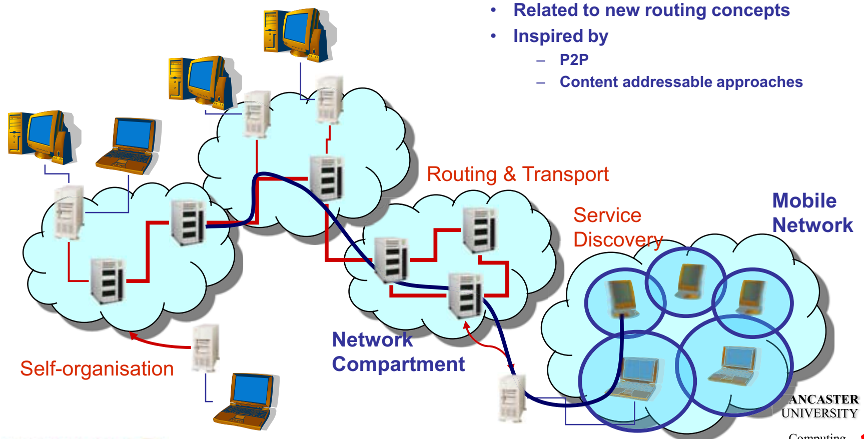
PROPHET Summary

- PROPHET has been compared to Epidemic routing
  - Simulation
    - Set-up
      - Community mobility with 5 nodes per community
      - 3000 seconds message generation, 8000 seconds for delivery
    - Metrics
      - Message delivery ability
      - Message delivery delay
      - Messages exchanges
  - Outperforms Epidemic routing for different queue sizes, hope counts and transmission ranges
- PROPHET relies on re-occurring movement and behaviour
  - Scenarios
    - Saami population of reindeer herders on the move
    - Remote villages in India and Cambodia
    - Military applications

CONTENT
LANCASTER UNIVERSITY
Computing Department

Autonomic Networking: Service Discovery

- **Service Discovery**
  - **Distributed algorithms**
    - **Related to new routing concepts**
    - **Inspired by**
      - **P2P**
      - **Content addressable approaches**



CONTENT
LANCASTER UNIVERSITY
Computing Department

Active Components & Services

- What are active components, active services?
  - Active services provide functionality beyond traditional communication tasks
    - For applications or other services
    - Enhance communication
  - They are programmable services
  - Examples are:
    - Transcoding and content adaptation services
    - Protocol translators and protocol boosters
    - Context aware communication services
      - Adaptation of communication to user context
    - Network data aggregation services
      - e.g. for attack detection systems
- Where are they?
  - Dynamically deployable at specific location throughout the network
    - At network nodes or adjacent to network nodes
    - On programmable platforms
      - Assuming system resource

➔ Functional components in Autonomic Networks can be represented as Active Components

Service Discovery

- Active service points need to be discovered to decide where to plant a service
  - Value-adding services need to be in a position where they are most frequented
    - Cost-benefit ration needs to be positive
  - Required services need to be planted at strategic locations to allow communication between network compartments
    - e.g. protocol translators
- Services need to be discovered to decide if they are in or close to the data path
  - Re-routing of traffic to use certain services
- Capabilities need to be discovered to allow communication in heterogeneous networks
  - In autonomic networks a protocol stack is a set of active components
  - A component might download a capability to be able to become part of a compartment

➔ Critical function within active networks

Service Types

- Variable service location/path – Fixed services/functions
  - Service can be placed at different locations in different data paths
  - Service function is well-defined and fixed
  - e.g. protocol translation service
- Fixed service location/path – Variable services/functions
  - Service is at a specific location(s) respectively part of clearly determined data paths
  - Service function might change
  - e.g. content adaptation service between network compartments
- Fixed service location/path – Fixed services/functions (but I need to discover them)
  - Well defined services at specific locations
  - e.g. network data aggregation service
- Variable service location/path – Variable service functions
  - Not very common

Summary

- What we covered
  - Characterisation of Autonomic Networking
  - View points and basic concepts
    - Networking View
    - Autonomic networking abstraction
    - Communication
    - Resilience
    - Service discovery
- What has not been cover
  - Biological, genetic, social and other concepts related to autonomous behaviour
  - Autonomous computing concepts
    - Emergence, etc.
- What we should have achieved
  - To develop an understanding for the ideas and background of Autonomic Networking
  - To know about concepts and some specific mechanisms underpinning the Autonomic Networking idea
- What is left
  - To proof the feasibility of Autonomic Networking concepts
    - Fully develop mechanisms
    - Benchmark the resulting system(s)

Reading List

- Primary Reading
  - Articles
    - * ANA Project: "ANA Blueprint - 1st Version", Deliverable D1.4/5/6v1, 15th February 2007
    - *A. Lindgren, A. Doria, Olov Schelén: "Probabilistic Routing in Intermittently Connected Networks", Proceedings of 1st international Workshop on Service Assurance with Partial and Intermittent Ressources (SAPIR2004), August 2004
  - Web-Sites
    - ***A***utonomic ***N***etwork ***A***rchitecture: http://www.ana-project.org/
    - Resilience: http://www.comp.lancs.ac.uk/resilience/
- Secondary Reading
  - Articles
    - *S. Schmid, M. Sifalakis, D. Hutchison: "Towards Autonoic Networks", 2006
    - *M. Siekkinen, V. Goebel, T. Plagemann, K.-A.- Skevik, M. Banfield, I. Brusic: "Beyond the Future Internet – Requirements of Autonomic Networking Architectures to Address Long Term Future Networking Challenges,
    - *C. Jelger, C. Tschuding, S. Schmid, G. Leduc: "Basic Abstractions for an Autonomic Network Architecture",
    - *R. Braden, D. Clark, S. Shenker, and J. Wroclawski. "Developing a Next-Generation Internet Architecture". July 2000
    - *D. Clark, K. Sollins, J. Wroclawski, T. Fader. "Addressing Reality: An architectural Response to Real-World Demands on the Evolving Internet". ACM SIGCOMM 2003.
    - *Tony McGregor. "IP Measurement Protocol (IPMP)". Internet draft. draft-mcgregor-ipmp-04
    - *S. Shalunov, B. Teitelbaum "One-way Active Measurement Protocol ". RFC 3763
    - *A. Vahdat, D. Becker: "Epidemic Routing in Partially-Connected Ad-Hoc Networks", Technical Report CS-2000, Internet Systems and Storage Group, Duke University, http://issy.cs.duke.edu/epidemic/epidemic.pdf
  - Web-Sites
    - BIONETS: http://www.bionets.org/
    - CASCADAS: http://www.cascadas-project.org/
    - Haggle: http://www.haggleproject.org/
    - CATNETS: http://www.iw.uni-karlsruhe.de/catnets/
    - NSF-GINI: http://www.nsf.gov/cise/geni/
    - NSF-FIND: http://find.isi.edu/

Thank You!