# Various Faces of Data Centric Networking

*Eiko Yoneki*

*University of Cambridge Computer Laboratory*

## Data Centric Networking

- Shift of communication paradigm
  - From end-to-end to data centric
  - Data as communication token
  - Multipoint communication (Anycast and Multicast)
- Integration of complex data processing with networking
  - A key vision for future computing
  - A huge number of data sources and high volume of data accessible to applications
  - Process data locally before moving over the networks
  - Use power of parallel processing in programming

2

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

## 5 Faces in DCN

1. Content-Centric Networking (CCN) and Content Distribution Networks (CDN)
2. Programming in Data Centric Environment
3. Stream Data Processing and Data/Query Model
4. Graph Structured Data: Network, Storage, and Query Processing
5. Network holds Data in Delay Tolerant Networks (DTN)

3

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

5 Faces in DCN

1. Content-Centric Networking (CCN) and Content Distribution Networks (CDN)
2. Programming in Data Centric Environment
3. Stream Data Processing and Data/Query Model
4. Graph Structured Data: Network, Storage, and Query Processing
5. Network holds Data in Delay Tolerant Networks (DTN)

4

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

## *Shift to Content Based Networking*

- Original Internet
  - 70s technology, conversational pipes, end-to-end
- Now, Internet use (>90%):
  - Content retrieval & Service access
  - Request & Delivery of *named data* - access content
- Shift to a content-centric view:
  - end-to-data
  - Content-awareness and massive storage
  - Source becomes less important – content itself matters
  - Existing approach – e.g. Publish/Subscribe overlay
- Efficiently handle high volume of information
  - No standard way to find and get *nearest* copy
  - Intelligent distribution of information (e.g. capacity, latency)
  - Understand semantic locality of data
  - Include content inspection, filtering…aggregation

5

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

## *Multi-Point Communication*

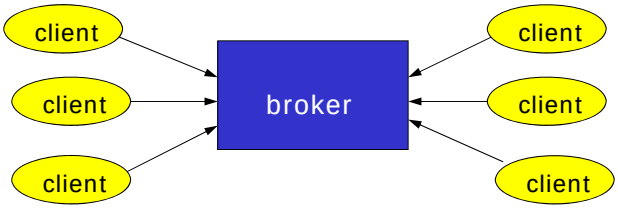
- Application level multicast
  - IP multicast is not supported well over wide area networks
  - Use DHT (Distributed Hashing Table)
  - Use tree routing in order to get logarithmic scaling
  - Bayeux/Tapestry and CAN
  - Service model of multicast is less powerful than content-based messaging system
- Research prototypes of messaging systems
  - Scribe (Topic-based system using DHT over Pastry)
  - SIENA (Content-based distributed event service)
  - JEDI (Content-based messaging system)
  - Gryphon (Topic/content-based message brokering system)

6

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

# Content Based Networking

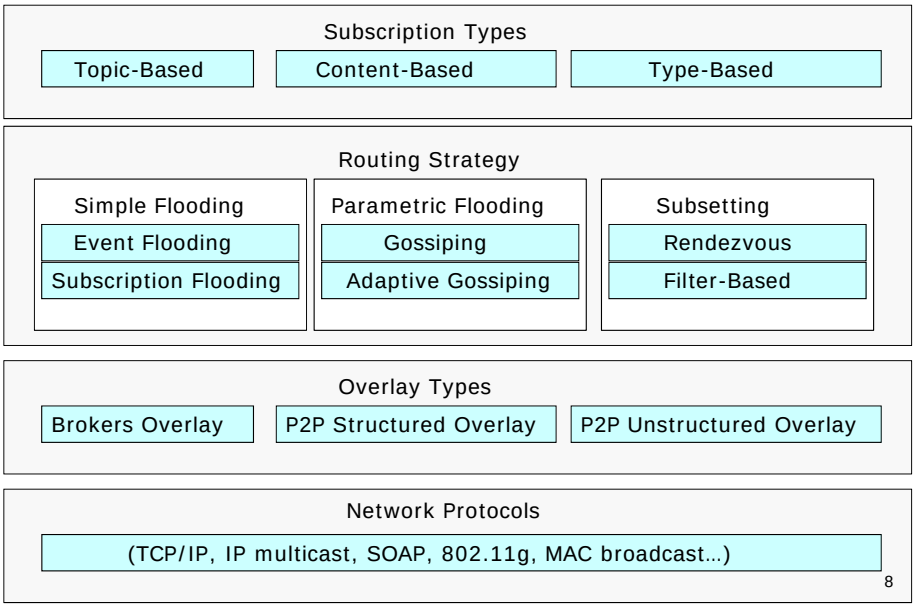
- Publish/Subscribe Paradigm
- Subscription model:
  - Topic-based (Channel)
    - Topics can be in hierarchies but not with several super topics
  - Content-based
    - Express interests as a query over the contents of data
    - How to turn subscriptions into routing mechanism in decentralised environments?






UNIVERSITY OF CAMBRIDGE
Computer Laboratory

# Publish/Subscribe Overlay Architecture



UNIVERSITY OF CAMBRIDGE
Computer Laboratory

## Content Distribution Networks

- Cache of data at various points in a network
- Content served closer to client→Edge Caching
  - Less latency, better performance
- Load spread over multiple distributed systems
  - Robust (to ISP failure)
  - Handle flashes better (load spread)
- Limitation
  - No mechanism with dynamic/personalized content, while more content is becoming dynamic
  - Difficult to manage content lifetimes and cache performance, dynamic cache invalidation
- CDN Providers
  - Coral Content Distribution Network
  - Akamai
  - BitTorrent
  - …



UNIVERSITY OF CAMBRIDGE
Computer Laboratory

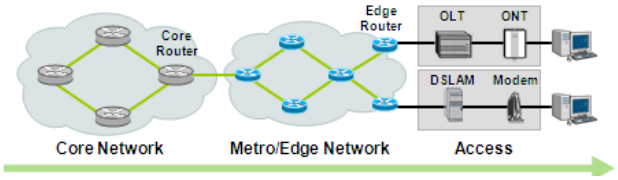
## Content Routing Principle

Pushing content closer to the users

Content from content servers nearer to the client

CDN Strategies:

- Limelight — placing CDN servers near a small # of ISP core nets
- Akamai — placing CDN servers deep into a large # of ISP networks' sites
- Nano Data Center (NaDa) — home gateways (STBs/modems) as CDN servers
  (peer-to-peer delivery among NaDa servers)



UNIVERSITY OF CAMBRIDGE
Computer Laboratory

## CCN (NDN)

- Content-Centric Networking (CCN), Named Data Networking (NDN)
- To networking that enables networks to self-organize and push relevant content where needed

  → From CDNs to native *Content Networks*
- Goals:
  - Remove the need to make DNS lookups
    - New naming system for services and data
    - Place the name lookup scheme in the network
  - Route to one of many possible service
  - Instances
    - Any-cast routing to a service instance
    - Find closest instance
  - Allow for service instances to move locations
  - Allow for self-certifying name



UNIVERSITY OF CAMBRIDGE
Computer Laboratory

## Goals of CCN

- Network delivers content from closest location
- Integrates a variety of transport mechanisms
- Integrated caching (short-term memory)
- Search for related information
- Verify authenticity and control access

## Existing Related Projects

- Next generation Internet proposals:
  - LNA, TRIAD, NIRA, ROFL, i3, DONA
- Van Jacobsen's CCN and NDN
- PSIRP (Publish/Subscribe Internet Routing Paradigm)
- 4WARD - Architecture and Design for the Future Internet
  - NetInf

...and...

- Traditional Publish/Subscribe Systems, P2P and sensor networks

13

## Related Open Source Projects

- ***CCN*** http://www.ccnx.org/ (http://www.named-data.net/)
- ***SIENA*** http://www.inf.usi.ch/carzaniga/cbn/
- ***Scribe*** http://research.microsoft.com/en-us/um/people/antr/overlays/overlays.htm
- ***CORAL*** http://www.coralcdn.org/
- ***Globule: an Open-Source Content Distribution Network*** http://www.globule.org/
- ***XML Blaster: Open Source XML event encoding with XPath expression subscription*** http://www.xmlblaster.org/

14

UNIVERSITY OF CAMBRIDGE Computer Laboratory

## *5 Faces in DCN*

1. Content-Centric Networking (CCN) and Content Distribution Networks (CDN)
2. Programming in Data Centric Environment
3. Stream Data Processing and Data/Query Model
4. Graph Structured Data: Network, Storage, and Query Processing
5. Network holds Data in Delay Tolerant Networks (DTN)



UNIVERSITY OF CAMBRIDGE Computer Laboratory

## *Programming in Data Centric Environment*

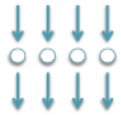
- Data Centre and Cloud environments
  - Applications = a service
  - Platform = a service (e.g. Google AppEngine, MS Azure)
  - Infrastructure = a Service (e.g. Amazon EC2)
  - Challenges:
    - Programming Model (exposure of concurrency, parallelism) and its implementation
    - Physical architecture (new communication protocols, structures)
    - High volume (e.g. billions of entities and terabytes of data) of data management in cloud infrastructure → Data oriented perspective
- Network meets data flow programming

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

## Cloud Programming Model

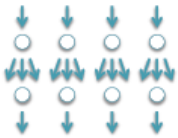
**Batch computing: Condor, Grid Engine, Amazon SQS**

- Programming Model: Relatively easy, but restricted
- Challenges: Scheduling, Load Balancing, Fault Tolerance
- Resources: Sufficient local memory & cores, fast file system



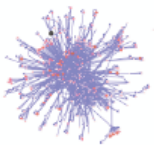
**Loosely coupled: Hadoop, Dryad, Amazon EMR**

- Programming Model: More complicated, more expressive
- Challenges: Parallel Communication



**Tightly coupled: MPI, Pregel, Hadoop**

- Programming Model: Most complicated, most expressive
- Challenges: Parallel Algorithms
- Resources: High Bandwidth, low latency interconnects



17

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

## Data Flow Programming

- Data parallel programming (e.g. MapReduce, Dryad/LINQ, Skywriting)
- Declarative networking (e.g. P2)
  - Declarative language: “ask for what you want, not how to implement it”
  - Declarative specifications of networks, compiled to distributed dataflows
  - Runtime engine to execute distributed dataflows
  - Adopting a data centric approach to system design and by employing declarative programming languages → simplify distributed programming

18

## *Skywriting*

- JavaScript-like job specification language
  - Supports functional programming
  - Data-dependent control flow
- Distributed execution engine (Ciel)
  - Assigns tasks to devices
  - Publish/subscribe for results





## *Data-Driven Declarative Networking*

- How to program distributed computation?
- Use Declarative Networking
  - Use of Functional Programming
    - Simple/clean semantics, expressive, inherent parallelism
  - Queries/Filer etc. can be expressed as higher-order functions that are applied in a distributed setting

http://www.cl.cam.ac.uk/~ey204/pubs/2009_MOBIHELD.pdf

## Related Open Source Projects

- ***Boom*** https://trac.declarativity.net/
- ***Ciel*** http://www.cl.cam.ac.uk/netos/ciel/
- ***Apache Hadoop*** http://hadoop.apache.org/
- ***DryadLINQ*** http://research.microsoft.com/en-us/projects/dryadling/
- ***MapReduce Online*** http://code.google.com/p/hop/
- ***P2*** http://p2.berkeley.intel-research.net/
- ***Opis*** http://perso.eleves.bretagne.ens-cachan.fr/~dagand/opis/



## 5 Faces in DCN

1. Content-Centric Networking (CCN) and Content Distribution Networks (CDN)
2. Programming in Data Centric Environment
3. Stream Data Processing and Data/Query Model
4. Graph Structured Data: Network, Storage, and Query Processing
5. Network holds Data in Delay Tolerant Networks (DTN)

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

## Stream Data Processing

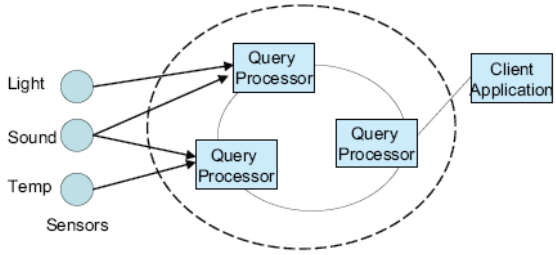
- Stream Data Processing and Data/Query Model
  - Stream: infinite sequence of {tuple, timestamp} pairs
  - Continuous query is result of a continuous query is an unbounded stream, not a finite relation
- Data stream processing emerged from the database community (90's)
- Database systems and Data stream systems
  - Database
    - Mostly static data, ad-hoc one-time queries
    - Store and query
  - Data stream
    - Mostly transient data, continuous queries
- Stream data processing is analogue to Complex Event Processing

23

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

## Sensor Networks and Data Query

- Sensor networks macro-programming
  - State-space, EnviroTrack, Hood, Abstract region
  - Declarative/query: TinyDB
- Data collection: streaming to distributed DB
- Continuous query: Allocation of operators

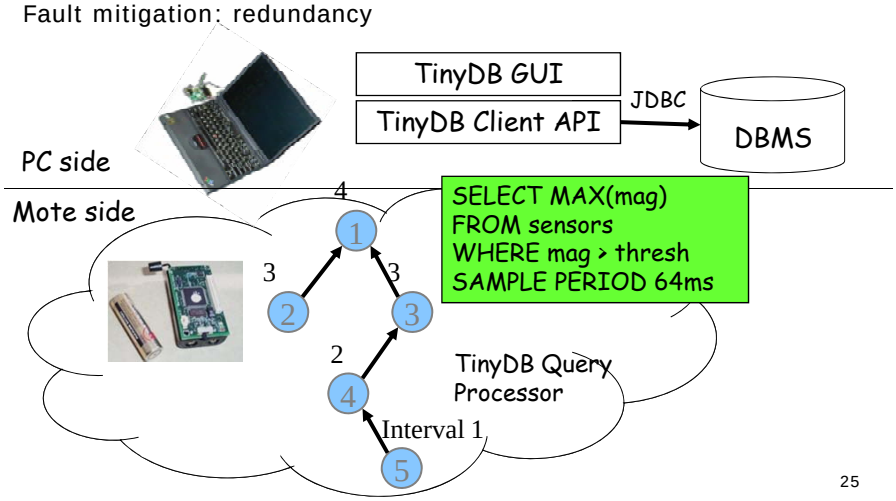



24

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

# TinyDB

- Declarative SQL-like query interface
- Multiple concurrent queries and persistent storage,
- In-network, distributed query processing
- Fault mitigation: redundancy




25

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

# Related Open Source Projects

- ***Borealis*** http://www.cs.brown.edu/research/borealis/public/
- ***Cayuga*** http://www.cs.cornell.edu/bigreddata/cayuga/
- ***STREAMS*** http://infolab.stanford.edu/stream/
- ***TelegraphCQ*** http://telegraph.cs.berkeley.edu/telegraphcq/v0.2/
- ***DSN*** http://db.cs.berkeley.edu/dsn/
- ***TinyDB*** http://telegraph.cs.berkeley.edu/tinydb/software.html
- ***Yahoo scalable streaming query system*** http://www.globule.org/
- ***Flask*** http://www.eecs.harvard.edu/~mainland/projects/flask/

26

UNIVERSITY OF CAMBRIDGE Computer Laboratory

## *5 Faces in DCN*

1. Content-Centric Networking (CCN) and Content Distribution Networks (CDN)
2. Programming in Data Centric Environment
3. Stream Data Processing and Data/Query Model
4. Graph Structured Data: Network, Storage, and Query Processing
5. Network holds Data in Delay Tolerant Networks (DTN)



UNIVERSITY OF CAMBRIDGE Computer Laboratory

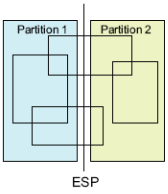
## *Graph Structured Data*

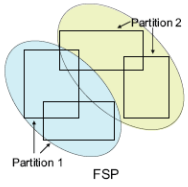
- Increasing demand to store and query data with an inherent graph structure
  - Social networks, Semantic Web, maps
- How to achieve large-scale data processing?
- Understanding semantic locality of data
  - Dynamics of graph topology
  - Social aspect of OSN data
- How to support rapid graph edge traversal?

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

## *Large Scale Data Processing*

- DHT/Key-value stores – scalability by random partitioning of data stores
  - Twitter uses Cassandra
- Semantic locality of data can be exploited for partitioning
  - OSN: Social proximity, Colocation
  - Distributed placement of subgraphs
  - Reduction of network traffic
  - Maximise in-memory processing




29

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

## *Graph Query*

- Traditional relational databases is short on querying graph data
  - Conventional schema (e.g. columns or fields) are required to be predefined thereby limiting flexibility
- Emergence of graph-based database
  - Pregel
  - Neo4j
  - Trinity
  - …

30

## *Network Topology for Network Protocol*

- Build network structure/topology for data dissemination (e.g. overlay construction) for improving performance or reliability
  - What context should be used for building a topology?
  - How to decide next hop (e.g. k random selection)?
- With given network graph/topology, how does data diffuse?
  - Data flow in network graph
  - Based on node capacity
- Understanding graph in networking

31

## *5 Faces in DCN*

1. Content-Centric Networking (CCN) and Content Distribution Networks (CDN)
2. Programming in Data Centric Environment
3. Stream Data Processing and Data/Query Model
4. Graph Structured Data: Network, Storage, and Query Processing
5. Network holds Data in Delay Tolerant Networks (DTN)

32

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

## *Delay Tolerant Networks*

- Delay Tolerant Networks (DTN)
  - Network holds data
  - Path existing over time
  - Store and forward paradigm
- Weak and episodic connectivity - Eventual connectivity
- Non-Internet-like networks
  - Stochastic mobility
  - Periodic/predictable mobility
  - Exotic links
    - Deep space [40+ min RTT; episodic connectivity]
    - Underwater [acoustics: low capacity, high error rates & latencies]
- DTN routing takes place on a time-dependent topology
  - Links come and go, sometimes predictably

33

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

## *Prototypes: Architecture*

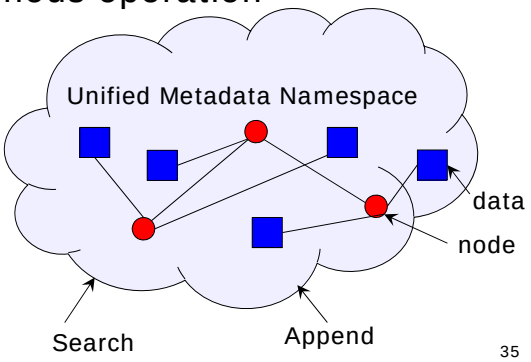
- Providing Connectivity to Developing Countries: DakNet
- Vehicular Communications: DriveThru, DieselNet
- Wildlife Tracking: ZebraNet
- Haggle: Pocket Switched Networks, Social Networking
- DTNRG and the Bundle Protocol (RFC 5050)
  - Mostly an engineering approach to implement the InterPlaNetary Internet

34

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

## Haggle Node Architecture

- Each node maintains a data store: its current view of global namespace
  - Persistence of search: delay tolerance and opportunism
- Semantics of publish/subscribe and an event-driven + asynchronous operation
- Multi-platform
  (written in C++ and C)
  - Windows mobile
  - Mac OS X, iPhone
  - Linux
  - Android




35

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

## Related Open Source Projects

- ***Haggle*** http://code.google.com/p/haggle/, http://www.haggelproject.org
- ***DTN at TKK Comnet*** http://www.netlab.tkk.fi/~jo/dtn/

36

## *5 Faces in DCN*

1. Content-Centric Networking (CCN) and Content Distribution Networks (CDN)
2. Programming in Data Centric Environment
3. Stream Data Processing and Data/Query Model
4. Graph Structured Data: Network, Storage, and Query Processing
5. Network holds Data in Delay Tolerant Networks (DTN)

See you next week!