Delay-Tolerant Information-Centric Networking

Scott Burleigh

IPN Group

25 May 2016

Provided through the courtesy of the Jet Propulsion Laboratory, California Institute of Technology. This research was performed at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. Government sponsorship acknowledged.

Information-Centric Networking

- Today, the Internet is more widely and heavily used for the distribution of content – often to multiple recipients, many of which may be mobile devices – than for any other purpose.
- So it now arguably makes sense to optimize Internet operations for efficient propagation based on the *nature* of content than on its specific current and intended *locations*.
- This general concept is sometimes termed Information-Centric Networking (ICN), and a number of architectures have been proposed for its achievement.

[For the material on the next few slides I am indebted to "A Survey of Information-Centric Networking" by Bengt Ahlgren, Christian Dannewitz, Claudio Imbrenda, Dirk Kutscher, and Börje Ohlman, in *IEEE Communications Magazine*, July 2012.]

DATA-ORIENTED NETWORK ARCHITECTURE (DONA)

- Data sources publish named data objects (NDOs) by registering them (possibly wild-carded, and possibly before the objects exist) with resolution handlers (RHs), a hierarchical infrastructure. Registrations have expiry times.
- Requests for data are routed by name anycast to the closest RH at which the requested NDOs are cached.
- Data are sent back in response, either directly or else through the reverse RH path, enabling caching at nodes other than the sources.

CONTENT-CENTRIC NETWORKING (CCN)

- Data sources publish NDOs by invoking routing protocols that distribute information about NDO location.
- Object names are hierarchical. CCN routers manage Pending Interest Tables (PITs) that enable routing to be aggregated based on these names.
- The routers forward requests for data as necessary and, in the reverse direction, they forward the data requested. In the course of forwarding NDOs they may cache the data, enabling future requests to be serviced more locally and rapidly.

PUBLISH-SUBSCRIBE INTERNET ROUTING PARADIGM (PSIRP)

- Data sources publish NDOs tagged with Scope Identifiers (SIs).
- A data requestor subscribes to an NDO by specifying SI and Rendezvous Identifier (RI).
- The rendezvous system matches SI and RI, resolving them to a Forwarding Identifier (FI) that is sent to the NDO's source.
- The data source then forwards the FI and data to the requestor via the PSIRP routers, guided by a Bloom filter in the FI.

Network of Information (NetInf)

- Data sources publish NDOs either by registering a name/locator binding with a name resolution service (NRS) or by announcing routing information in a routing protocol.
- A data requestor can invoke the name resolution services of an NRS to obtain a set of locators and then retrieve NDOs directly from the indicated source(s).
- Alternatively the requestor can simply issue a request by data name. NetInf routers will then forward the request, using name-based routing, until a source is reached; the source then returns the data to the requestor.

Unifying Concepts

- Data objects are named.
- The name of a data object must somehow enable cryptographic protection of the object's integrity and possibly guarantee its authenticity.
- Data issuance, requisition, routing and forwarding are performed on the basis of object names.
- Data are cached not only at the edges of the network but at nodes in the interior of the network as well.

Advantages

- Scalable and efficient content distribution, thanks to caching and aggregation.
- Preservation of the identity of a data object regardless of the locations at which copies reside at any moment.
- Integrity and authenticity of each data object are preserved not only in transit but also at rest, at each cache point.
- Location independence. Mobility and multi-homing of data sources and requestors is easily accommodated.
- Disruption tolerance: no need to sustain end-to-end connected transport sessions – store/carry/forward works.

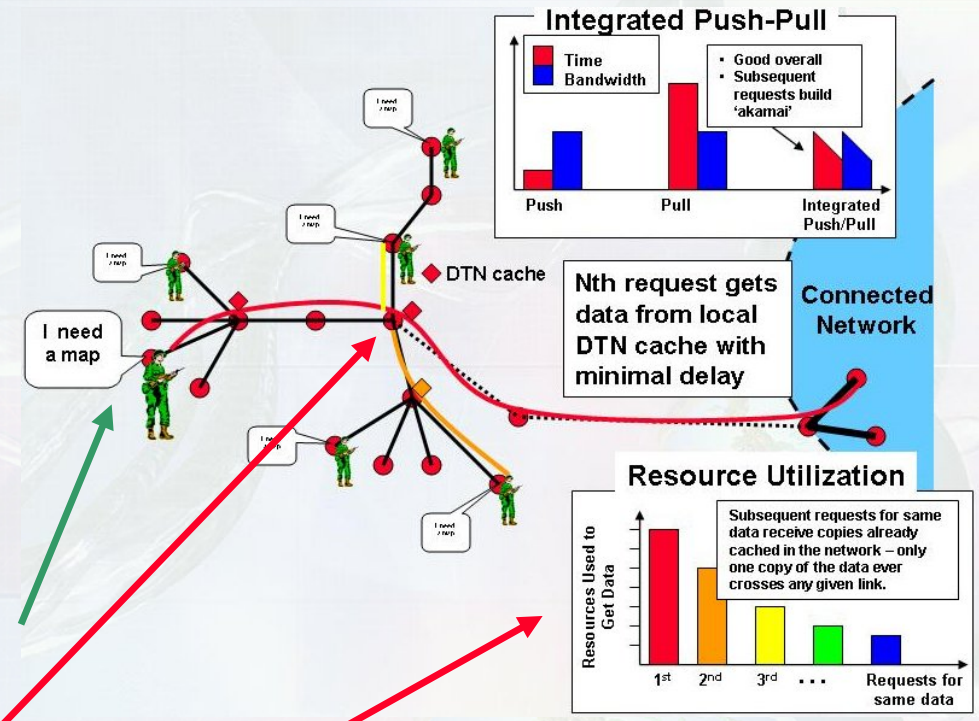
Delay-Tolerant Networking (DTN)

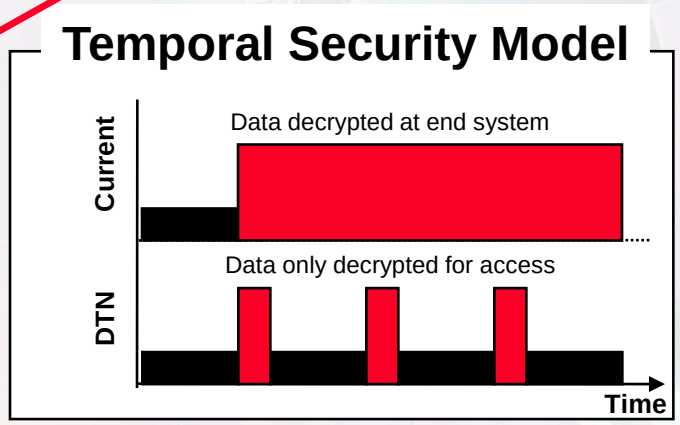
- Some kinds of electronic communication networks are characterized by very long signal propagation latencies (e.g., undersea acoustic links), frequent transient network partitions (e.g., smartphones that go out of range of service towers), or both (e.g., deep space mission operations).
- For these networks, the Internet protocols don't work well. The Internet architecture assumes uninterrupted end-to-end connectivity and very brief round-trip communication times.
- Delay-tolerant networking was developed to enable automated network communications to be extended to these problematic networks: RFCs 4838, 5050, and others.

DTN Network Persistence Can Solve Fundamental Internet Application Shortfalls

Right information... Right place... Right time

- **DTN makes applications over disrupted networks robust**
- **DTN is also an *Opportunity* to solve *Fundamental Problems* we've never before had a handle on, using *Network-Managed Persistence***
  - **Access information by content or type rather than by network address**
    - **"I want maps for my area" instead of "I want to ftp to 192.168.4.17"**
  - **Retrieve once, provide to local users as requested**
  - **Learn from actual network usage**
  - **Exploit in-network storage/caches and pub/sub protocols to create a dynamic and self-forming "Akamai"**
  - **Use *temporal* security rather than *physical* security**





(Preston Marshall, 2006)

DTN for ICN

- Proposal: extend DTN slightly to form yet another ICN architecture – DTICN.
- The key difference between DTICN and other ICN architectures:
  - No unique, persistent data object names.
  - Instead, non-unique textual digests characterize the payloads of uniquely identified DTN *bundles*.

How would it work? (1 of 3)

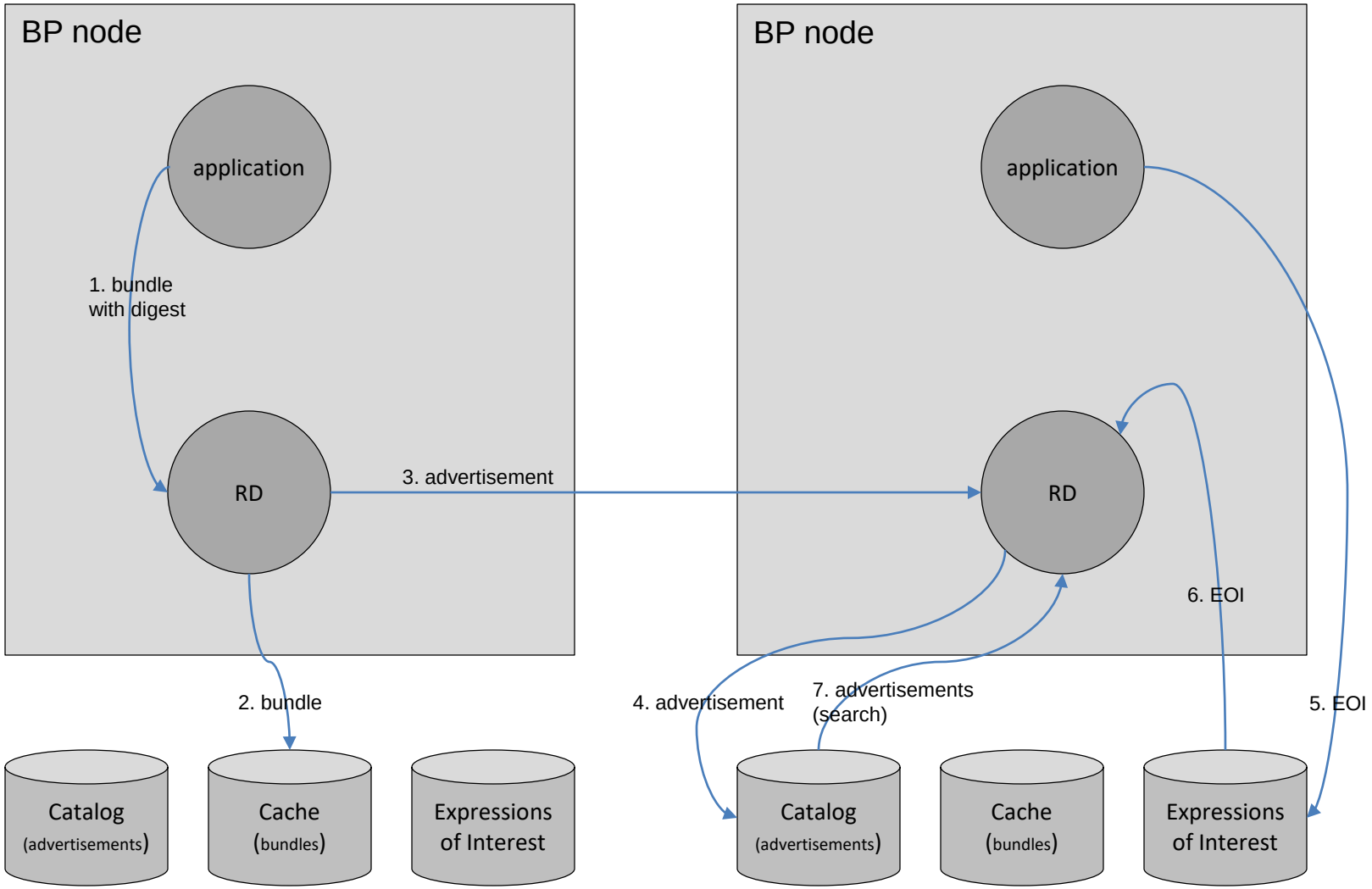
- DTICN bundles have *digest* extension blocks: title, metadata.
- Nodes have local *repository daemons* (RDs) that manage ICN.
- When data source publishes a bundle, a copy always goes to the local RD, which caches it. The RD then publishes an *advertisement* for each such bundle it receives, containing the bundle's digest, payload size, and expiration time and noting the multicast group to join in order to receive a copy.
- Every RD subscribes to advertisements and remembers them in a *catalog*.

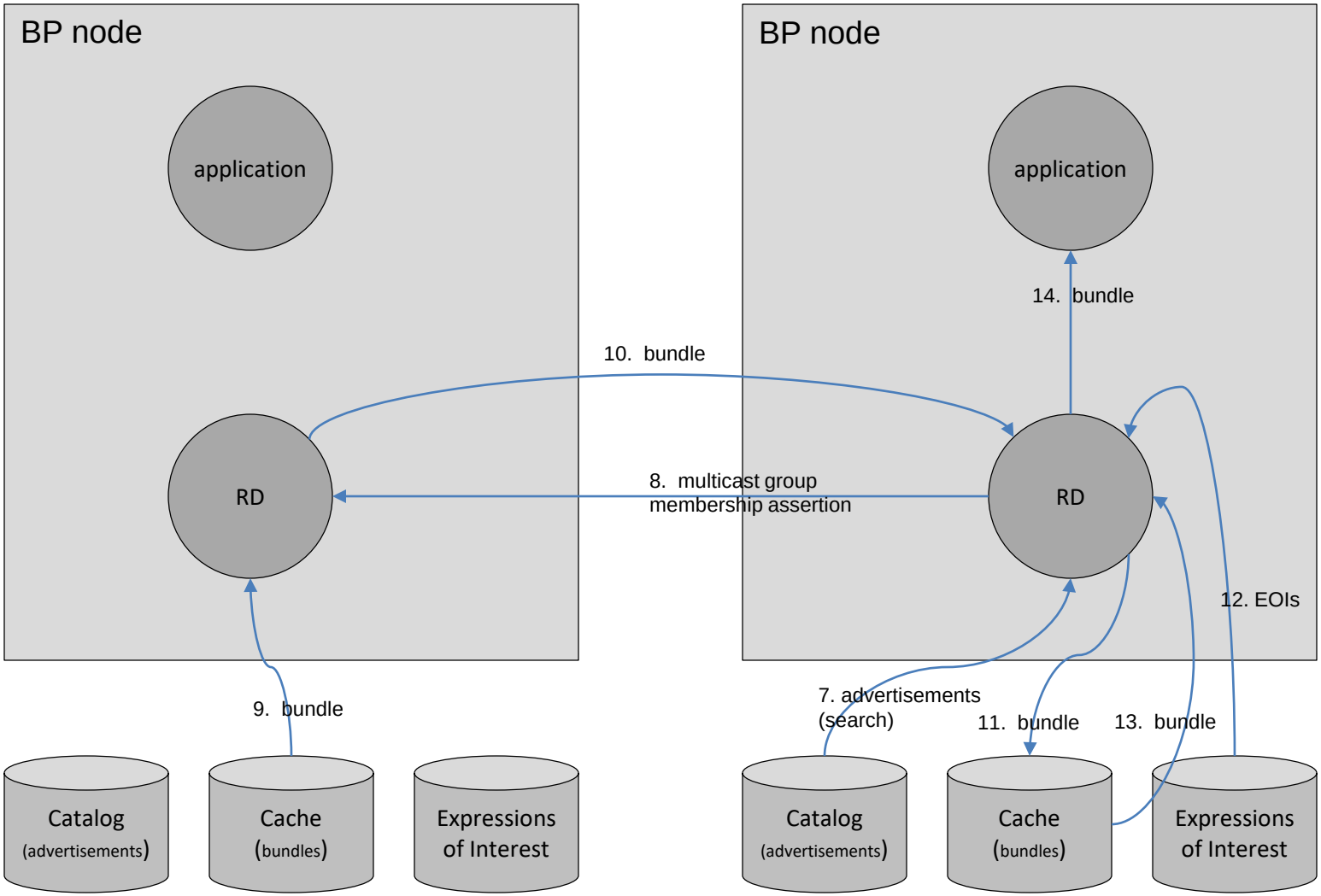
How would it work? (2 of 3)

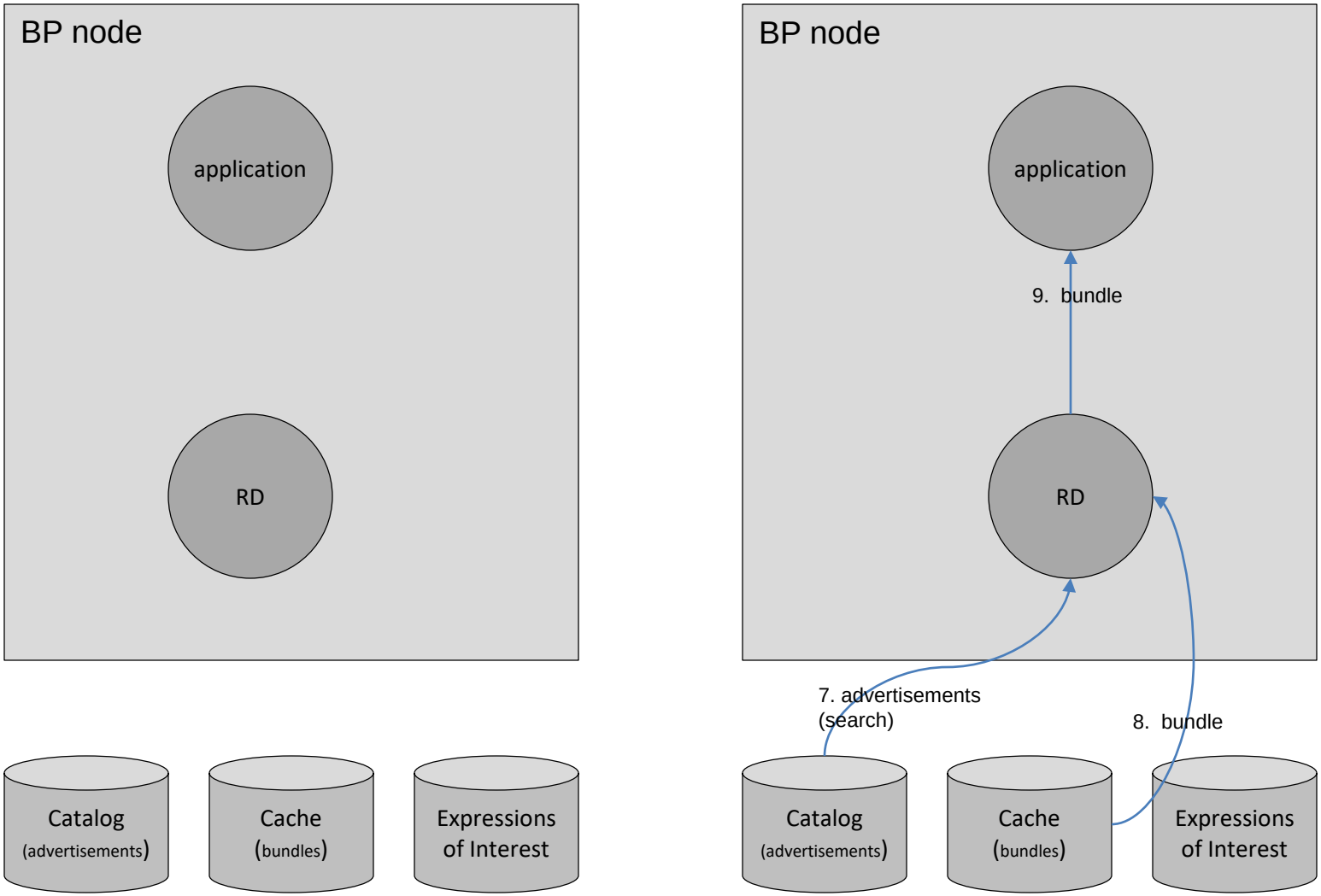
- Data requestors post *expressions of interest* (EOIs, i.e., search phrases) locally.
- RD at each node notes new expressions of interest, uses them to search through the catalog. For each matching advertisement:
  - If the associated bundle is already cached locally, then the RD simply delivers the bundle to the requesting application.
  - Otherwise, the RD obtains the associated bundle from the network.

How would it work? (3 of 3)

- To obtain a bundle from the network:
  - The node joins the associated multicast group noted in the advertisement. This causes multicast membership to be propagated until it reaches a node whose RD has cached the associated bundle.
  - That node's RD forwards a copy of the bundle toward the new group member.
- Whenever a node relays a DTICN bundle, the node's RD caches it.
- Whenever a data bundle is delivered to the RD of a node that is a member of the bundle's multicast group, the RD caches the bundle and delivers it to all current requesting application(s) according to the current list of EOIs.


BP node
application
1. bundle
with digest
RD
3. advertisement
2. bundle
Catalog
(advertisements)
Cache
(bundles)
Expressions
of Interest
BP node
application
RD
6. EOI
4. advertisement
7. advertisements
(search)
5. EOI
Catalog
(advertisements)
Cache
(bundles)
Expressions
of Interest


BP node
application
RD
BP node
application
RD
14. bundle
10. bundle
8. multicast group
membership assertion
12. EOIs
9. bundle
7. advertisements
(search)
11. bundle
13. bundle
Catalog
(advertisements)
Cache
(bundles)
Expressions
of Interest
Catalog
(advertisements)
Cache
(bundles)
Expressions
of Interest


BP node
application
RD
Catalog
(advertisements)
Cache
(bundles)
Expressions
of Interest
BP node
application
9. bundle
RD
7. advertisements
(search)
8. bundle
Catalog
(advertisements)
Cache
(bundles)
Expressions
of Interest

Achieves Key Goals of ICN…

- Repository structure enables scalable and efficient content distribution, thanks to caching and aggregation.
- Bundle Protocol Security preserves integrity and authenticity of each bundle not only in transit but also at rest, at each cache point.
- DTN late binding ensures that mobility and multi-homing of data sources and requestors is easily accommodated.
- DTN is innately disruption-tolerant.

…and Goes a Little Further

- No namespace issues. EOIs are just search phrases.
- All “rendezvous” (i.e., search) activity occurs in the node that is local to the requisitioning consumer.
  - Distributes the computational task as widely as possible.
  - Eliminates some network traffic (the flow of EOIs).
  - Protects the privacy of consumers’ expressions of interest.
  - No transmission of content objects to cache (rendezvous) points that have not yet received any expressions of interest in those objects.
- Selection of cache points is easy and automatic.
- Transmission is reliable.
- No chance of privately transmitted data being published.

Summary

- DTICN is a simple extension to existing DTN architecture that appears to achieve the objectives of ICN.
- No conceptual obstacles to deployment, as DTN is designed for seamless integration with the Internet from the outset.
- 10 years later, maybe we can tick this item off Preston Marshall's checklist.

Questions?