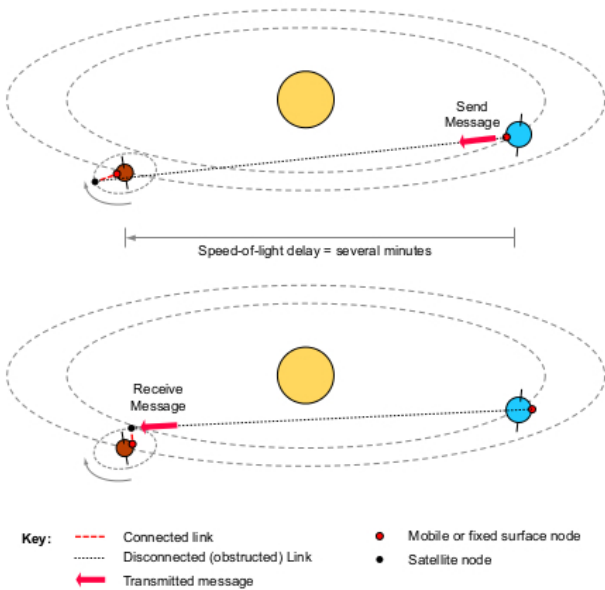




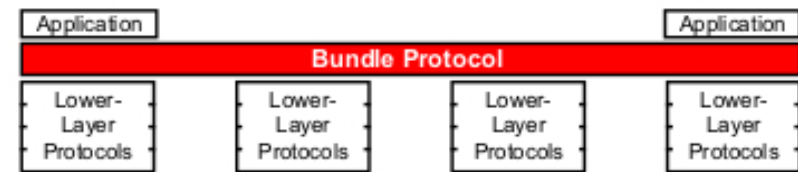


## Scheduled Contacts: An Example

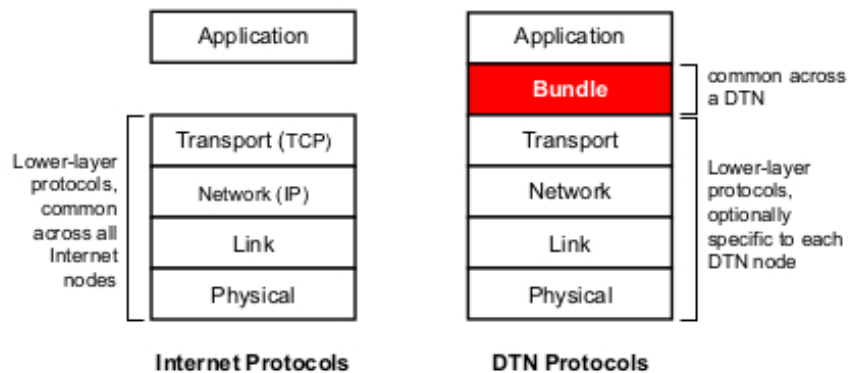


## Main Concepts of Bundle Protocol

- Implements store-and-forward message switching.
- Overlays a new transmission protocol (the bundle protocol) on top of the lower layers (e.g., the Internet protocols).
- Ties together the lower layers so that application programs can communicate across the same or different sets of lower-layer protocols under conditions that involve long network delays or disruptions



## IP stack vs DTN stack



## Bundles

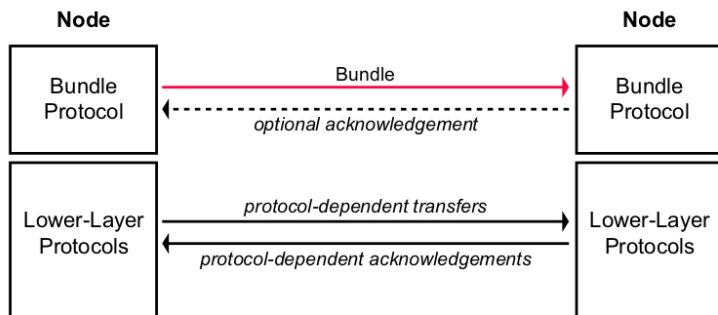
Bundles consist of three things:

- 1 a bundle header consisting of one or more DTN blocks inserted by the bundle-protocol agent,
- 2 a source-applications user data, including control information provided by the source application for the destination application that describes how to process, store, dispose of, and otherwise handle the user data, and
- 3 an optional bundle trailer, consisting of zero or more DTN blocks, inserted by the bundle-protocol agent (not shown in the figure below). Like application-program user data, bundles can be arbitrarily long.



## A Non Conversational Protocol

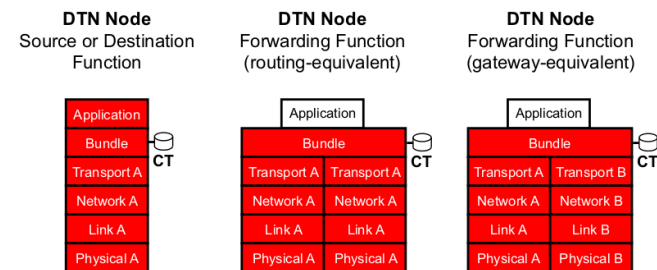
- DTN nodes communicate using simple sessions with minimal or no round-trips.
- Acknowledgements are optional.
- The Lower-layer protocols may be conversational (e.g., like TCP).



## DTN Node Roles

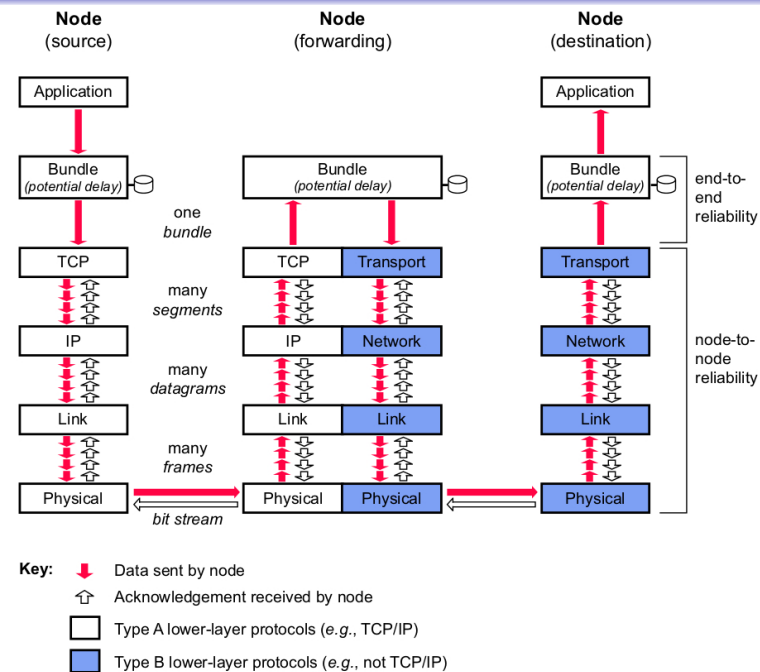
At any moment, a given node may act as a source, destination, or forwarder of bundles:

- **Source or Destination Function**
- **Forwarding Function**
  - 1 Routing-Equivalent Forwarding – implement same lower layer protocols.
  - 2 Gateway-Equivalent Forwarding – implement multiple stacks of lower layer protocols.



## Delay Isolation via Transport-Protocol Termination

- TCP protocol provides end-to-end (source-to-destination) reliability by retransmitting any segment that is not acknowledged
- The network, link, and physical protocols provide other types of data-integrity services.
- The bundle protocol relies on these lower-layer protocols to insure the reliability of communication.
- However, all DTN nodes terminate lower-layer transport protocols.
- **Problem:** The bundle protocol agents thus act as surrogates for end-to-end sources and destinations.
- **Opportunity:** Conversational lower-layer protocols are isolated by the bundle protocol from long delays elsewhere in the end-to-end path.



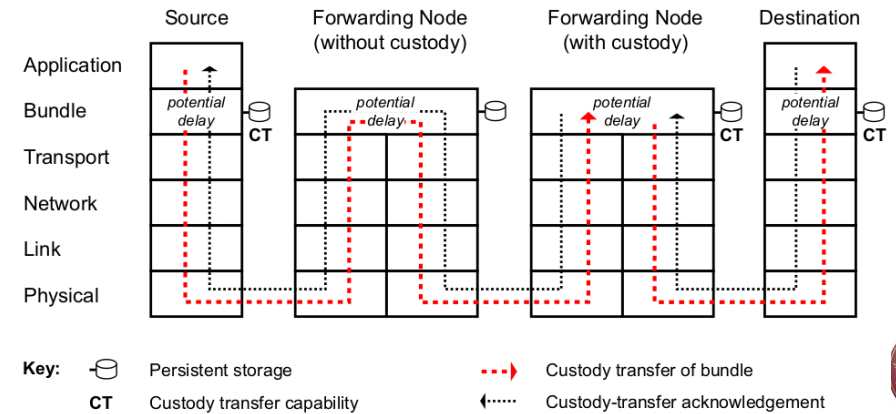
# Custody Transfers

- How to support node-to-node retransmission of lost or corrupt data ?
  - No single transport protocol typically operates end-to-end across a DTN.
  - End-to-end reliability can only be implemented at the bundle layer.
- Support node-to-node retransmission by means of **custody transfers**.
  - Custody transfers enhance end-to-end reliability,
  - but they do not guarantee it.
- Such transfers are arranged between successive nodes.
  - Not all successive nodes need to be custodian.
  - If the next successive node accepts custody, it returns an acknowledgment to the sender.



# Bundle Custodian

- A bundle custodian must store a bundle until either
  - 1 Another node accepts custody, or
  - 2 Expiration of the bundles time-to-live.

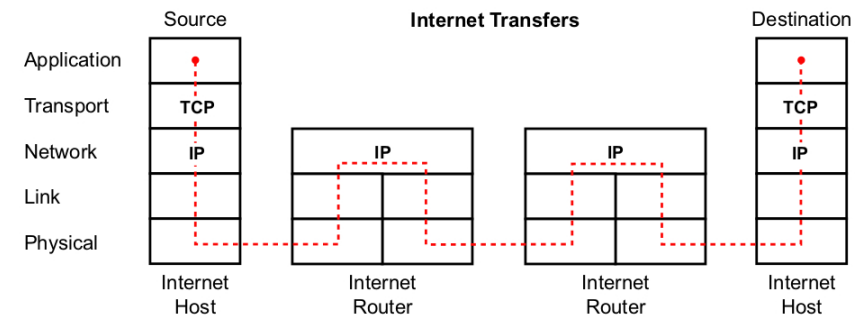


# Internet routing vs DTN routing

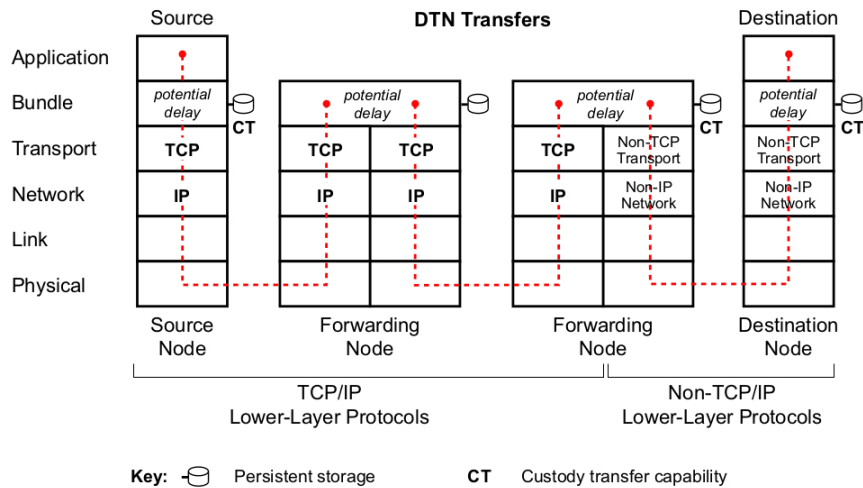
- On the Internet, the TCP and IP protocols are used throughout the network.
  - TCP operates at the end points of a path.
  - TCP manages reliable end-to-end delivery of TCP segments.
  - IP operates at all nodes on the path.
  - IP routes IP datagrams.
- In a DTN, all nodes implement both the bundle protocol and a lower-layer protocols.
  - Nodes that forward bundles can implement either the same or different lower-layer protocols on either side of the forwarding.
  - Nodes functions are comparable to Internet routers or gateways, respectively.



# Internet routing



## DTN routing



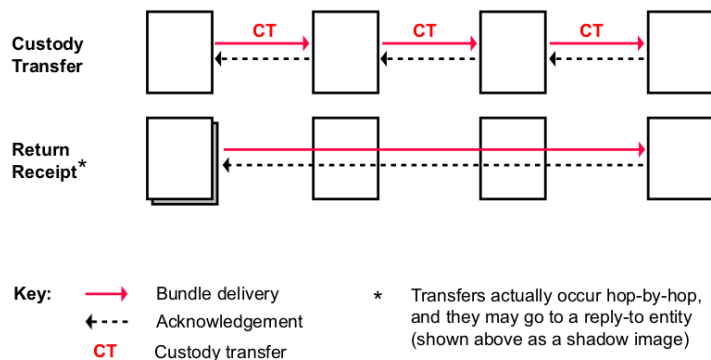
## Classes of Bundle Service

The bundle protocol provides six classes of service for a bundle:

- **Custody Transfer:** Delegation of retransmission responsibility by one node to another accepting node, so that the first node can recover its retransmission resources. The accepting node returns a custodial-acceptance acknowledgement to the previous custodian.
- **Return Receipt:** Confirmation by the destination to the source, or its reply-to entity, that the bundle has been received by the destination application. Reception by the source, or its reply-to entity, of the return receipt provides end-to-end assurance of delivery.
- **Priority of Delivery:** Bulk, Normal, or Expedited.
- **Time-to-Live**



## Classes of Bundle Service



## Endpoint IDs

- A bundle endpoint is a set of zero or more nodes that all identify themselves by the same endpoint ID.
- Common case: only one node has a given endpoint ID – called a **singleton endpoint**.
- Source nodes are always singleton endpoints or null (anonymous source) endpoints.
- Destination nodes may or may not be singleton endpoints.
- Endpoints may also be multicast (multiple destination nodes with the same endpoint ID) or null (no nodes).
- Endpoints may contain multiple nodes.
- Nodes may be members of multiple endpoints.



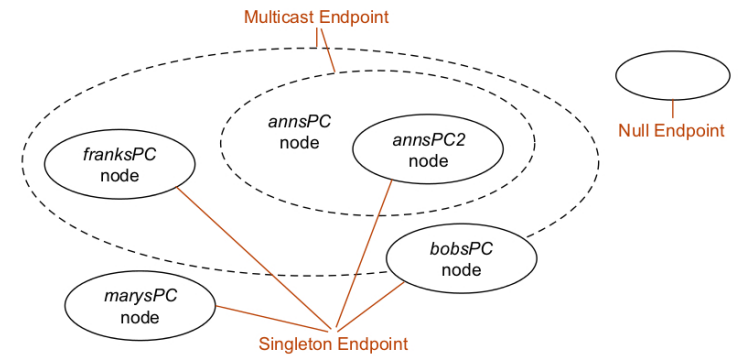
## Endpoint IDs

- An endpoint ID is a uniform resource identifier (URI) text string using the syntax:  
`<scheme_name>:<scheme-specific_part>`
- the scheme name is either `dtm` or `ipn`.
- The scheme-specific part comes in two flavors:
  - 1 Application-specific, used to identify a source or destination node, or
  - 2 Administrative, used when forwarding bundles from node to node.

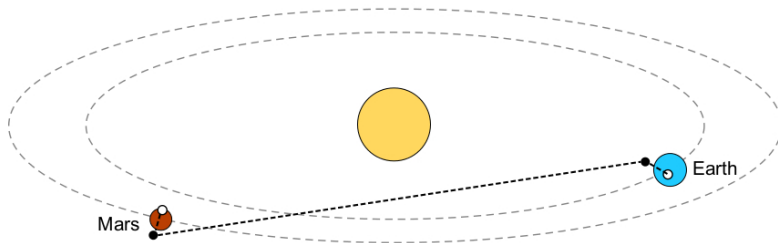
- `dtm://bobsPC/files` (application-specific)
- `dtm://bobsPC/` (administrative)
- `ipn:81.2` (application-specific)
- `ipn:81.0` (administrative)



## Naming Example



## A Simple Example



**Key:** - - - - - IPN link      ○ Source or destination node  
 ● Forwarding node



## A Simple Example

Node	Endpoint IDs	
Earth Source	<code>ipn:81.2</code>	(application-specific ID)
Earth Forwarding	<code>ipn:81.0</code>	(administrative ID)
	<code>ipn:49.0</code>	(administrative ID)
Mars Forwarding	<code>ipn:49.0</code>	(administrative ID)
	<code>ipn:65.0</code>	(administrative ID)
Mars Destination	<code>ipn:65.7</code>	(application-specific ID)







