Practical and Secure Accounting in Content-Centric Networking

Cesar Ghali
Gene Tsudik
Christopher A. Wood
Edmund Yeh

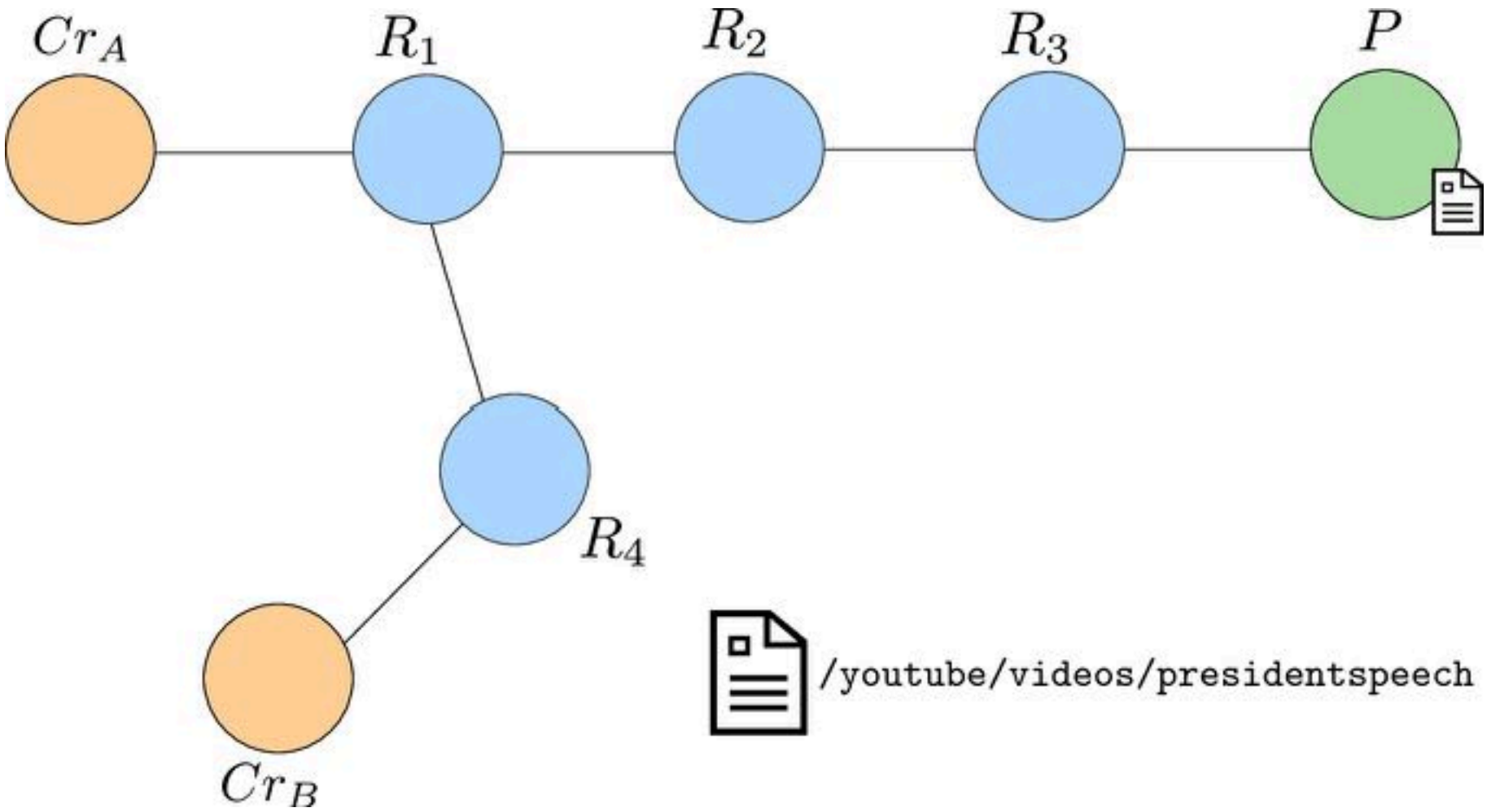
Agenda

- CCN Overview
- Accounting Information
  - Individual, distinct, and aggregate
  - Cache hits vs content requests
- Accounting techniques
  - Encryption
  - Push Interests
    - Format and processing steps
- Secure accounting
  - Only aggregate and distinct are possible!
- Recommendations
  - Include nonce
- Experimental analysis
  - Show overhead for two scenarios

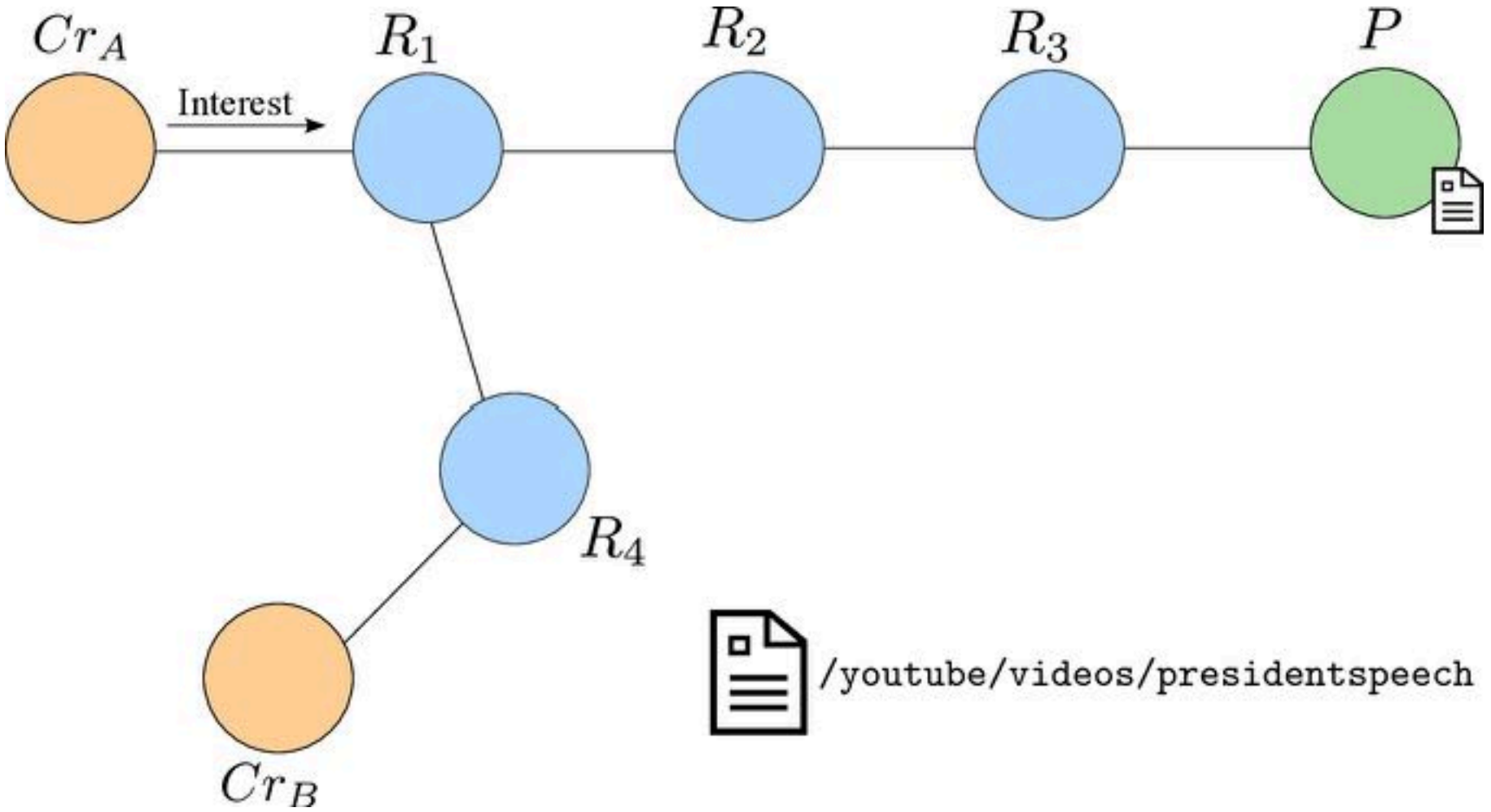
CCN/NDN Overview

- CCN and NDN are network architectures for transferring **named data**
  - Data is obtained via an explicit request for the name with an **interest**
  - **Consumers** issue interests are routed towards the data **producer** (using the name)
  - A **content object** carries the data back to the consumer

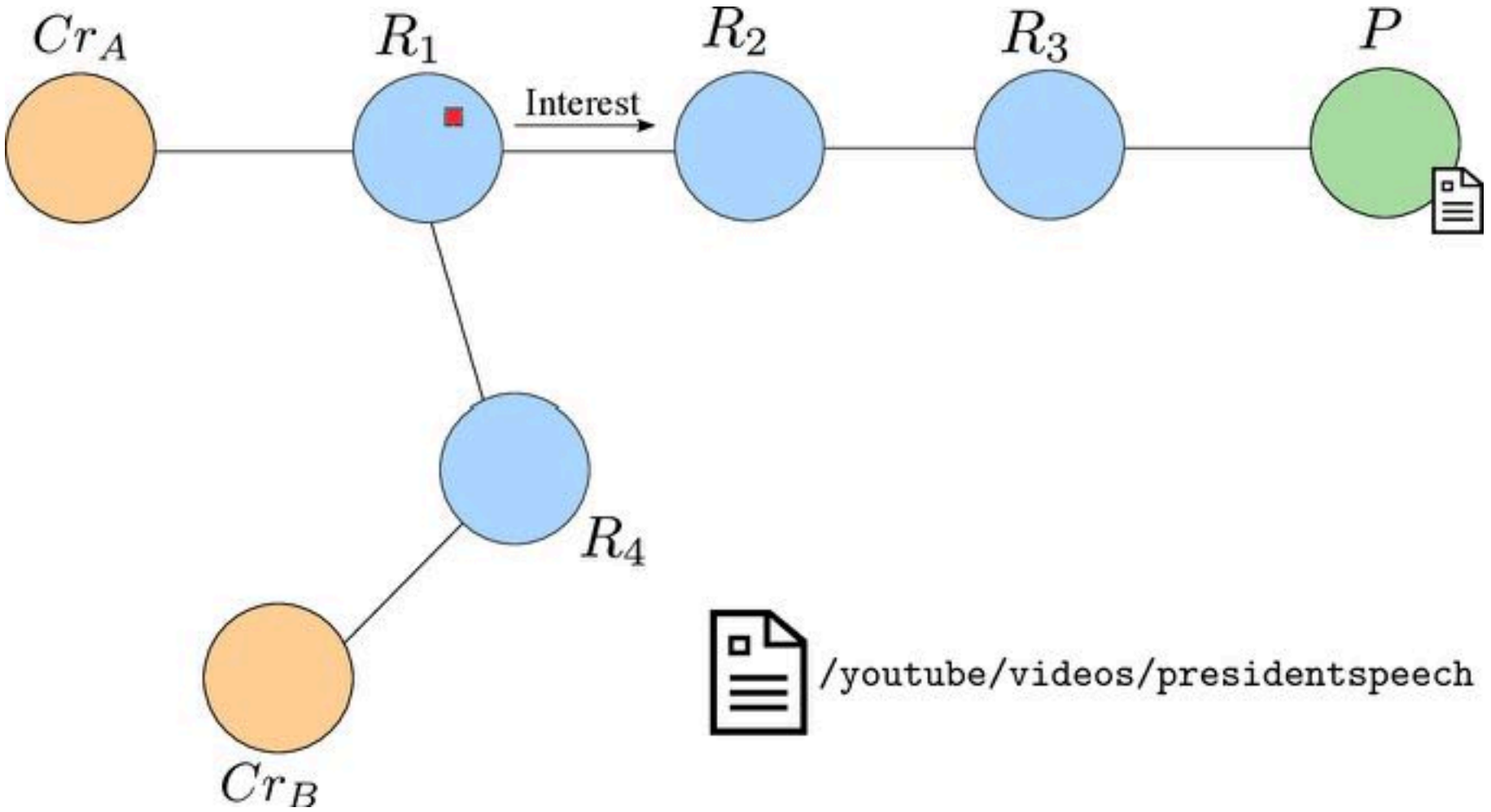
Example



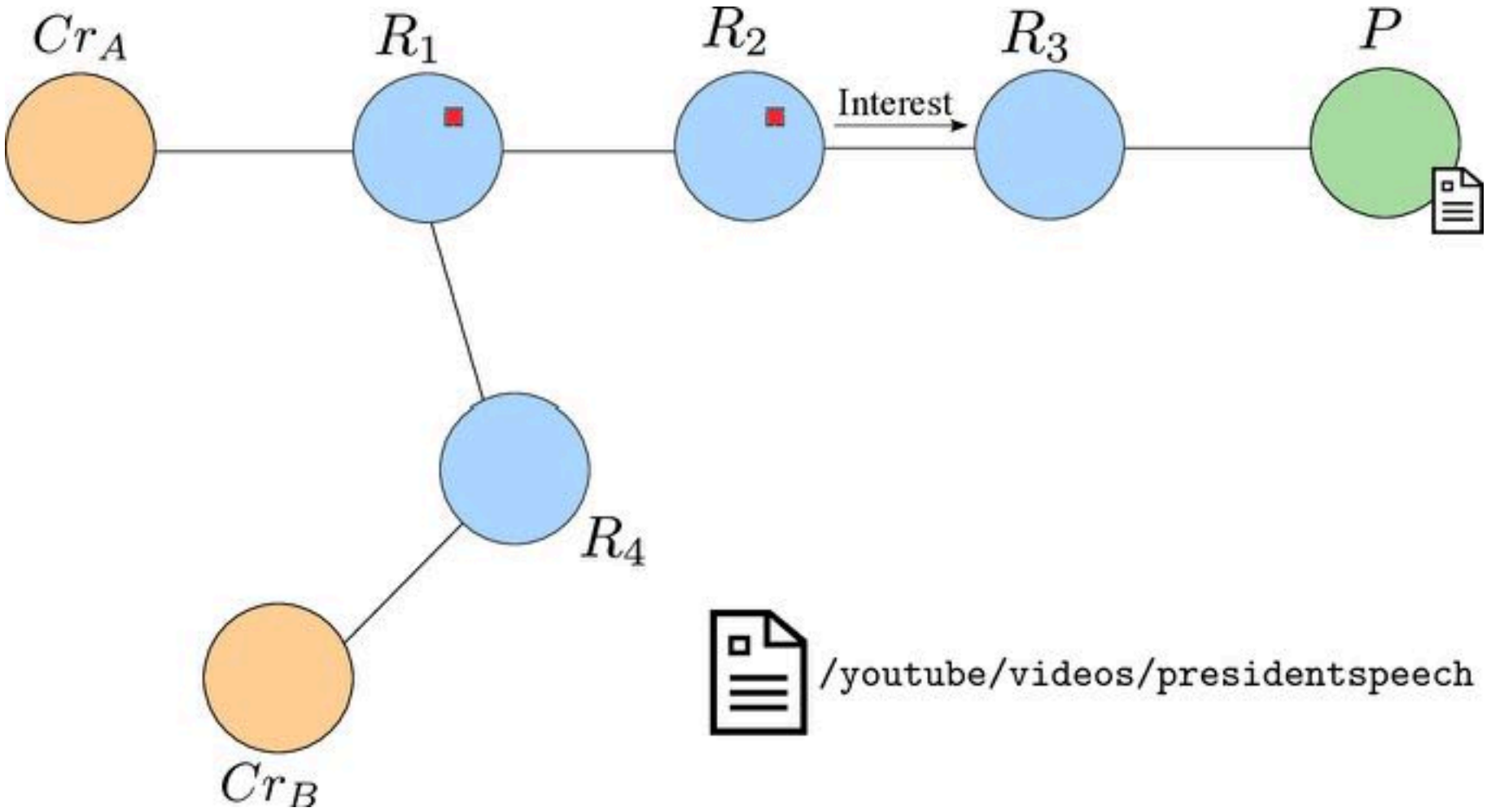
Example


Cr_A
R_1
R_2
R_3
P
Interest
R_4
Cr_B
/youtube/videos/presidentspeech

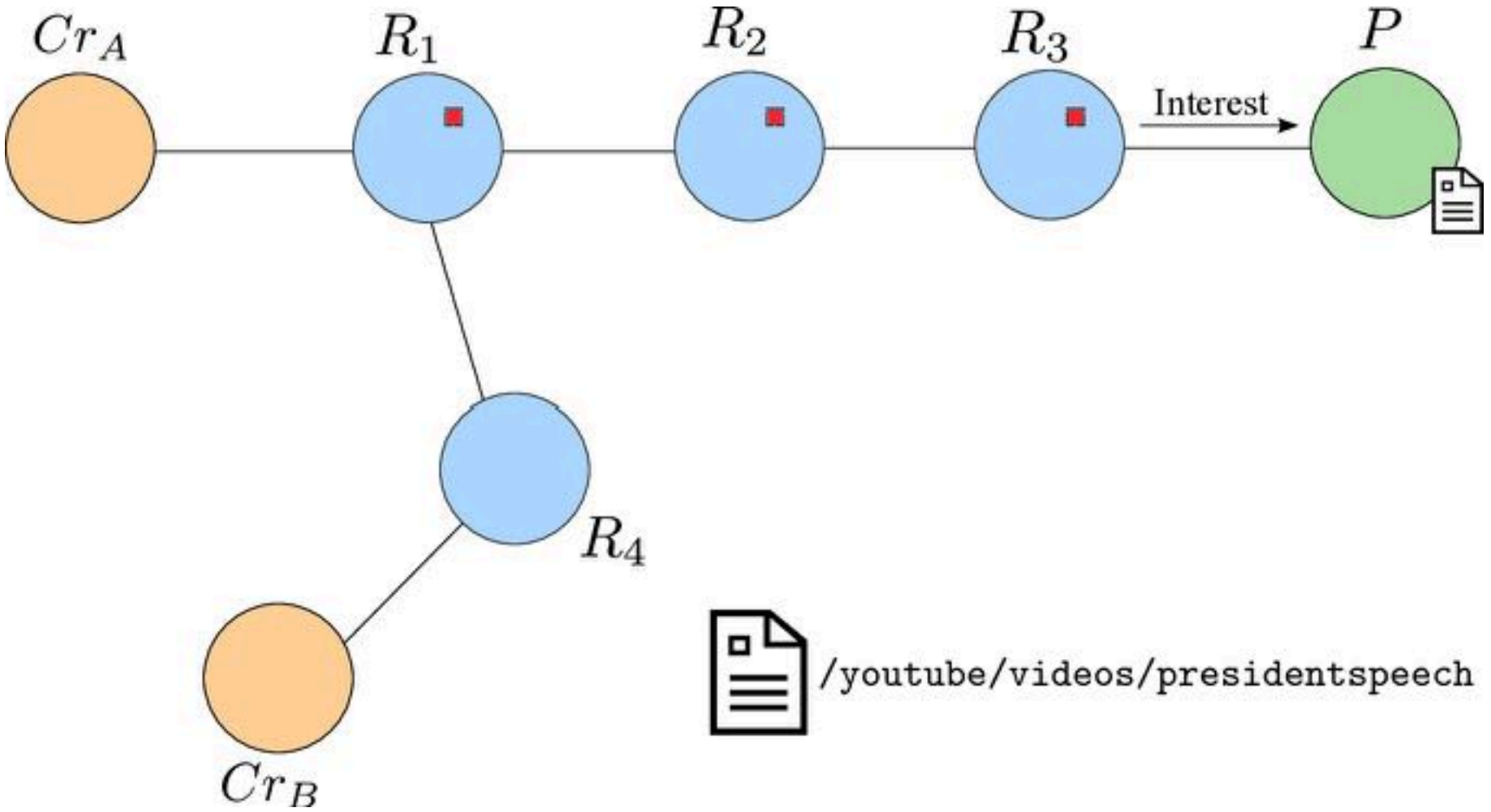
Example


Cr_A
R_1
Interest
R_2
R_3
P
R_4
Cr_B
/youtube/videos/presidentspeech

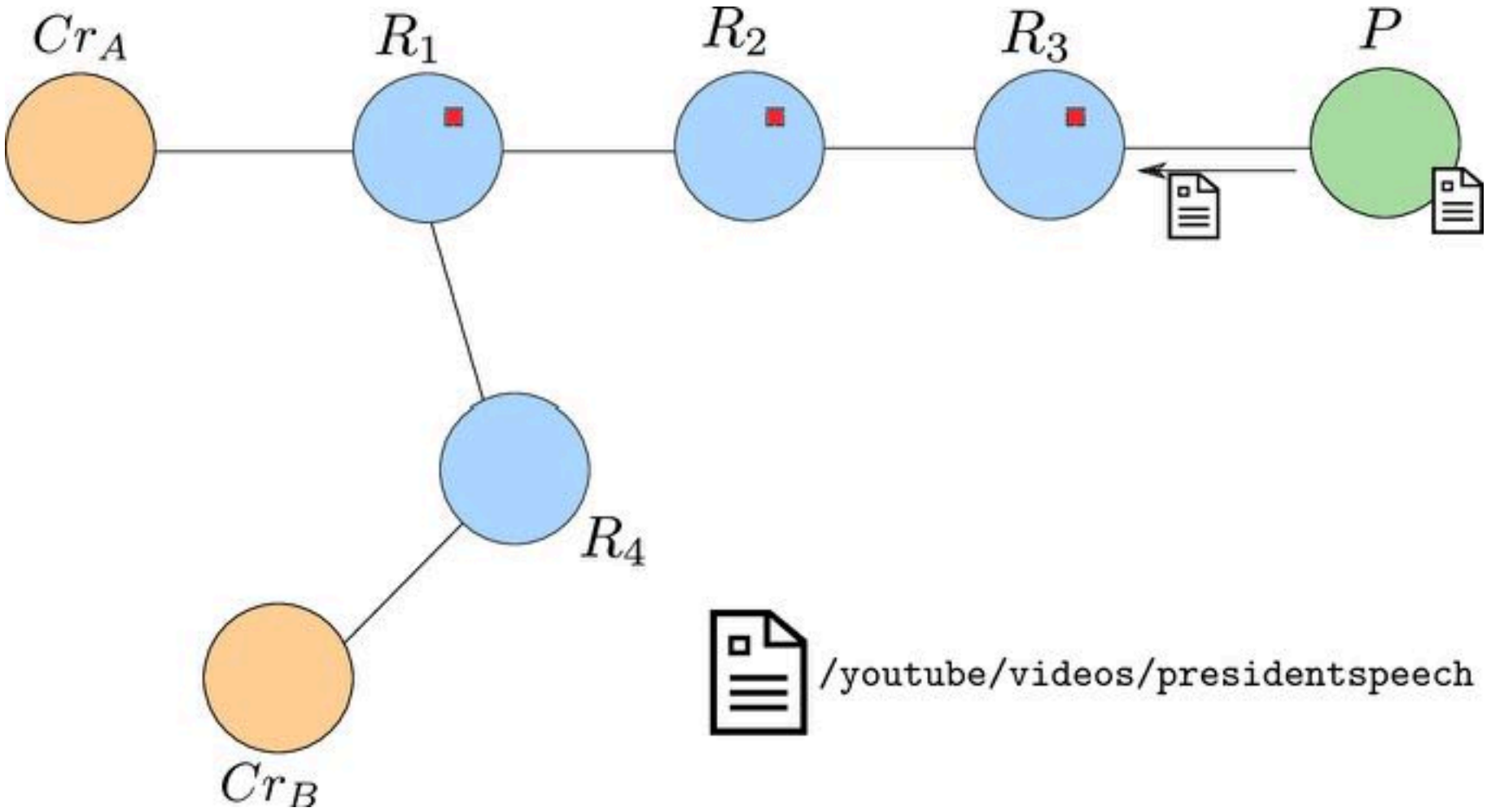
Example


Cr_A
R_1
R_2
R_3
P
Interest
R_4
Cr_B
/youtube/videos/presidentspeech

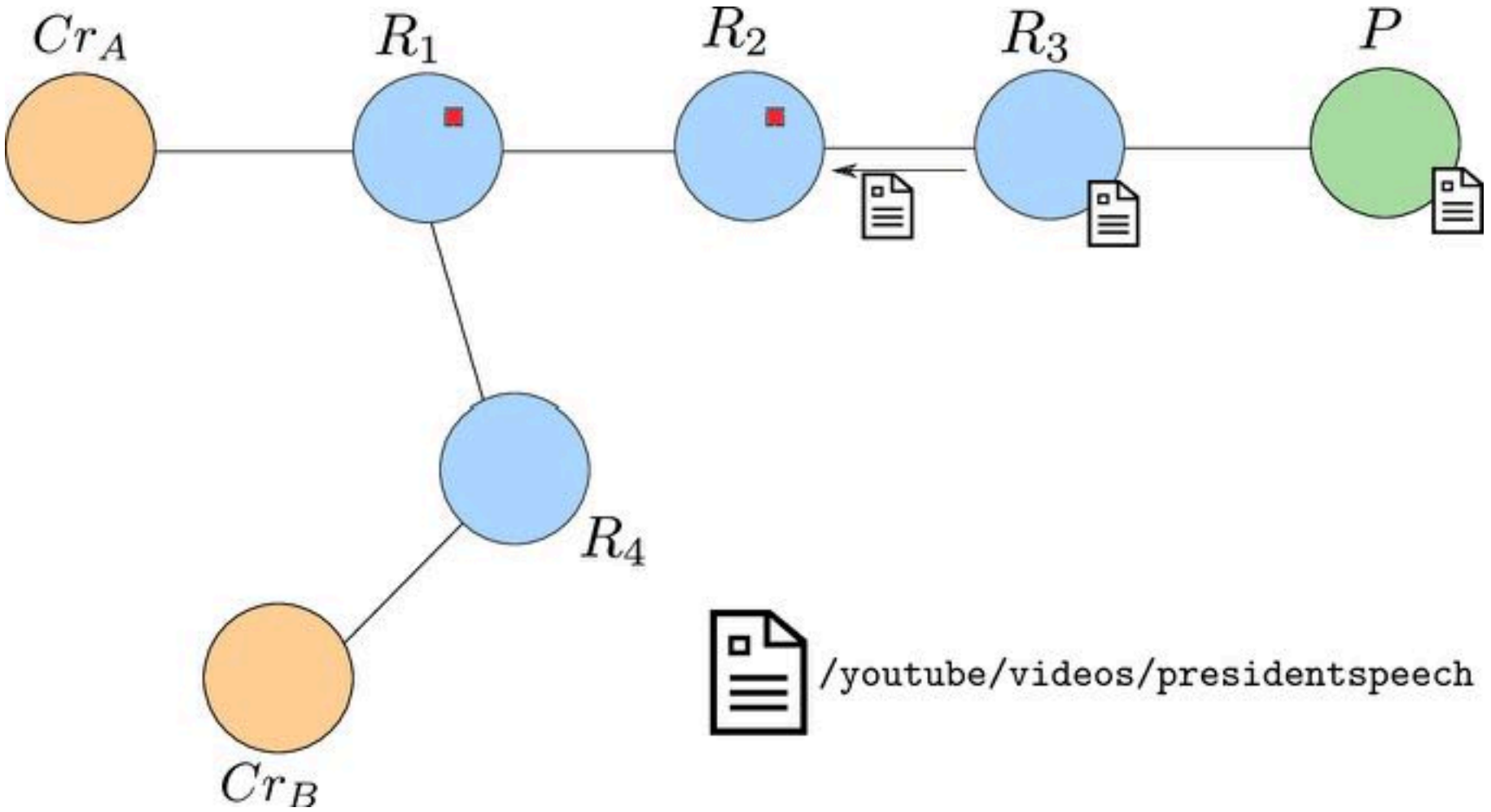
Example


Cr_A
R_1
R_2
R_3
P
Interest
R_4
Cr_B
/youtube/videos/presidentspeech

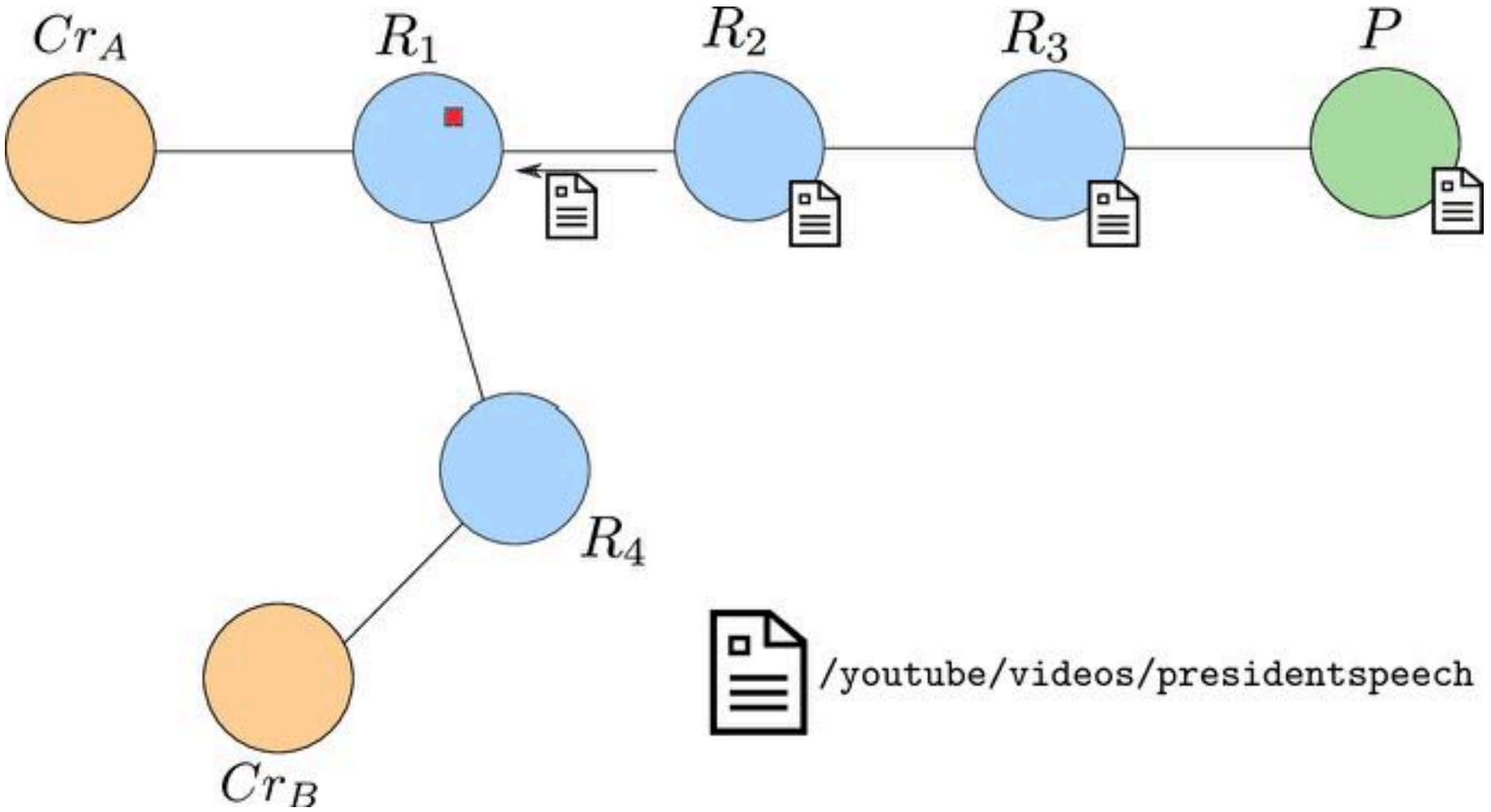
Example



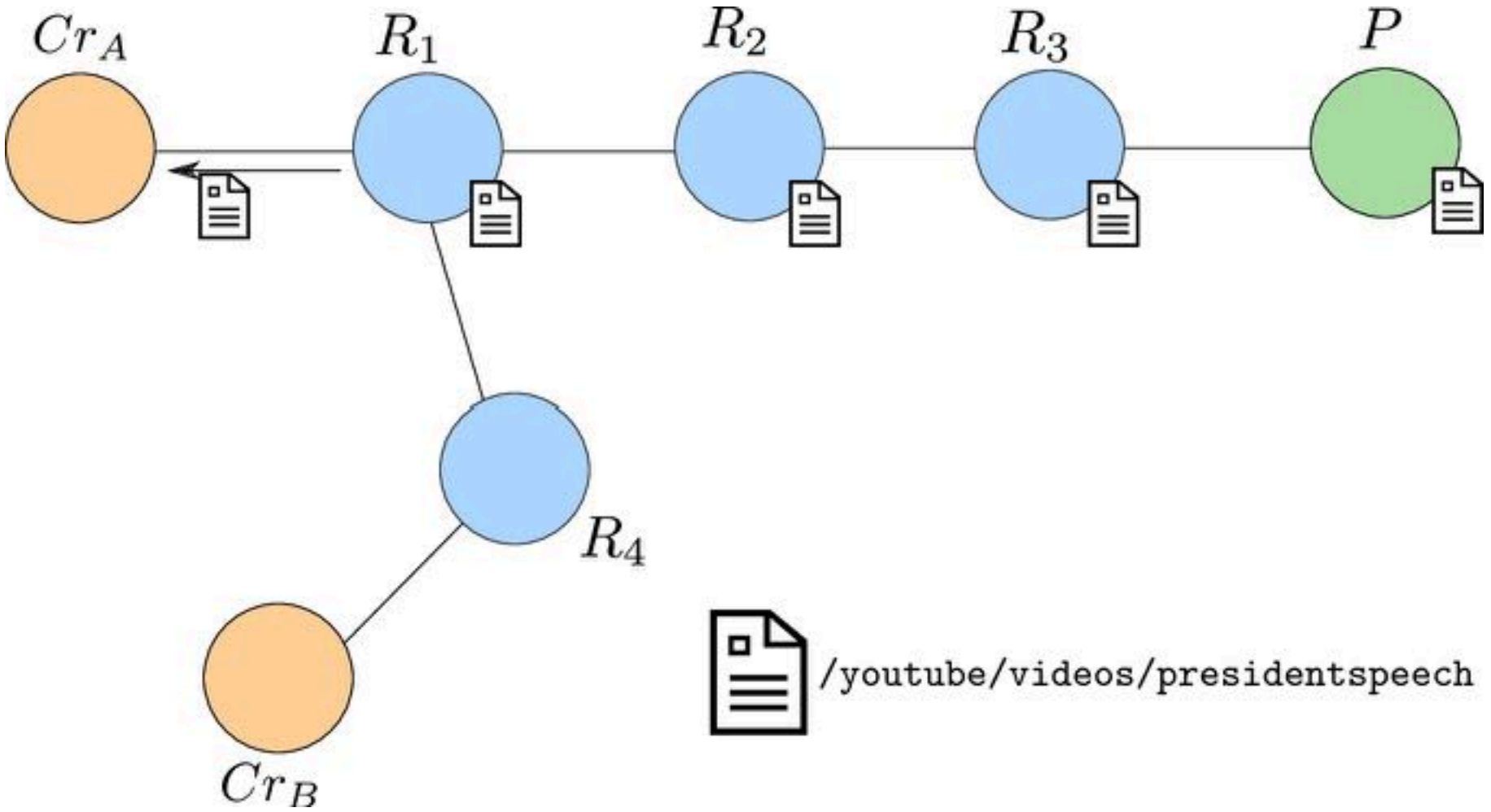
Example



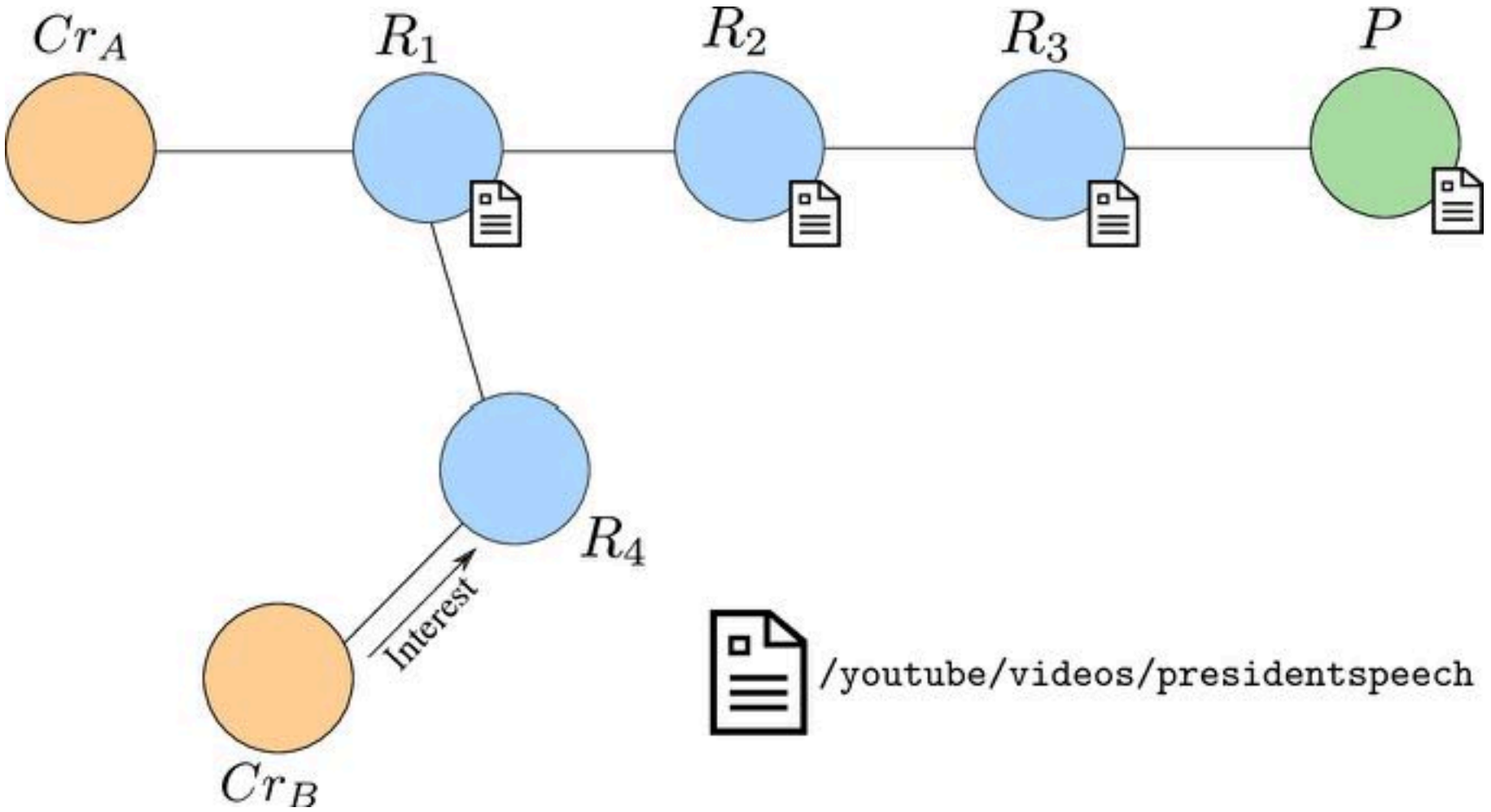
Example



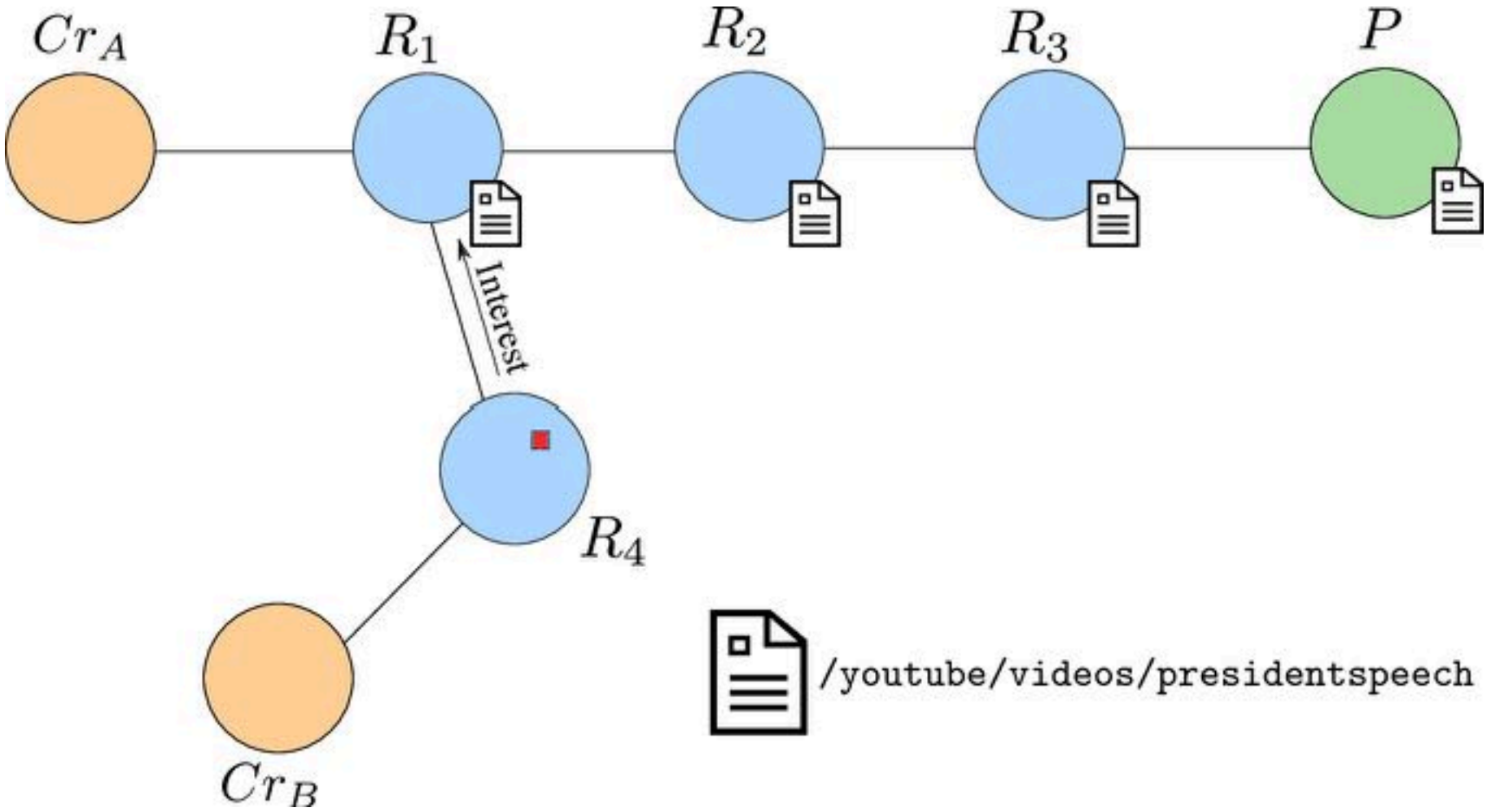
Example



Example


Cr_A
R_1
R_2
R_3
P
R_4
Interest
Cr_B
/youtube/videos/presidentspeech

Example


Cr_A
R_1
R_2
R_3
P
Interest
R_4
Cr_B
/youtube/videos/presidentspeech

Data Decoupling

- Data is decoupled from its origin
- Routers may cache content to satisfy future duplicate requests
- Benefits:
  - Upstream congestion and bandwidth utilization is reduced via caching
  - Consumer latency is reduced
  - QoS improves (theoretically)
- Drawbacks:
  - Content must be digitally signed (or verified by some other means) to ensure authenticity
  - **Producers have no way to know how many times their content was requested**

The Accounting Problem(s)

- Problem #1: How can a producer learn the number of times each content object was requested?
- Problem #2: How can a producer learn the above **and** which consumers requested the content?

Accounting Information

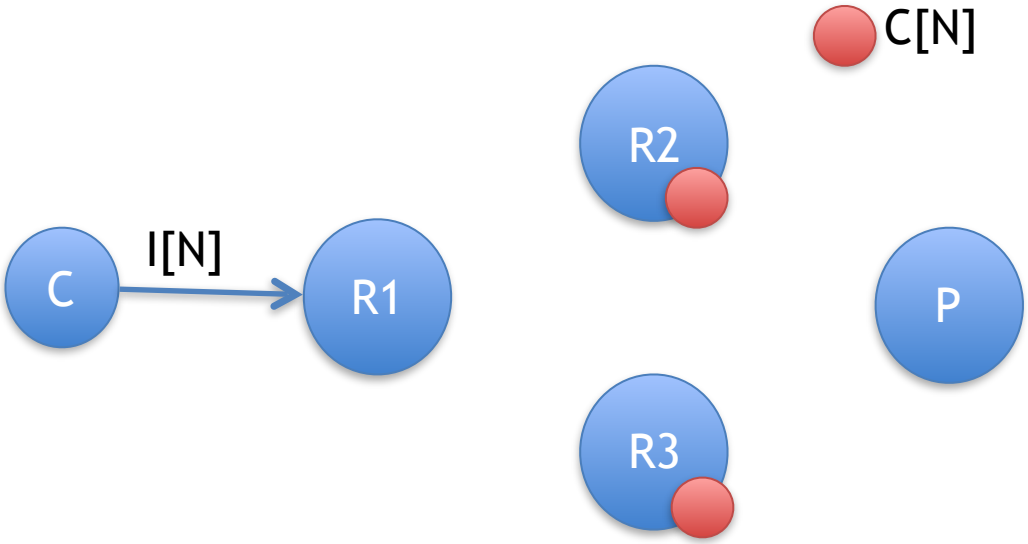
- Varying degrees of accounting information:
  - **Individual**: information about individual requests (and their sources)
  - **Distinct**: information about each individual request (but not about their source)
  - **Aggregate**: a simple count of the number of requests

Cache Hits vs Content Requests

- Question: Is the number of cache hits equal to the number of content requests?

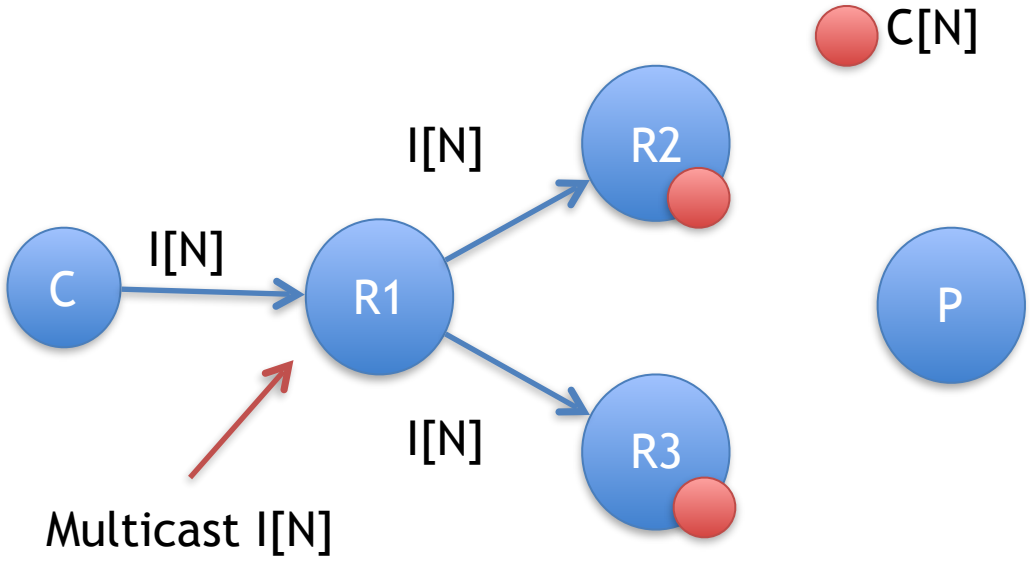
Cache Hits vs Content Requests

- Question: Is the number of cache hits equal to the number of content requests?
- Answer: No!



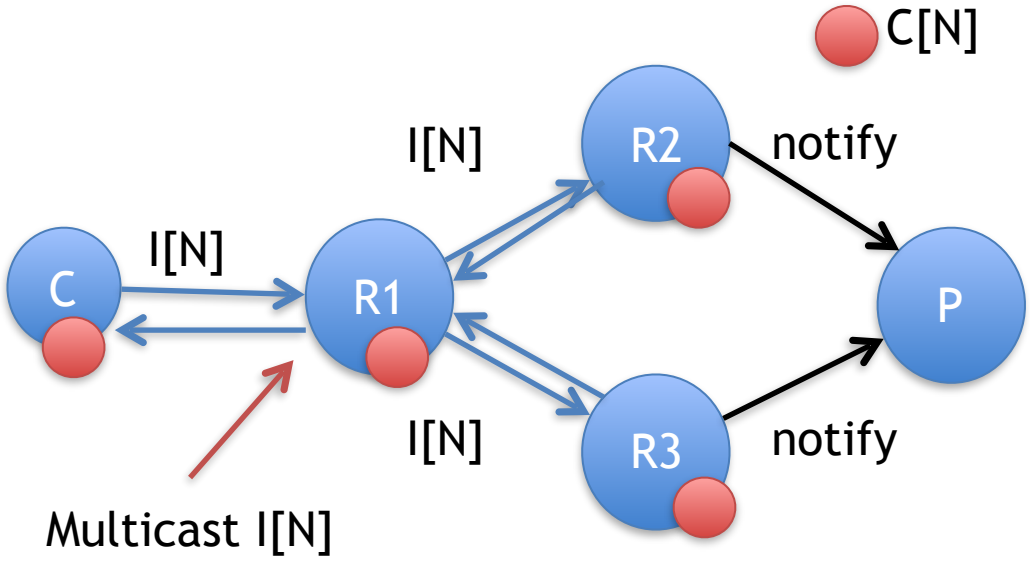
Cache Hits vs Content Requests

- Question: Is the number of cache hits equal to the number of content requests?
- Answer: No!



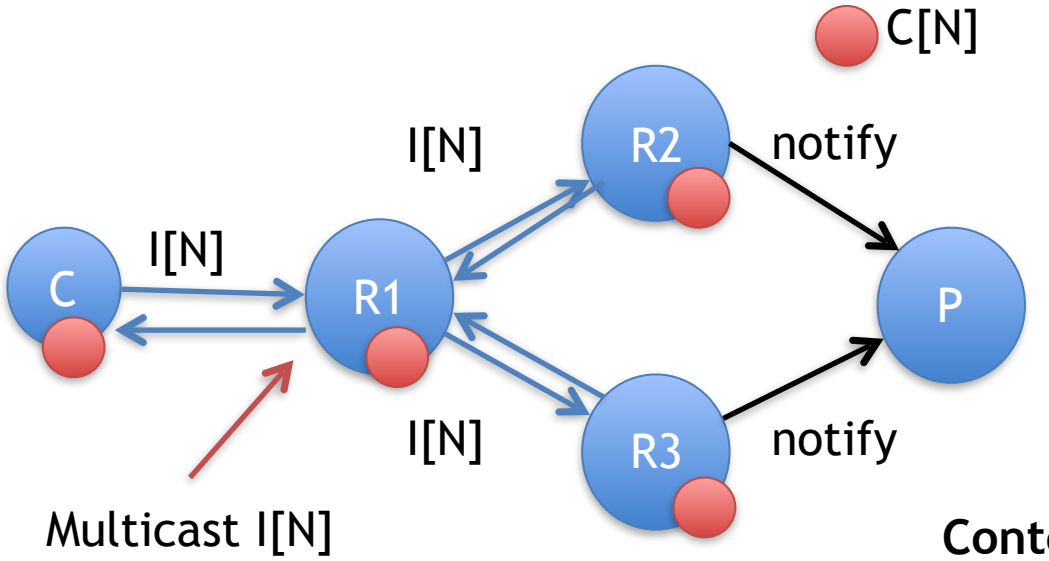
Cache Hits vs Content Requests

- Question: Is the number of cache hits equal to the number of content requests?
- Answer: No!



Cache Hits vs Content Requests

- Question: Is the number of cache hits equal to the number of content requests?
- Answer: No!



Content requests: 1

Cache hits: 2

Goals

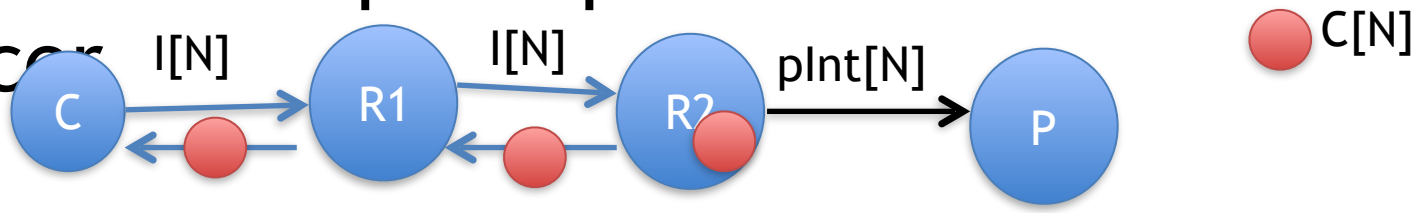
- Devise a mechanism to enable the collection of individual, distinct, and aggregate accounting information that is **correct** (accurate) and **secure** (when possible)
- Two approaches:
  - Encryption
  - Push Interests (our solution)

Accounting by Encryption

- If content were encrypted, every request would need to be accompanied with a request for the decryption key
  - The producer would learn about every request and could, if desired, enforce consumers to reveal their identities
- Problems:
  - Enforces two RTTs for a single piece of content
  - Mixes two unrelated problems (access control and accounting)

Solution: Use *Push Interests*

- A **push interest** (pInt) is an interest that leaves no PIT state in a router because it expects no response
- pInts carry a name and some payload (TBD)
- When a cache hit occurs a router creates and forwards a pInt upstream to the producer



Creating pInts

- The producer indicates the type of accounting information desired with an **accounting flag** ACCT in content:
  - ACCT $\in$ {INDIVIDUAL, DISTINCT, AGGREGATE}
- pInts echo this flag along with:
  - ORIGIN: an identifier for the originating router (e.g., prefix or public key digest)
  - COUNT: the number of requests served by the catalyst cache hit
  - CDATA: nonce or consumer-specific data

CDATA

- Aggregate
  - Contents: Random nonce and timestamp
  - Rationale: Distinguish between cache hit and content request cases
- Distinct:
  - Contents: Random (application-specific) nonce and timestamp
  - Rationale: Same as above
- Individual:
  - Contents: Something to identify the consumer
  - Rationale: No other way to identify the consumer! (There are no source addresses)

Main Theorems

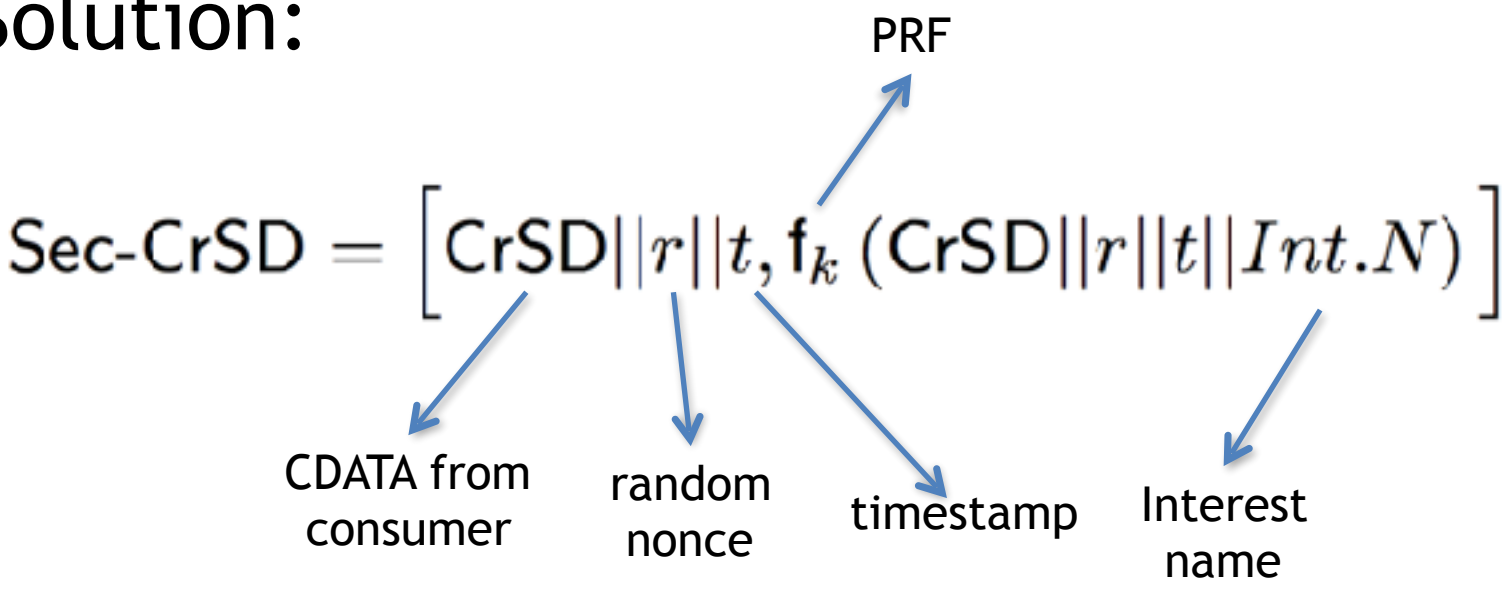
- Correctness: pInts with consumer-supplied nonces and timestamps are sufficient for guaranteeing correctness of the accounting scheme
  - Proof: see the paper
- Secure: coming up next

Secure Accounting

- Adversarial model: malicious router generating pInts for **bogus interests** when individual accounting is required
- This excludes:
  - Routers that don't generate pInt messages or generate pInt messages without forwarding content
  - A consumer that maliciously tries to inflate accounting information
  - A distributed adversary that can control routers and links

Security Result

- Claim: If a router cannot replay or forge individual accounting information, then the scheme is secure
- Solution:



Security in Practice

- We must assume:
  - Consumers know **what** CDATA to provide in an interest
  - Consumers have the public key of the producer to generate the secure CDATA
  - **Consumers behave honestly**

Dishonest Consumers

- Problem: If we assume consumers behave dishonestly (i.e., they generate fake CDATA for individually accountable content), they can bypass the accounting mechanism
- Solution: Routers must verify the secure CDATA contents
- **Problem:** Routers cannot verify CDATA contents at the network layer
  - **Individual accounting via pInts is not possible.**

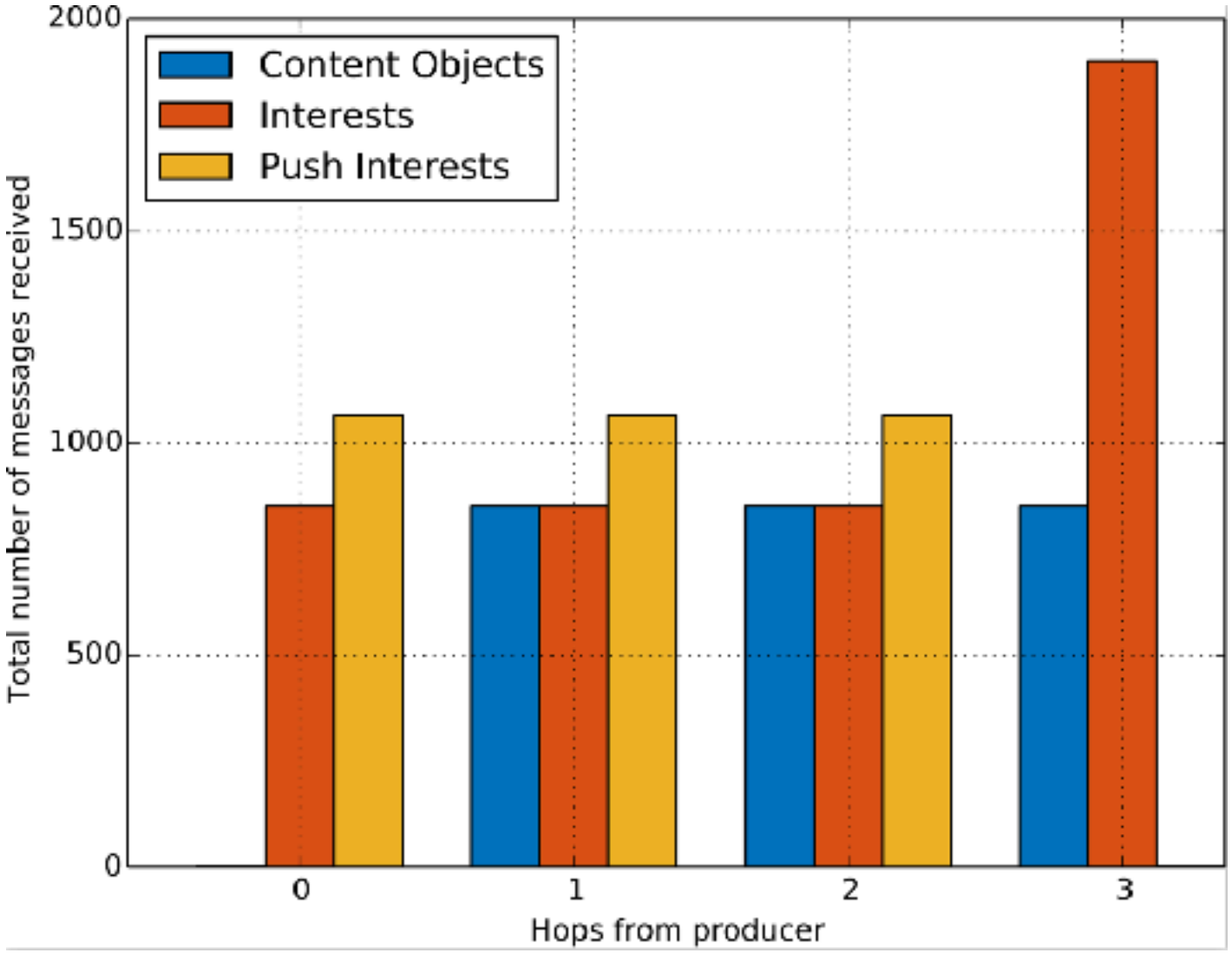
Recommendations

- Consumers always include a random nonce in interests
  - NDN does this, CCN does not!
- Routers copy these nonces into pInts regardless of the ACCT flag for cached content

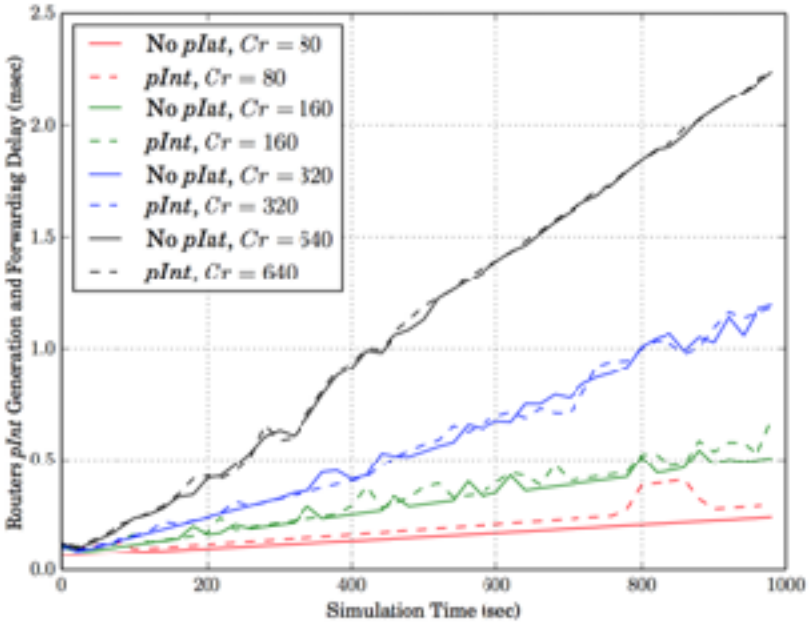
Performance Assessment

- Interested in the following metrics:
  - Upstream congestion (due to pInts)
  - Forwarder overhead from processing pInts
- Ran simulations in ndnSIM based on the popular DFN and AT&T topologies

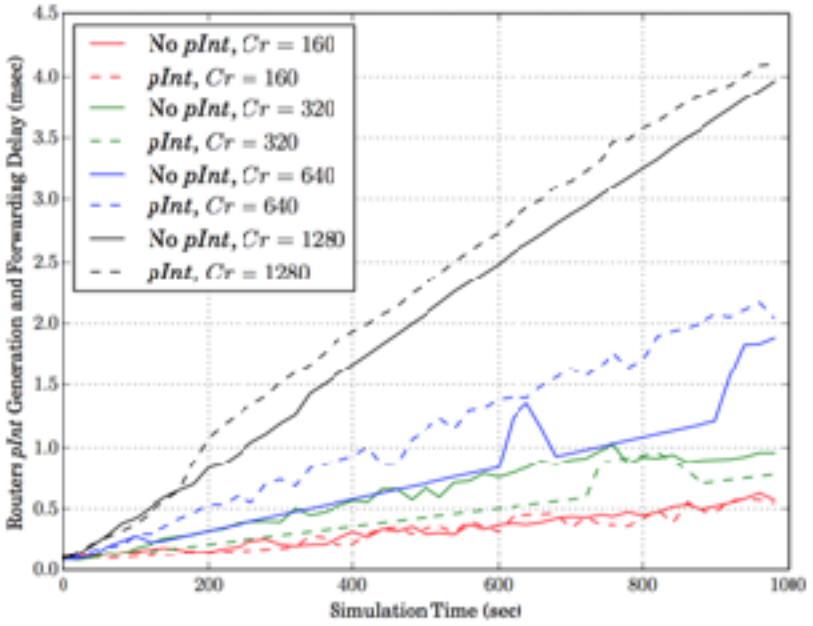
Upstream Congestion



Forwarder Overhead



(a) DFN topolcgy.



(b) AT&T topology.