Syndicate: a CDN-Powered Distributed Read/Write Filesystem

Jude Nelson Larry Peterson

Problem

– Reading large datasets across a network can be **slow** and **expensive**, due to both the underlying media and network constraints.

– Existing distributed systems achieve scalable read performance on large data through **replication**, but at the **expense** of a client or mirror server.

Background

– Internet content providers alleviate media and network constraints with a **CDN**, which implements a series of **network caches** which can be used to place content closer to users.

– A CDN provides **replication as a network service**.

– A CDN implements its own policies on which content can be cached, and assumes that the content is **immutable**.

Solution

– Provide an **application-agnostic** way to **resolve**, **publish**, and **withdraw** content in a CDN. This is naturally achieved with a filesystem interface.

– Each client is an **origin server** for its locally-created files and has permission to cache its content in the CDN.

– Clients synchronize file metadata state with a network-accessible **metadata service**, which defines the filesystem hierarchy and file data URLs.

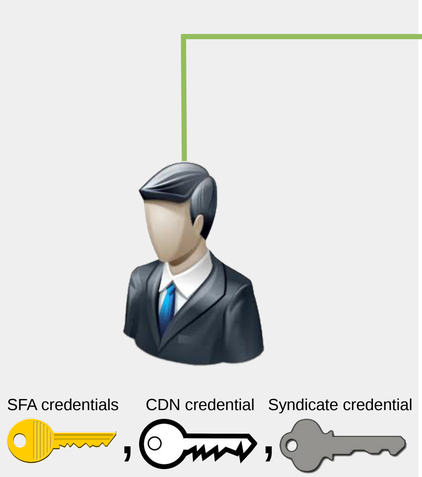
– File names are unique to each content version to **provide cache consistency** in the CDN.

– The metadata service resolves write-conflicts with last-write-wins to achieve AFS-style **close-to-open file consistency**.

– Keep all data on underlying local storage whenever possible to allow **arbitrary persistence measures** to be taken.

Implementation

Administrators

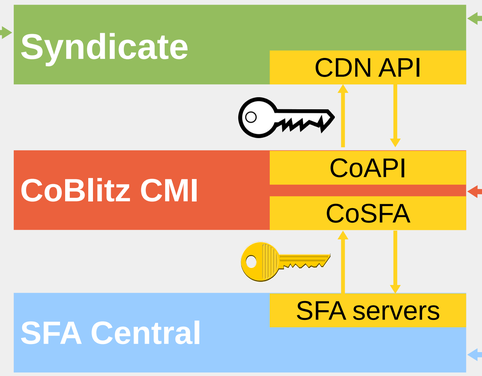


– Our prototype uses the CoBlitz CDN, which is deployed in a **SFA federation**.

– Syndicate administrators have credentials for Syndicate, the CDN, and the SFA federation.

– Administrators manage Syndicate users and Syndicate filesystems.

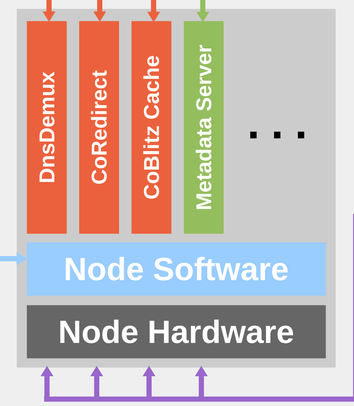
Service Managers



– Service managers are logically centralized software systems that control one or more slices in the SFA federation.

– With administrator-given credentials, Syndicate controls the CDN and its metadata service, and the CDN controls its slices in the SFA federation.

SFA Slices

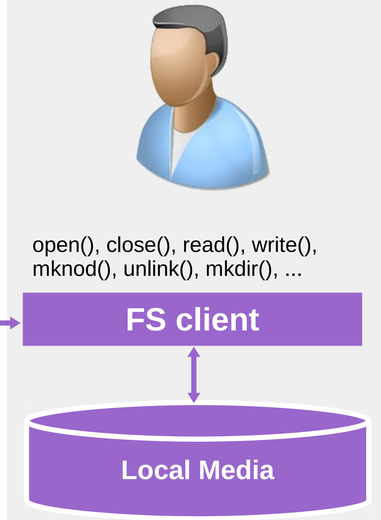


– The CDN components run in one or more slices in a SFA federation.

– The metadata service runs as a separate slice.

– Each metadata service sliver runs arbitrarily many filesystem metadata servers.

Clients



– **Creating** a file publishes it on the CDN.

– **Deleting** a file withdraws it from the CDN.

– **Reading** a file resolves it in the CDN and streams it from the CDN to the reader.

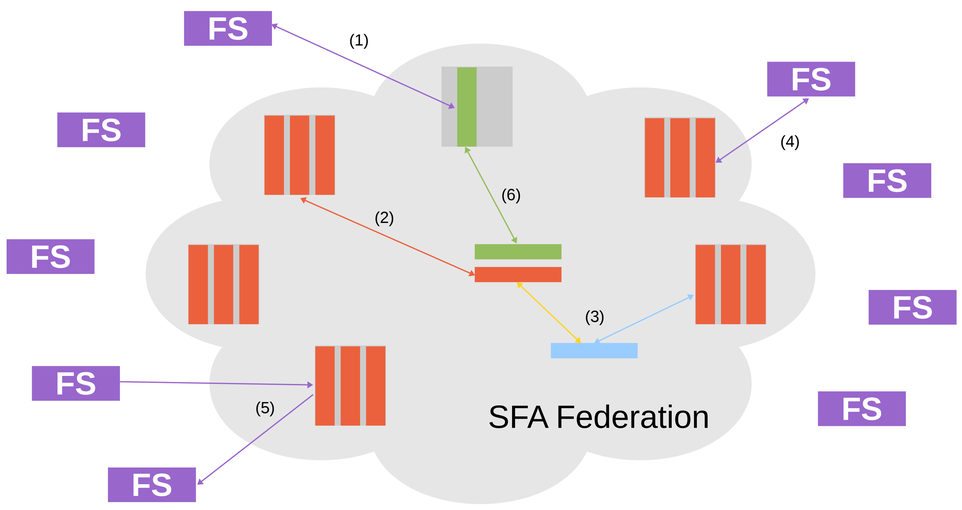
– **Writing** a file re-publishes it on the CDN with a new URL.

– Written data is kept on underlying storage.

– Metadata is encrypted, but content is not.

– Only synchronizes metadata deltas for speed.

– Users' machines must be authorized by the administrator to host CDN-cacheable content.



(1) Metadata synchronization

(2) CoBlitz configuration update

(3) Slice update (via CMI)

(4) Remote read, cache hit

(5) Remote read, cache miss

(6) Metadata server update