DIRECT ACCESS FILE SYSTEM (DAFS)

Studente: Antonio Allocca

Docente: Giuseppe Catteneo

Syllabus

- **RDMA**: Direct memory-to-memory data transfer operations between remote nodes
- **DAFS**: Direct Access File System
- **DAT**: Direct Access Transports
- **RDDP**: Remote Direct Data Placement
- **MMI**: memory-to-memory interconnects
- **NIC**: Network Interface Card
- **DAPL**: Direct Access Provider Layer

Introduzione

Ideato come protocollo per accedere ad un file system remoto.

Le applicazioni utilizzano il DAFS tramite una libreria di I/O definita a livello utente per non caricare il sistema operativo.

DAFS acquisisce molto concetti e schemi di NFSv4 che a sua volta acquisisce feature di CIFS.

A differenza però di NFSv4, DAFS fornisce delle API native che prendono il nome di DAFS API.

Obiettivi del DAFS

low-latency
high-throughput
low-overhead
Della comunicazione tra client e server. Così facendo si alleggerisce anche il carico sui client e di conseguenza si aumentano le performance del DAFS

Fornire la semantica per un accesso affidabile e condiviso in un sistema con cluster. Vengono inclusi miglioramenti per permettere lo scambio di file tra i client e la cooperazione di essi tramite file condivisi.

Motivazioni

Possiamo accedere ai file in 3 modi:

1. Disco locale
2. Disco remoto condiviso
3. NFS

Disco remoto condiviso

Per accedere ad un disco remoto condiviso tra dei client bisogna tenere conto di cose che nei local file system non si prende in considerazione, ovvero la distribuzione dei dati condivisi, protocolli e la complessa gestione delle failure.

Inoltre è richiesta omogeneità tra i client, cosa che vogliamo evitare.

Ha un vantaggio: la velocità è relativamente comparabile alla velocita di accesso alle risorse persistenti dei Dischi locali

Problemi con NFS

Gli NFS permttono l'eterogeneità dei client e sfruttano una rete comune per funzionare.

Tale caratteristiche però hanno uno svantaggio, ovvero rendono gli NFS più lenti.

Le cause:		
overhead della CPU	aumento della latenza dovuta ai layer dei protocolli di rete	banda limitata

Miglioramenti

Molte NFS implementano protocolli a scambio ti messaggi per eseguire operazioni sul server e c'è meno data locality e cooperazione di un cluster file system.

DAFS è un NFS che usa Direct Access Transport basato su RDMA

Differenza con NFSv4: Orientato al data center e implementa il modello local sharing

DAFS quindi fornisce più data locality e cooperazione tra i client e le applicazioni che vengono eseguite su di essi.

Facendo così si migliorano le performance dei client che eseguono le applicazioni all'interno del cluster, mantenendo però le caratteristiche del modello di fault isolation che caratterizza gli NFS.

Obiettivo principale

Massimizzare i benefici di una DAT (Direct Access Transport)

La DAFS permette ai client di trasferire dati dal server senza coinvolgere la CPU

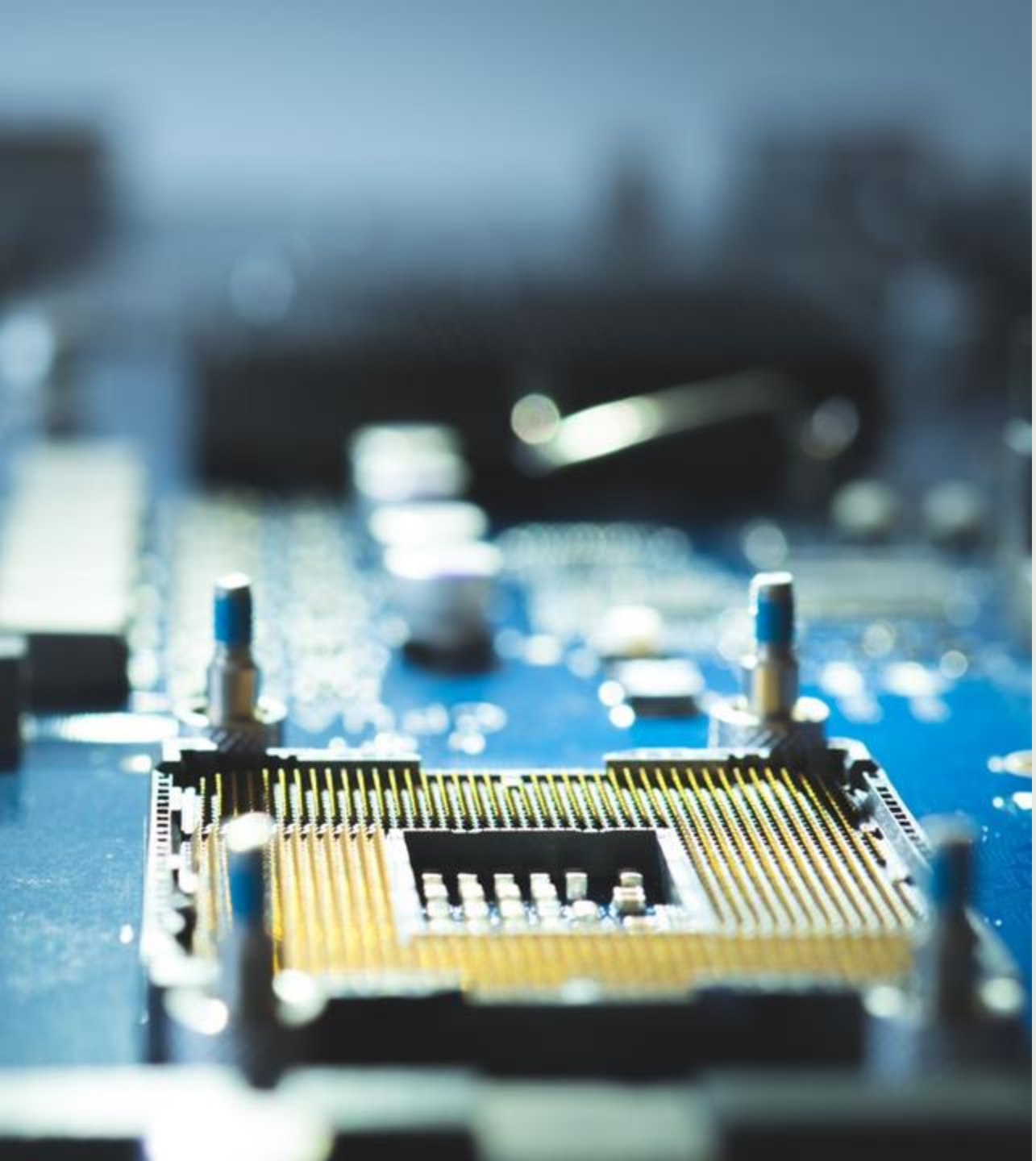
La DAT permette di non creare molto carico sulle interfacce di rete che supportano i low-level network protocol

DAT (Direct Access Transports)

Il DAT al momento del rilascio dell'articolo era la tecnologia che permetteva il trasferimento più veloce dei dati infatti la community dell'high-performance computing da tempo utilizza teconologie memory-to-memory interconnections che riducono l'overhead del sistema operativo e forniscono accesso diretto alla memoria.

Cosa fornisce il DAT:

- **RDMA** (comunicazione dirett tra i nodi senza intermediari)
- **kernell bypass** (riduzione del coinvolgimento dell'OS)
- **interfacce asincrone** (pipelining efficiente)
- **memory registration** (specifica come i NIC accedono alle regioni di memoria dell'host per essere usate come destinazione delle operaione RDMA)



Misurazioni

Ogni sistema di immagazzinamento dati costruito a partire dal DAT può essere misurato in termini di:

1. throughput
2. latenza
3. client CPU (la CPU del server non è un fattore limitante in quanto la struttura è facilmente scalabile)

Il protocollo DAFS

il DAF è un protocollo client-server bassato su sessioni. La sessione viene stabilita quando il client comunica con il server. Quando la comunicazione comincia il client deve autenticarsi e deve stabilire:

- byte-ordering
- checksum
- message flow control e transport buffer
- credenziali

Descrittori

La comunicazione avviene tramite descrittori pre-allocati per inviare e ricevere i dati.

Ogni descrittore può contenere più segmenti.

La maggior parte dei messaggi dei DAFS sono di piccola taglia. Ciò permette ai server di allocare poche risorse in attesa delle richieste dei client.

Per i file di grande taglia, invece, vengono adoperate azioni di read e write.

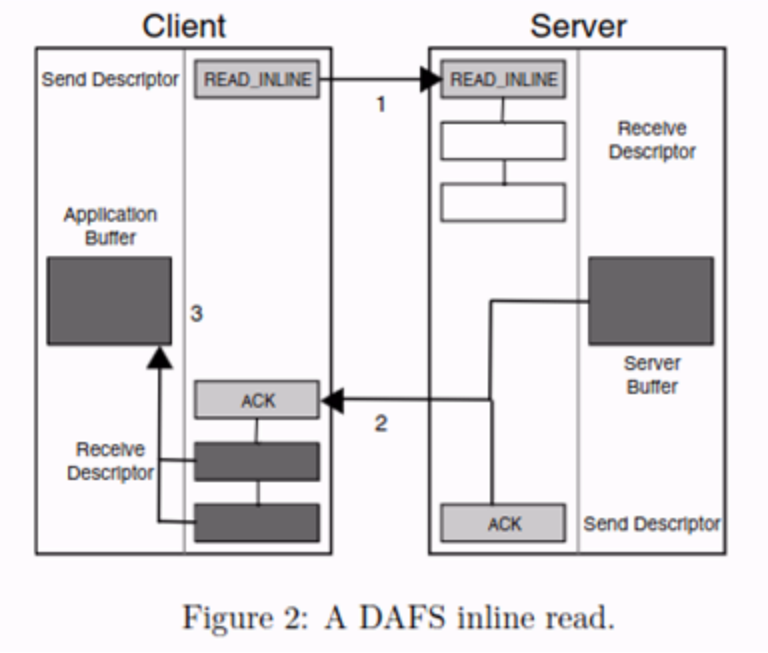


Figure 2: A DAFS inline read.

Miglioramenti delle performance

Tra le varie caratteristiche che danno la possibilità di migliorare le performance ci sono:

- Formato dei messaggi
- Autenticazione basata su sessione (ridotti tempo e CPU per fornire la sicurezza per ogni singola richiesta)
- Operazioni I/O (per file di grande dimensione)
- Ridotto l'overhead dei client (nessuna copia su buffer o intermediario in quanto la posizioni delle risorse è conosciuta.)
- Batch I/O facility (controllo sulla pipeline I/O e selle risorse all'interno dei buffer)
- Cache hints
- Chaining di operazioni (il messaggio ha il formato standard facilitando il marshalling e il parsing)

Nuove features per il file sharing

Due feature importanti che differiscono dai tradizionali NFS come NFSv4

1) Accesso condiviso (applicazioni su cluster e meccanismi di condivisione delle chiavi)

2) Comportamenti per il recovery del sistema (meccanismi addizionali di locking e il comportamento "exactly-once failure" che aiuta con il recovery)

I comportamenti:

- Elaborato sistema di locking (basato su timeout come NFS e con sistema di auto-recovey)
- Cluster fencing (abolizione di un cluster manager per rendere i nodi indipendenti)
  N.B. Non viene abolito il server
- hared key reservations
- Request throttling (gestito tramite la sessione)
- Exactly-once failure semantics (sorge il problema dei messaggi non arrivati e della trasmissione, ciò non può essere gestito solamente tramite timeout quindi si danno dei crediti ai client che non sono riusciti a comunicare con il server)

Sicurezza e Autenticazione

Il protocollo DAFS si basa sul livello di trasporto per fornire privacy e cifratura.

Il client del DAFS possono essere creati a livello utente, quindi sulla stessa macchina ci possono essere più client

Nasce il bisogno di un forte meccanismo di autenticazione.

Si può effettuare un classico login con username/password oppure utilizzare le DAFS API.

LE DAFS API

Le DAFS API sono delle semplici interfacce programmabili per una DAFS. Fornisce la trasparenza e nasconde ciò che viene fatto nel protocollo come la gestione delle sessioni ecc.

Sono definite a livello utente e supportano l'interazione con l'OS solo quando strettamente necessario.

Differenza DEFS API - POSIX

DAFS API differiscono da POSIX per **4 aree importanti**:

1. **Asincronia**
2. **Memory registration** (la memoria dell'utente deve essere registrata in una NIC per poter ricevere dati dalle operazioni RDMA)
3. **Completition groups** (metodo di aggregazione per operazioni completate)
4. **Comportamento esteso**

- Powerful batch I/O API
- Cancel and expedite functions
- fencing API
- Opzioni aggiuntive quando viene aperto un file tipo cache hints e shared keys.
- Un'infrastruttura di autenticazione estendibile.

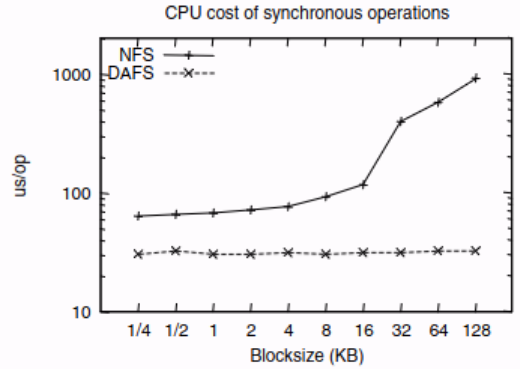


Figure 4: CPU time consumed per synchronous read request.

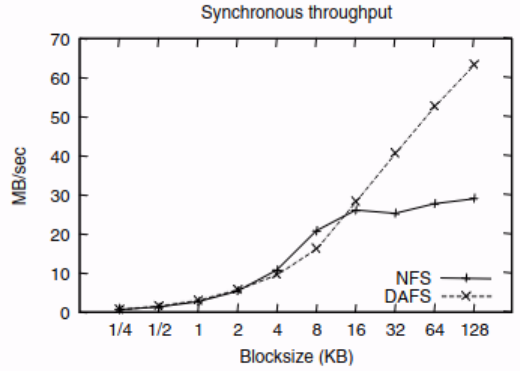


Figure 5: Synchronous throughput.

PERFORMANCE

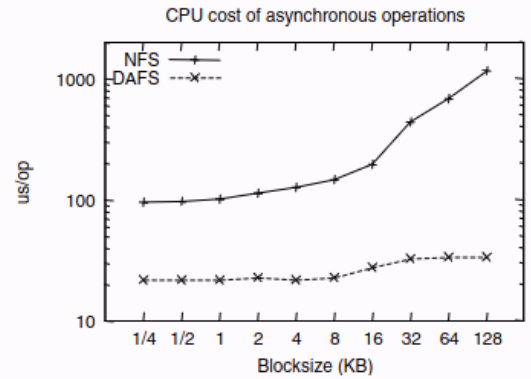


Figure 6: CPU time consumed per asynchronous read request.

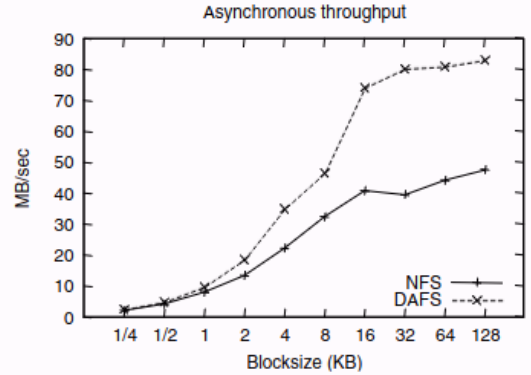


Figure 7: Asynchronous throughput.

Conclusioni

Il protocollo DAFS permette un file sharing locale, come in data center, ad alte performance ed ha come obiettivo la fruizione di file ad un cluster di client. Per raggiungere gli obiettivi proposti si è dovuta gestire la condivisione dei dati.

DAFS poneva le basi per un NFS ad alte prestazioni, scalabile e condivisibile che sfrutta le tecnologie di quel tempo.

GRAZIE PER L'ATTENZIONE