
Lustre

El sistema de almacenamiento para granjas

Resumen

Lustre es el sistema de archivos que necesitan los supercomputadores que tienen requisitos de rendimiento y robustez para su sistema de almacenamiento muy altos. Actualmente es opensource y ofrece muchas más ventajas que desventajas

¿Qué es Lustre?

Es un sistema de archivos opensource distribuido que está pensado para el entorno de la computación de altas prestaciones. Es capaz de trabajar con varias interfaces de red al mismo tiempo, es extremadamente escalable, seguro, robusto y altamente disponible. Es capaz de servir a más de 10000 nodos de computación, proporcionar petabytes de almacenamiento y hacer transferencias de cientos de GB sin inmutarse.

Historia

1999 - Peter Braam en la *Carnegie Mellon University* (Pennsylvania) comienza a desarrollar Lustre como un proyecto de investigación.

2001 - Braam funda su propia empresa, *Cluster File Systems*, y es invitada a un proyecto del Departamento de Energía de EEUU junto a HP e Intel para desarrollar Lustre.

2007 - *Sun (Oracle)* absorbe Cluster File Systems, desarrollando y manteniendo Lustre.

2010 - *Sun* abandona Lustre. Se crean comunidades opensource, y otras grandes se interesan en su desarrollo (*Open Scalable File Systems Inc.*, *European Open File Systems* y otras).

2013 - *Seagate* adquiere la propiedad intelectual de Lustre y lo desarrollan junto a las comunidades opensource.

2015 - *Seagate* dona Lustre a la comunidad de usuarios.

¿Quién usa Lustre?

En el top500, 60 de los 100 primeros utilizan Lustre, incluyendo a seis de los diez mejores. Algunos de ellos son:

Tianhe-I, que fue 1º en el TOP500, en China en el 2011.

Titan, 2º en el TOP500, en EEUU.

Jaguar, 3º en el TOP500, en EEUU.

K computer, 4º en el TOP500, en Japón.

Otros supercomputadores que lo utilizan son el Lawrence Livermore National Laboratory (LLNL), el Oak Ridge National Laboratory, el Pacific Northwest National Laboratory, el Texas Advanced Computing Center, la NASA, el Tokyo Institute of Technology, TOTAL y muchos otros.

¿Qué ofrece Lustre?

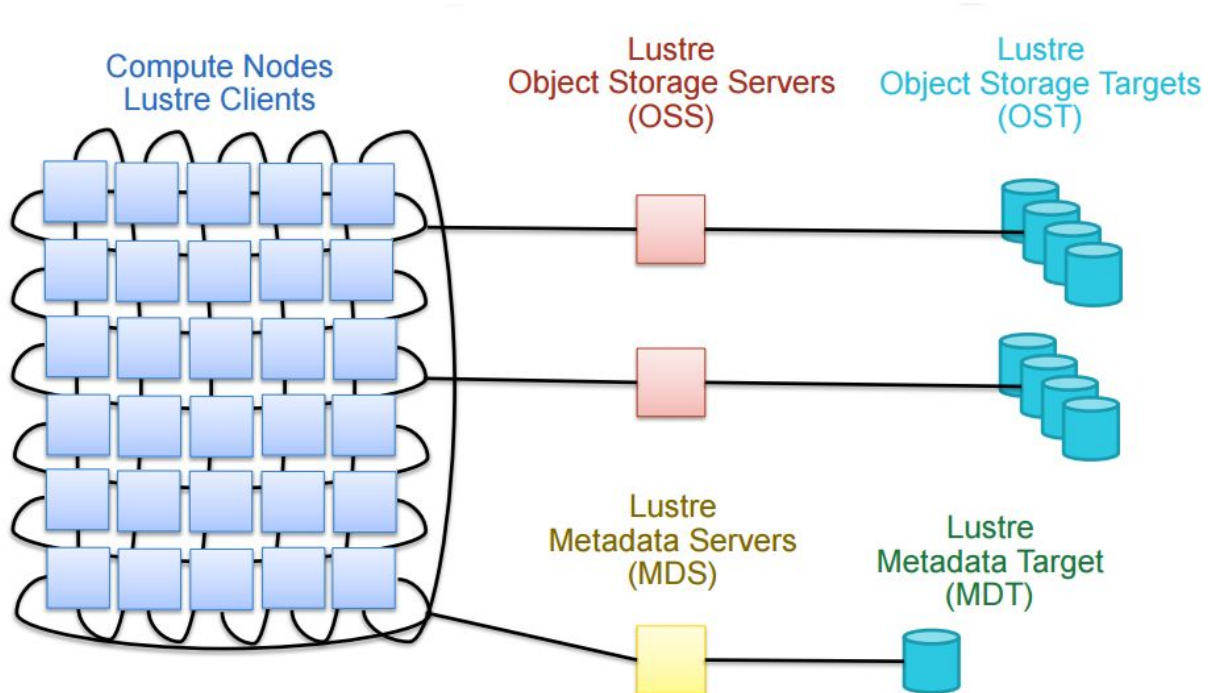
Ventajas:

- Pensado para un sistema distribuido:
 - Escalabilidad inigualable.
 - Fiabilidad probada.
 - Alto rendimiento.
- Es de **código abierto**. Esto tiene dos ventajas: es gratis y su código fuente es abierto.
- Ofrece un **elevado ancho de banda**: cada vez que se añade un OSS (y su correspondiente OST), se aumenta el ancho de banda.
- **No es específico de un fabricante de hardware** de servidores o cabinas de almacenamiento o de interconexión. Da igual el servidor que se use, da igual el almacenamiento que uses y da igual la interconexión. Obviamente, cuanto más uniforme sea el cluster, más sencillo será de configurar.

Desventajas:

- Solo soporta versiones muy concretas de Linux 2.6 kernel , Red Hat ENterprise linux 4,5 y SUSE 9, 10.
- Las wikis están obsoletas.
- Las versiones no son retrocompatibles.
- Hacer un backup de Lustre tiene una serie de problemas. El volumen de datos generalmente hablamos de *PetaBytes* de datos, lo que hace que sea muy complicado hacer backups al necesitar grandes cantidades de tiempo seguido. Además el software de backup tiene que ir fuera de Lustre por lo que NO es LAN free, es decir, es por red y esto hace que vaya muy lento.
- No tiene ni es compatible con sistemas de almacenamiento jerárquicos (HSM).
- No soporta protocolos como NFS y/o CIFS
- No es fácil de configurar ni de administrar

¿Cómo se estructura Lustre?



-
- **Metadata Server (MDS)** – El MDS tiene los metadatos almacenados para los clientes de Lustre. Cada MDS gestiona los nombres y directorios en el sistema de ficheros de Lustre, proporcionando la solicitud de red para la manipulación de uno o más MDTs. Es necesario mencionar que puede haber varios servidores MDS distribuidos por toda la red del clúster para ofrecer la respuesta más rápida a un nodo cliente sin importar su localización.
 - **Metadata Target (MDT)** – El MDT guarda los metadatos (como nombres de archivo, directorios, permisos y la disposición de los archivos) en algún dispositivo de almacenamiento que está conectado a al MDS.
 - **Object Storage Servers (OSS)** – El OSS ofrece un servicio de E / S de archivos y además de gestionar la red encargada a manejar las peticiones de uno o más OST.
 - **Object Storage Target (OST)** – Es el lugar donde se almacenan los datos del fichero del usuario, estos fichero se graban en unos o más objetos y cada objeto se separa en un OST.
 - **Cliente Lustre** – Se ejecuta en el resto de las máquinas del clúster. El software de cliente Lustre proporciona una interfaz entre el sistema de archivos virtual de Linux y los servidores de Lustre. El software de cliente incluye un Management Client (MGC), un Metadata Client (MDC), y varios Object Storage Clients (OSCs), uno correspondiente a cada OST en el sistema de archivos. Un volumen objeto lógico (LOV) agrega las OSCs para proporcionar acceso transparente en todos los OST. Así el cliente no se entera de cómo esta organizado.

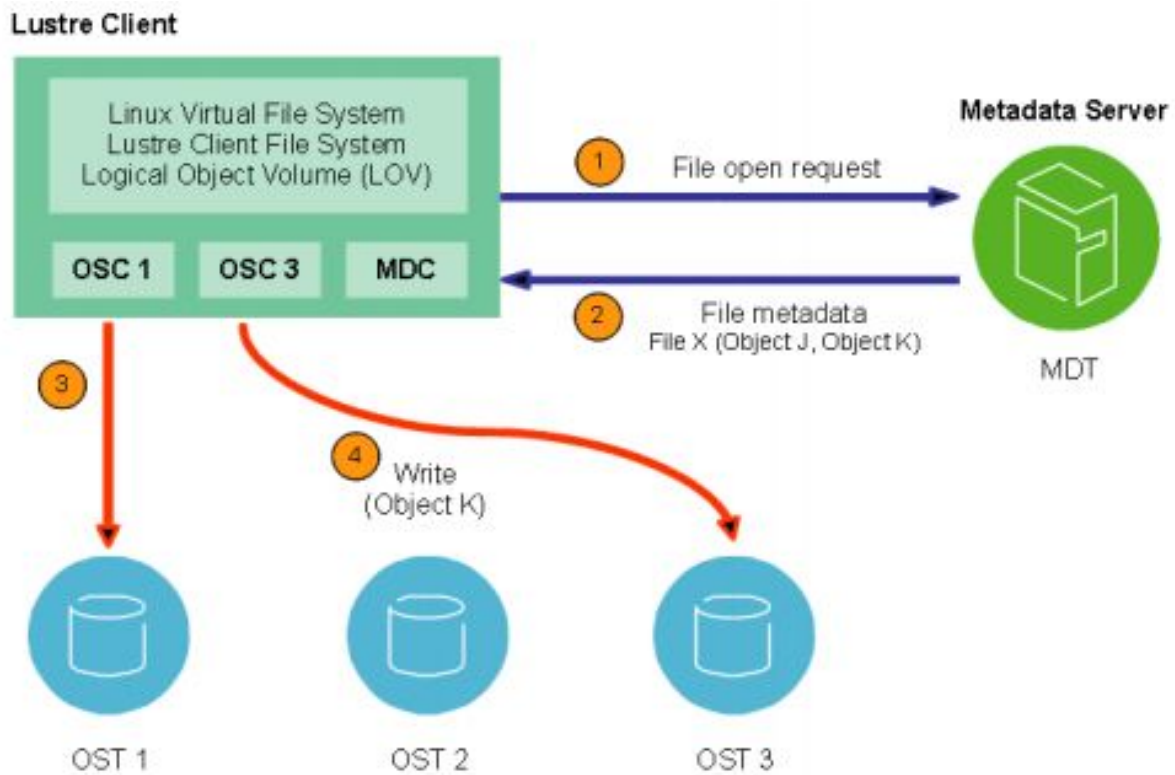
Funcionamiento de Lustre

En Lustre, un archivo de metadatos hace un seguimiento de los trozos en los cuales está partido un archivo determinado, los permisos que tiene, etc. Cada trozo contiene datos y está almacenado en un OST. Si el archivo de metadatos apunta a más de un objeto, significa que el archivo que el cliente necesita está distribuido entre varios OST al estilo RAID 0.



Cuando un cliente abre un archivo, la operación transfiere la estructura del archivo del MDS al cliente. El cliente utiliza esta información para realizar E/S en el archivo e interactúa directamente con los nodos de OSS donde se almacenan los objetos. Los clientes solicitan la disposición de los archivos del MDS y, a continuación, realizan operaciones de archivo de E/S mediante la comunicación directamente con el OSS que

gestionan los datos del archivo. En el diagrama siguiente se ilustran los pasos seguidos para obtener un archivo completo:



Fuentes:

[Manual de Lustre](#)

[Oracle - Lustre File System](#)

[Lustre.org](#)

[Nasa - Lustre best practices](#)

[Muy Linux - Oracle mantiene linux](#)