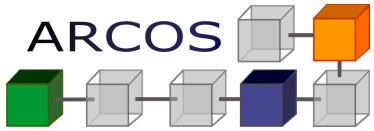


Localidad de datos en sistemas HPC utilizando MPI-IO y HDFS

José Rivadeneira

Félix García Carballeira

Jesús Carretero

Javier García Blas

Jornadas Sarteco

22-24 Septiembre 2021 Málaga - España

Este centro es beneficiario de ayudas para la realización de Programas de actividades de I+D entre grupos de investigación de la Comunidad de Madrid en Tecnologías 2018, cofinanciados con los Programas Operativos del Fondo Social Europeo y del Fondo Europeo de Desarrollo Regional, 2014-2020.

Referencia del programa: S2018/TCS-4423
Acrónimo: CABAHLA-CM

Índice

Background

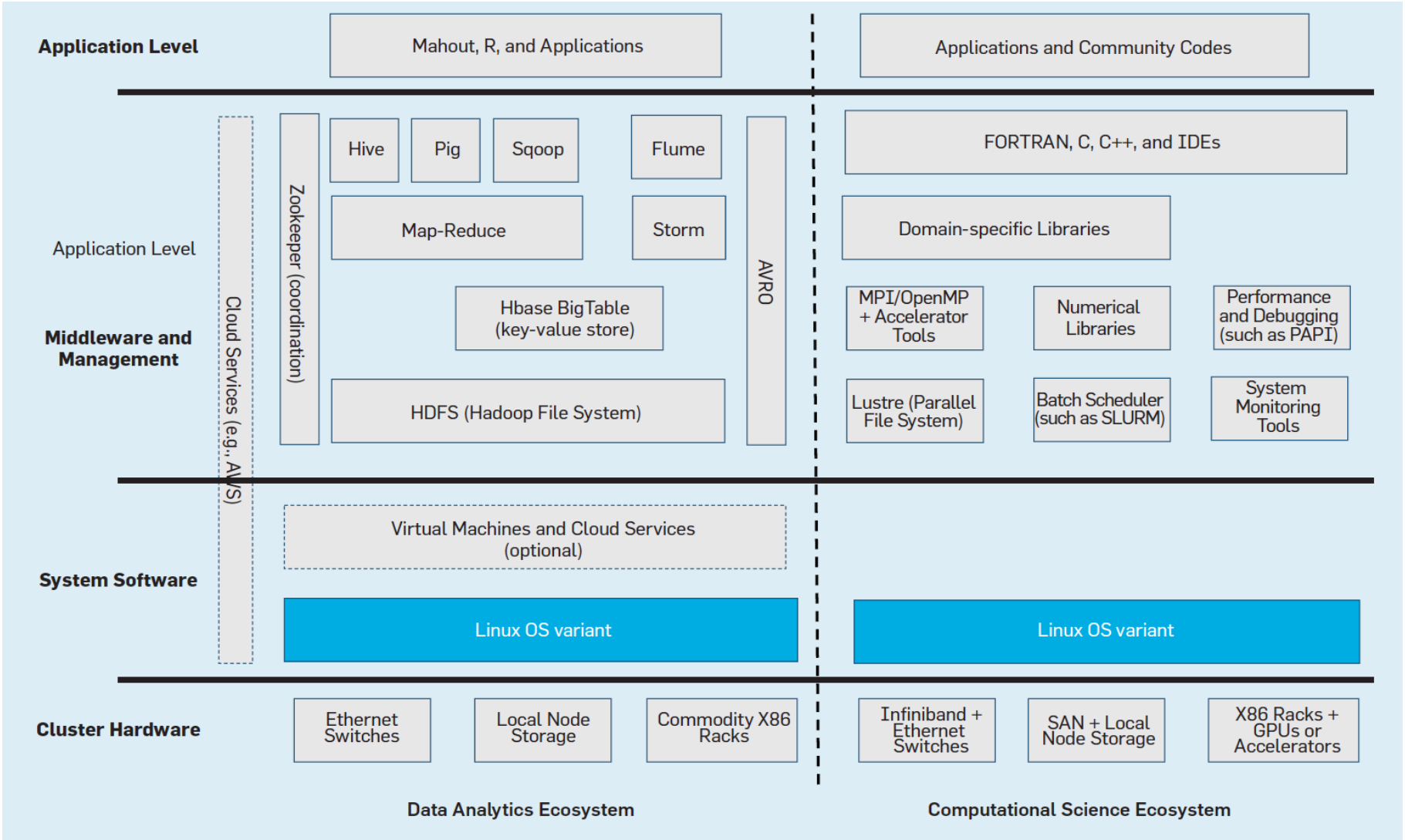


Imagen obtenida del artículo Exascale computing and Big Data (Daniel A. Reed and Jack Dongarra)

Diferencias del almacenamiento en Big Data y en HPC

Big Data

- Cada nodo de computo tiene un dispositivo de almacenamiento.
- Los datos se encuentran repartidos entre los nodos
- Cada trabajador conoce donde se encuentran almacenados los datos.

HPC

- Emplea nodos dedicados de entrada y salida.
- Los datos son enviados de los nodos de entrada y salida a los nodos en los que se va a procesar.
- Cada trabajador no sabe donde se encuentran almacenados los datos.

Motivación

Problema: Los frameworks actuales de Map Reduce no funcionan de forma óptima en los entornos de HPC.

¿Que queremos mejorar?

- Las aplicaciones de Map Reduce diseñadas para HPC haciendo uso de técnicas utilizadas en Big Data.

Objetivos

Mejorar las aplicaciones de Map Reduce para el entorno de HPC utilizando la localidad de los datos.

- **O1:** Extender la interfaz de MPI-IO para poder hacer uso de la localidad de los datos.
- **O2:** Crear un nuevo conector para sistemas de ficheros de BigData dentro de MPI-IO.
- **O3:** Usar nuestras propuestas en un framework de Map Reduce para HPC y evaluar el rendimiento.

Extensión de MPI-IO

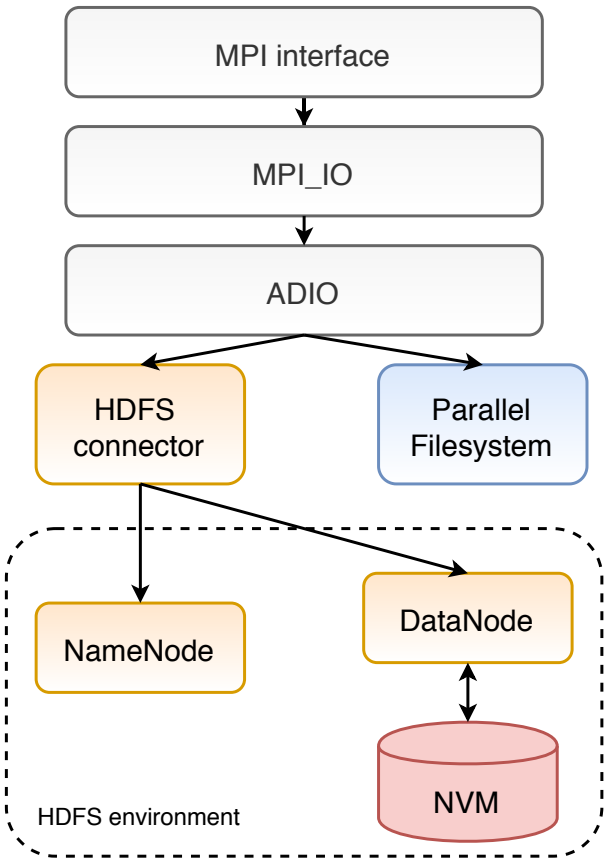
Con el objetivo de incluir la localidad de datos dentro de MPI-IO hemos propuesto dos nuevas funciones.

1. **MPIX_File_get_locality (MPI_File, MPI_Offset, MPI_Offset, char****, int *) :** Esta función devuelve la identidad del nodo(s) donde se encuentra almacenado el bloque indicado por argumento.

2. **MPIX_File_get_replication (MPI_File, int *) :** Esta función devuelve el número de réplicas almacenadas en el sistema de ficheros.

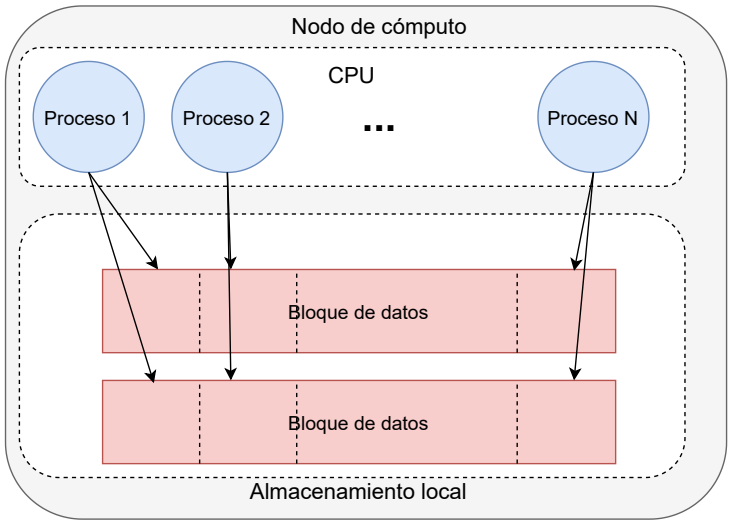
Añadiendo HDFS en MPI-IO

- Hemos diseñado un conector dentro de la interfaz abstracta para entrada y saluda paralela (ADIO).
- MPI_Info se pueden utilizar para personalizar como se crea un fichero y escribe un fichero.
- Nuestra solución implementa el modelo un escritor, múltiples lectores.



Integración de HDFS dentro Mimir

- Cada trabajador solo procesa los bloques que están almacenados localmente en la máquina.
- Si el bloque se encuentra almacenado en varios nodos se procesa por el nodo que tenga una menor carga de trabajo.
- La asignación de bloques se hace por nodo.



Experimentos

1 MPI-IO con HDFS

- Para evaluar la sobrecarga de nuestra propuesta contra el uso de HDFS con las funciones nativas escritas en C.

2 Integrando Mimir con HDFS

- Se han utilizado dos aplicaciones de BigData.
- **WordCount (wc):** Esta aplicación cuenta el número de ocurrencia de cada palabra de un fichero.
- **Protein matching application (pm):** Esta aplicación busca una determinada proteína en un gran conjunto de datos.

Sobrecarga de nuestra propuesta

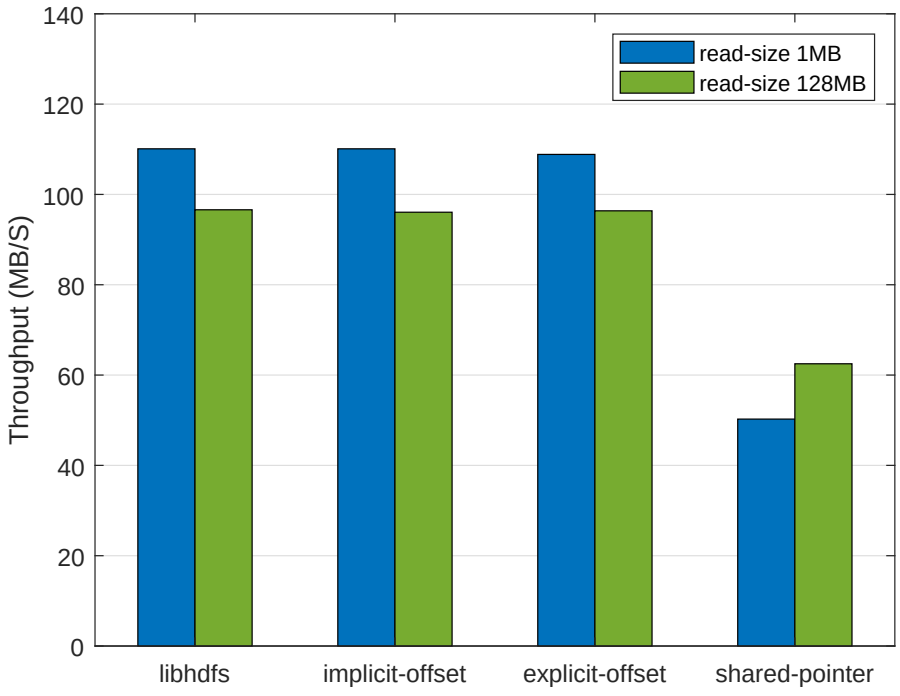


Figura: *MB/S lectura 32GB*

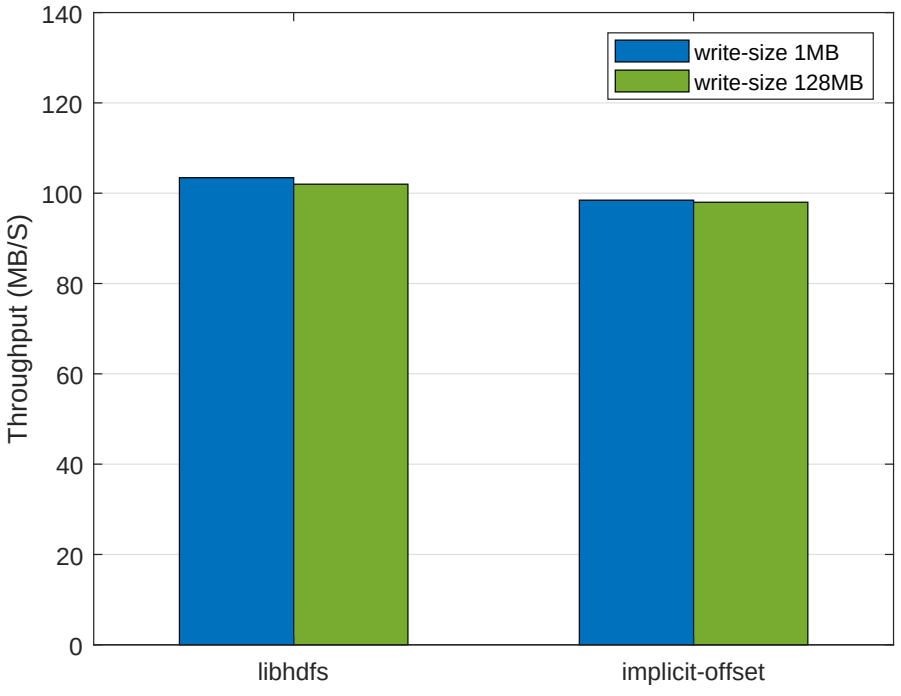
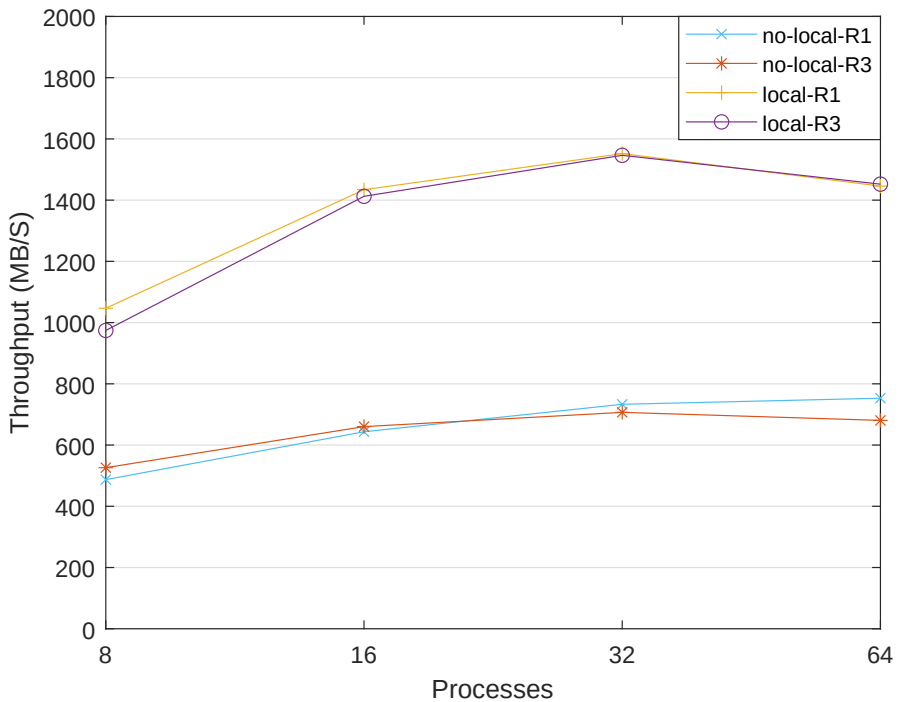
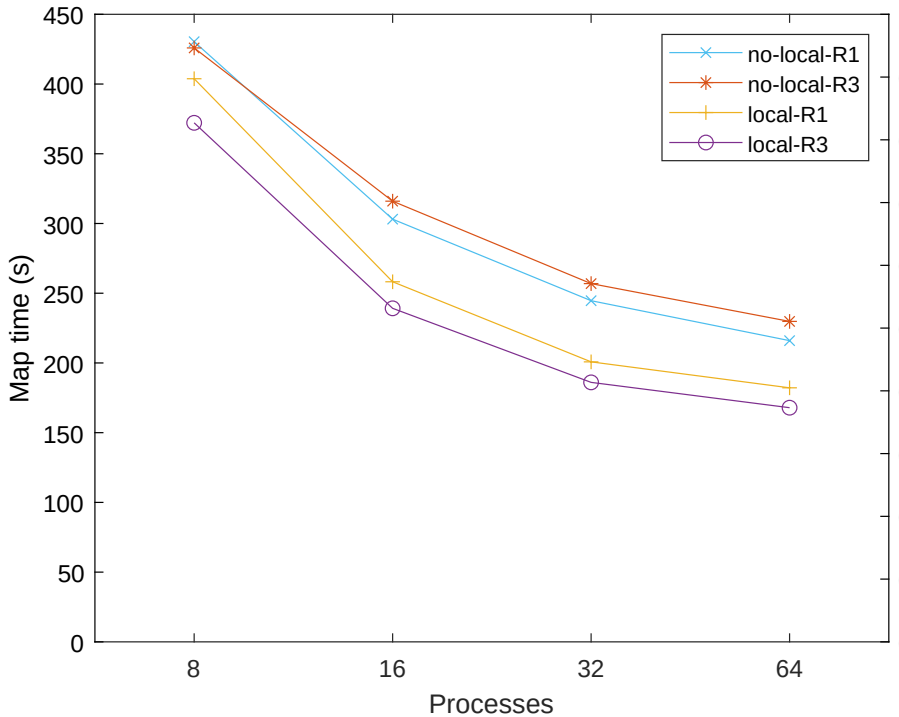


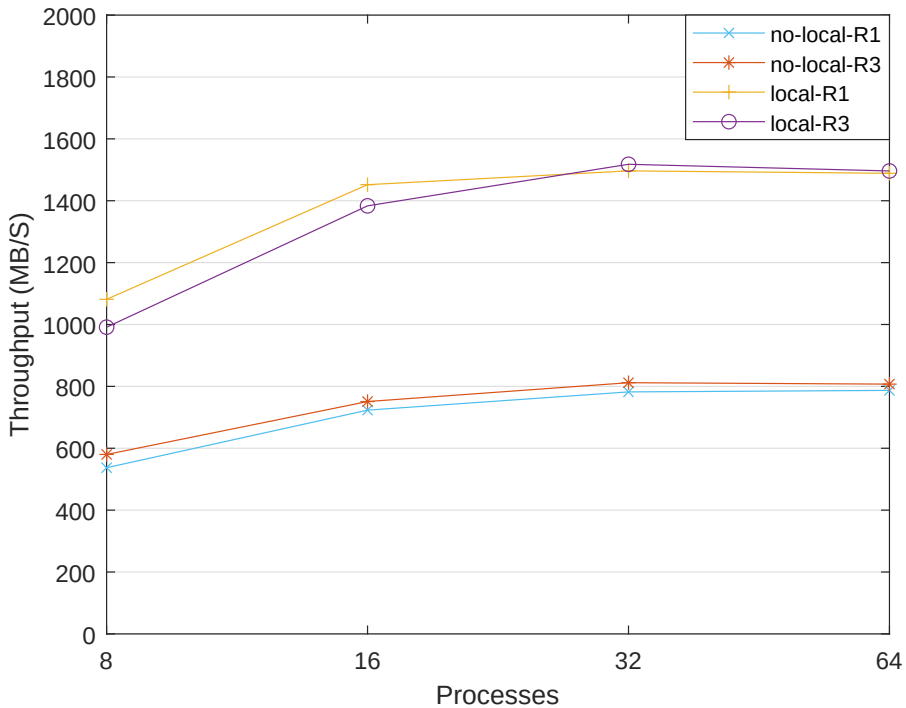
Figura: *MB/S escritura de 1GB*

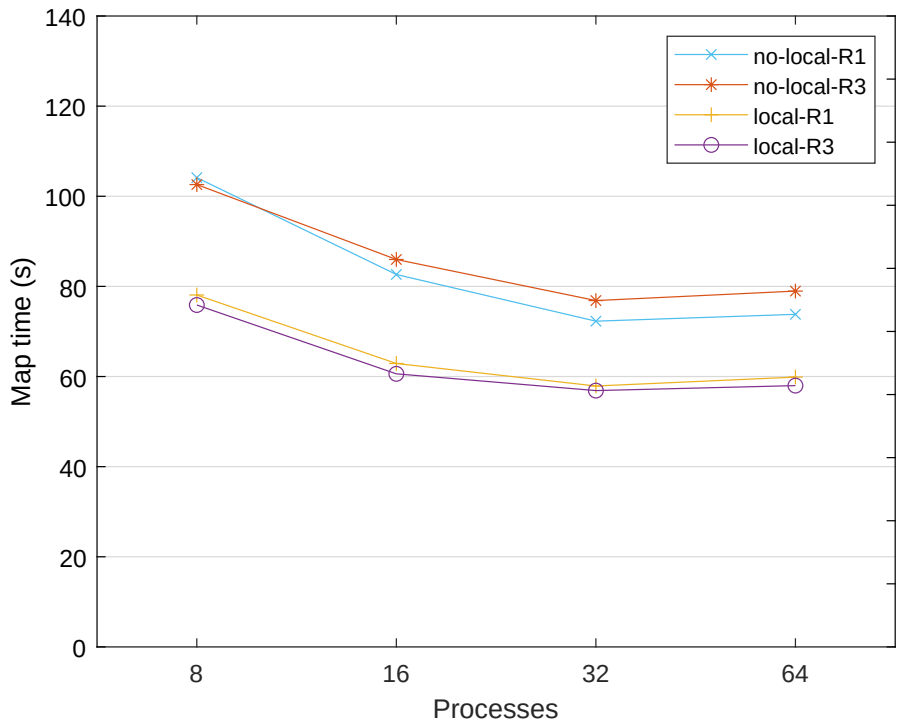
Rendimiento de la aplicación wordcount





Rendimiento de la aplicación protein matching





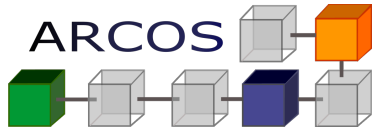
Conclusiones y trabajos futuros

Conclusiones

1. El framework de Mimir puede ser optimizados haciendo uso de la localidad de los datos.
2. La localidad de los datos puede reducir el tiempo de entrada y salida en los sistemas de HPC.

Trabajos futuros

1. Hacer una propuesta para incluir la funcionalidad en el estandar de MPI.
2. Crear nuevos conectores dentro de ADIO para proveer información de la localidad a otros sistemas de ficheros.



Contacta con nosotros

José Rivadeneira
jrivaden@pa.uc3m.es

Félix García Carballeira
felix.garcia@uc3m.es

Jesús Carretero
jesus.carretero@uc3m.es

Javier García Blas
fjblas@inf.uc3m.es