MPIIO - Tuning

LNCC
www.lncc.br
Laboratório Nacional de Computação Científica

Escola Supercomputador SDUMONT

Introdução E/S Paralela no SDUMONT
MPIIO - Tuning

André Ramos Carneiro
Bruno Alves Fagundes

MPIIO - Tuning

Roteiro:

- MPIIO
- ROMIO
- OMPIO

MPIIO

- Desenvolvido em 1994 no Laboratório Watson da IBM
  - 1996: Adotado pela NASA e incorporado pelo Fórum do MPI no MPI-2.
  - 1997: Publicação do MPI 2 com o MPI-IO já definido dentro.
- Fornecer suporte de Operações de E/S paralelas ao MPI
- Chamadas de função do MPI-IO -> chamadas MPI.
- Escrever arquivos MPI -> enviar mensagens MPI.
- Ler arquivos MPI -> receber mensagens MPI.

MPIIO - ROMIO

- Implementação do MPIIO
- Utilizada por qualquer implementação MPI
- Já faz parte do código da maioria das implementações
- Duas técnicas de otimização de desempenho

1) “Data Sieving”

- Focado em operações independentes com dados não contíguos
- Lê grandes blocos contíguos de dados e depois extrai as áreas de interesse

2) “Collective buffering” ou “Two phase I/O”

- Focado em operações coletivas
- Conjunto de processos (agregadores)
- Leem os dados do disco e distribuem para os demais processos
- Coletam os dados de todos os processos e os escrevem no disco

MPIIO – ROMIO - Otimização

- Hints
  - Forma de “ajudar” o desempenho da aplicação
- Data Sieving
  - ind_rd_buffer_size (BYTES)
  - ind_wr_buffer_size (BYTES)
  - romio_ds_read [enable, disable, **automatic**]
  - romio_ds_write [enable, disable, **automatic**]

MPIIO – ROMIO - Otimização

- Hints
  - Forma de “ajudar” o desempenho da aplicação
- Collective Buffering
  - cb_buffer_size: (BYTES)
  - cb_nodes: (Num nós/processos) → **Nº hosts únicos**
  - romio_cb_read: [enable, disable, **automatic**]
  - romio_cb_write: [enable, disable, **automatic**]
  - cb_config_list: Pode ser utilizado para um controle mais refinado
    - *:2 → Cada host utilizará 2 processos para realizar E/S
    - `sdumontXXXX:24,*:0` → O nó especificado utilizará 24 processos para serem agregadores. Os demais nós não serão utilizados.

MPIIO – ROMIO - Otimização

- Hints
  - Forma de “ajudar” o desempenho da aplicação
- Lustre
  - romio_lustre_co_ratio: [Int] **C**liente/**O**ST (**1**)
  - romio_lustre_coll_threshold: (BYTES) – limite para realizar E/S Col.
  - romio_lustre_cb_ds_threshold: (BYTES) - limite para realizar Sieving.

MPIIO – ROMIO - Otimização

- Hints
  - Forma de “ajudar” o desempenho da aplicação
- Utilizar
  - Um arquivo contendo:
    - `hint_1 valor`
    - `hint_2 valor`
    - `Etc....`
  - Configurar a variável de ambiente ROMIO_HINTS, apontando para o arquivo:
  - `export ROMIO_HINTS=/scratch/PROJETO/USUARIO/arquivo_de_hints`

MPIIO – ROMIO - Otimização

- Hints
  - Forma de “ajudar” o desempenho da aplicação
- BullxMPI (Exclusivo, através do MCA)
  - io_romio_optimize_stripe_count: define a estratégia do “stripe count”:
    - Negativo: deixa o romio decidir o melhor stripe count
    - 0: utiliza o stripe count definido no sistema de arquivos
    - Positivo: força o ROMIO utilizar o número máximo de stripe count (= número total de OSTs)
  - `export OMPI_MCA_io_romio_optimize_stripe_count=0`

MPIIO – ROMIO - Ativar

- BullxMPI e OpenMPI 1.10
  - Ativado por padrão
  - `export OMPI_MCA_io=romio`
- **Intel MPI (Ativa outras otimizações inerentes) *****
  - `export I_MPI_EXTRA_FILESYSTEM=on`
  - `export I_MPI_EXTRA_FILESYSTEM_LIST=lustre`
- OpenMPI 2.X
  - Ativado através do MCA = Modular Component Architecture
  - `export OMPI_MCA_io=romio314`

MPIIO – OMPIO

- Implementação desenvolvida pelo Open-MPI.org
- Introduzida a partir da versão 1.7 (default a partir da versão 2.0)
- Suporte ao Lustre a partir da versão 2.0

1) Aumenta a modularidade da biblioteca de E/S Paralela
2) Permite aos frameworks utilizarem diferentes algoritmos de decisão
3) Melhora a integração das funções de E/S paralelas

MPIIO – OMPIO

- fs framework: Gerenciamento de todas as operações com arquivos
- fbtl framework: Operações individuais de E/S blocking e non-blocking
- fcoll framework: Operações coletivas de E/S blocking e non-blocking
- sharedfp framework: Operações de arquivos compartilhados

MPIIO – OMPIO – Parâmetros

- `io_ompio_cycle_buffer_size: (BYTES)`
- `io_ompio_bytes_per_agg: (BYTES)`
- `io_ompio_num_aggregators: (INT)`
- `io_ompio_grouping_option: [1-6]`
- `fs_lustre_stripe_size: (BYTES)`
- `fs_lustre_stripe_width: (INT)`

- Cada framework possui sua própria lista de parâmetros
- Lista completa de todos os parâmetros:
  - `ompi_info –all`

MPIIO – OMPIO – Parâmetros

- Para utilizar : OpenMPI 2.X

```
export OMPI_MCA_io=ompio
export OMPI_MCA_io_ompio_bytes_per_agg=536870912
export OMPI_MCA_io_ompio_num_aggregators=5
export OMPI_MCA_fs_lustre_stripe_width=5
export OMPI_MCA_fs_lustre_stripe_size=5242880
```

MPIIO – OMPIO – Parâmetros

Dúvidas?