

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

- 1 Buts
- 2 Notions de base et Vocabulaire
- 3 Exemple de base: travail sur un dataset simple
- 4 Fonctionnalités avancées
- 5 Étude de cas : données CEA
- 6 Conclusion

Les problèmes

Sauvegarde des données : portabilité

- Description du contenu
- Format texte
- Précision ?

Suuegarde des données : compacité

- Pas d'en tête
- Format binaire (problème relecture)
- Documentation ?

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

Sauvegarde des données : portabilité

- Description du contenu
- Format texte
- Précision ?

Suvegarde des données : compacité

- Pas d'en tête
- Format binaire (problème relecture)
- Documentation ?

Caractéristiques de Hdf5

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

Qu'est-ce ?

- Format de fichier adapté aux données scientifiques
- API C, C++, Fortran 90 de manipulation de ces fichiers

Apports

- Stockage de données tableau, image, pleines, creuses
- Portabilité (pas de problèmes big/little endian)
- Compression des données
- Gestion entrées/sorties parallèles
- Description, structuration des données
- **Simplification** des routines d'entrée/sortie dans le code

Caractéristiques de Hdf5

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

Qu'est-ce ?

- Format de fichier adapté aux données scientifiques
- API C, C++, Fortran 90 de manipulation de ces fichiers

Apports

- Stockage de données tableau, image, pleines, creuses
- Portabilité (pas de problèmes big/little endian)
- Compression des données
- Gestion entrées/sorties parallèles
- Description, structuration des données
- **Simplification** des routines d'entrée/sortie dans le code

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

- 1 Buts
- 2 **Notions de base et Vocabulaire**
- 3 Exemple de base: travail sur un dataset simple
- 4 Fonctionnalités avancées
- 5 Étude de cas : données CEA
- 6 Conclusion

Organisation d'un fichier

Principe : contenu fichier comparable à une arborescence de système de fichier

Entités de base

- Groupe : structure permettant de rassembler plusieurs objets hdf5.
Associée à une description (meta-data)
- Dataset : tableau multidimensionnel de données élémentaires.
Associé à une description (meta-data)

Nommage

- / : racine du fichier hdf5 = groupe
- /gr1/dset1 : dataset dans un groupe

Organisation d'un fichier

Principe : contenu fichier comparable à une arborescence de système de fichier

Entités de base

- Groupe : structure permettant de rassembler plusieurs objets hdf5.
Associée à une description (meta-data)
- Dataset : tableau multidimensionnel de données élémentaires.
Associé à une description (meta-data)

Nommage

- / : racine du fichier hdf5 = groupe
- **/gr1/dset1** : dataset dans un groupe

Fonctionnement d'une écriture de dataset

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

Principales étapes

- Création d'un fichier
 - (Création d'un groupe dans ce fichier)
 - Description des **dimensions** des tableaux de données à sauvegarder = Création d'un **dataspace**
 - Création **dataset** en associant le chemin dans le fichier au dataspace
 - Accès des **données du code** correspondant au dataset et transfert
1 ligne de code
 - (Fermeture du groupe)
- Fermeture du fichier

Fonctionnement d'une écriture de dataset

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

Principales étapes

- Création d'un fichier
- (Création d'un groupe dans ce fichier)
- Description des **dimensions** des tableaux de données à sauvegarder = Création d'un **dataspace**
- Création **dataset** en associant le chemin dans le fichier au dataspace
- Accès des **données du code** correspondant au dataset et transfert
1 ligne de code
- (Fermeture du groupe)
- Fermeture du fichier

Fonctionnement d'une écriture de dataset

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

Principales étapes

- Création d'un fichier
- (Création d'un groupe dans ce fichier)
- Description des **dimensions** des tableaux de données à sauvegarder = Création d'un **dataspace**
- Création **dataset** en associant le chemin dans le fichier au dataspace
- Accès des **données du code** correspondant au dataset et transfert
1 ligne de code
- (Fermeture du groupe)
- Fermeture du fichier

Fonctionnement d'une écriture de dataset

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

Principales étapes

- Création d'un fichier
- (Création d'un groupe dans ce fichier)
- Description des **dimensions** des tableaux de données à sauvegarder = Création d'un **dataspace**
- Création **dataset** en associant le chemin dans le fichier au dataspace
 - Accès des **données du code** correspondant au dataset et transfert
1 ligne de code
- (Fermeture du groupe)
- Fermeture du fichier

Fonctionnement d'une écriture de dataset

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

Principales étapes

- Création d'un fichier
- (Création d'un groupe dans ce fichier)
- Description des **dimensions** des tableaux de données à sauvegarder = Création d'un **dataspace**
- Création **dataset** en associant le chemin dans le fichier au dataspace
- Accès des **données du code** correspondant au dataset et transfert
1 ligne de code
- (Fermeture du groupe)
- Fermeture du fichier

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

Familles de fonctions

- Manipulations de fichier : **h5f...**
- Opérations sur les groupes : **h5g...**
- Opérations sur les dataspace : **h5s...**
- Opérations sur les datasets : **h5d...**

API Fortran 90

- Module hdf5
- Ajout de H5_Open_f en début de code
- Ajout code de retour dans les paramètres
- Présence de paramètres optionnels

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

- 1 Buts
- 2 Notions de base et Vocabulaire
- 3 Exemple de base: travail sur un dataset simple**
- 4 Fonctionnalités avancées
- 5 Étude de cas : données CEA
- 6 Conclusion

Exemple de code - 01-write.c

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

En pratique

- Écriture d'un dataset comprenant uniquement des entiers
- Affichage du contenu du fichier

Apports ?

- Quoi de plus qu'un **write** ?
- Relecture du fichier de données avec un programme externe
- Fonctions HDF5 spécifiques des datasets

Exemple de code - 01-write.c

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

En pratique

- Écriture d'un dataset comprenant uniquement des entiers
- Affichage du contenu du fichier

Apports ?

- Quoi de plus qu'un **write** ?
- Relecture du fichier de données avec un programme externe
- Fonctions d'interrogation de contenu.

Exemple de code - 01-write.c

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

En pratique

- Écriture d'un dataset comprenant uniquement des entiers
- Affichage du contenu du fichier

Apports ?

- Quoi de plus qu'un **write** ?
- Relecture du fichier de données avec un programme externe
- Fonctions d'interrogation de contenu.

Exemple de code - 01-write.c

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

En pratique

- Écriture d'un dataset comprenant uniquement des entiers
- Affichage du contenu du fichier

Apports ?

- Quoi de plus qu'un **write** ?
- Relecture du fichier de données avec un programme externe
- Fonctions d'interrogation de contenu.

Exemple de code lecture - 01-read.c

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

En pratique

- Lecture du dataset précédemment écrit
- Affichage du contenu du fichier

Apports ?

- Quoi de plus qu'un **read** ?
 - Portabilité entre architectures
 - Fonctions d'arrangement de contenu

Exemple de code lecture - 01-read.c

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

En pratique

- Lecture du dataset précédemment écrit
- Affichage du contenu du fichier

Apports ?

- Quoi de plus qu'un **read** ?
 - Portabilité entre architectures
 - Fonctions d'interrogation de contenu.

Exemple de code lecture - 01-read.c

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

En pratique

- Lecture du dataset précédemment écrit
- Affichage du contenu du fichier

Apports ?

- Quoi de plus qu'un **read** ?
- Portabilité entre architectures
- Fonctions d'interrogation de contenu.

Exemple de code lecture - 01-read.c

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

En pratique

- Lecture du dataset précédemment écrit
- Affichage du contenu du fichier

Apports ?

- Quoi de plus qu'un **read** ?
- Portabilité entre architectures
- Fonctions d'interrogation de contenu.

Lecture à partir de rien

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

En pratique 01-complete-read.c

- Ouverture d'un fichier
- Parcours des groupes à partir de /
- Détermination des caractéristiques des datasets
- Transfert des données
- Accès aux données en mémoire

Autres exemple de lecture

- Même principe de fonctionnement
- Programme h5dump (debug)
- hdfview
- Attention : ordre des indices dans le tableau

Lecture à partir de rien

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

En pratique 01-complete-read.c

- Ouverture d'un fichier
- Parcours des groupes à partir de /
- Détermination des caractéristiques des datasets
- Transfert des données
- Accès aux données en mémoire

Autres exemple de lecture

- Même principe de fonctionnement
- Programme h5dump (debug)
- hdfview
- Attention : ordre des indices dans le tableau

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

- 1 Buts
- 2 Notions de base et Vocabulaire
- 3 Exemple de base: travail sur un dataset simple
- 4 Fonctionnalités avancées**
- 5 Étude de cas : données CEA
- 6 Conclusion

Utilisation

- Types caractérisés par leur représentation mémoire.
- 152 Types
- Utiliser les natifs de la machine `HDF5_NATIVE_DOUBLE` par exemple
- Lors de l'écriture dans le fichier, un type explicite sera utilisé par hdf5 (spécification complète taille, endianness, types spécifiques)

Exemple de code - 02-read-array.c

- explication paramètres `H5DRead`
- La spécification complète = spécification du transfert
- Permet de lire des tranches de tableaux

Utilisation

- Types caractérisés par leur représentation mémoire.
- 152 Types
- Utiliser les natifs de la machine `HDF5_NATIVE_DOUBLE` par exemple
- Lors de l'écriture dans le fichier, un type explicite sera utilisé par hdf5 (spécification complète taille, endianness, types spécifiques)

Exemple de code - 02-read-array.c

- explication paramètres `H5DRead`
- La spécification complète = spécification du transfert
- Permet de lire des tranches de tableaux

Sélection de données - 02-simple-read-slice.c

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

Dans le code ...

- Tableau 3D volumineux stocké dans le fichier
- Stockage d'une tranche en mémoire
- Dataspace source = le tableau
- Dataspace destination = la tranche

Dans le code ...

- Tableau 3D volumineux stocké dans le fichier
- Stockage d'une tranche en mémoire
- Dataspace source = le tableau
- Dataspace destination = la tranche

Sélection de données - 02-simple-read-slice.c

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

Dans le code ...

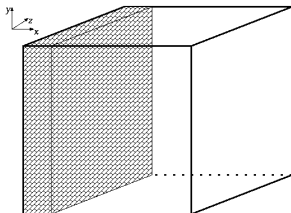
- Tableau 3D volumineux stocké dans le fichier
- Stockage d'une tranche en mémoire
- Dataspace source = le tableau
- Dataspace destination = la tranche

Dans le code ...

- Tableau 3D volumineux stocké dans le fichier
- Stockage d'une tranche en mémoire
- Dataspace source = le tableau
- Dataspace destination = la tranche

Exemple

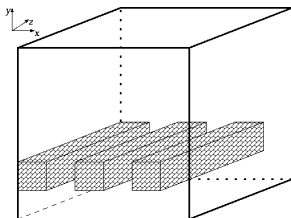
- explication paramètres H5DRead
- La spécification complète = spécification du transfert
- Permet de lire des tranches de tableaux



Stride 1,1,1 Block 1 × DIMY × DIMZ

Exemple

- explication paramètres H5DRead
- La spécification complète = spécification du transfert
- Permet de lire des tranches de tableaux



Stride 2,1,1 Block $\text{DIMX}/2 \times 1 \times \text{DIMY}$
Start 0,1,0

Possibilités

- Utilisation de types non élémentaires
- COMPOUND : structures du C
- tableau de types pré-existants où déjà définis

Fonctionnement

- Création + nommage
- Description de la configuration du type en mémoire
- Enregistrement du type
- Fermeture après utilisation

Possibilités

- Utilisation de types non élémentaires
- COMPOUND : structures du C
- tableau de types pré-existants où déjà définis

Fonctionnement

- Création + nommage
- Description de la configuration du type en mémoire
- Enregistrement du type
- Fermeture après utilisation

Exemple : type structuré - 02-write-structured.c

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

Dans le code

- Taille totale du type
- Type des composantes + Emplacement mémoire
- Validation
- Report du type dans le write. Aucun changement dataspace

Report des données dans h5dump

- AFFICHAGE SORTIE h5DUMP
- Apparition nouveau type
- Dump correct éléments

Exemple : type structuré - 02-write-structured.c

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

Dans le code

- Taille totale du type
- Type des composantes + Emplacement mémoire
- Validation
- Report du type dans le write. Aucun changement dataspace

Report des données dans h5dump

- AFFICHAGE SORTIE h5DUMP
- Apparition nouveau type
- Dump correct éléments

Compression de données

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

Principes

- Ajout de **propriétés** lors de la création du dataset
- Deux mécanismes de compression SZIP et ZLIB
- Partition des données en blocs avant compression
- Paramétrage de la compression

Exemple

- Définition des propriétés
- Paramétrage : Famille de fonctions h5p
- Utilisation des propriétés lors du transfert
- **Aucun** changement à la lecture

Compression de données

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

Principes

- Ajout de **propriétés** lors de la création du dataset
- Deux mécanismes de compression SZIP et ZLIB
- Partition des données en blocs avant compression
- Paramétrage de la compression

Exemple

- Définition des propriétés
- Paramétrage : Famille de fonctions h5p
- Utilisation des propriétés lors du transfert
- **Aucun** changement à la lecture

Entrées/Sorties parallèles - 02-parallel-io.c

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

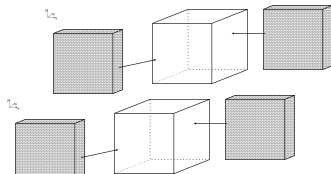
Conclusion

Principes

- Définition d'un modèle d'accès (access template)
- Association à la couche basse utilisée pour les I/O
- Définition d'un hyperslab par processus : où sont les données de mon processus

Dans le code

- Volume distribué sur les processus
- Chaque processus : responsable d'un plan



Stride 1,1,nb_procs, Start 1 × 1 × ra

Entrées/Sorties parallèles - 02-parallel-io.c

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

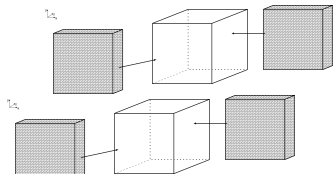
Conclusion

Principes

- Définition d'un modèle d'accès (access template)
- Association à la couche basse utilisée pour les I/O
- Définition d'un hyperslab par processus : où sont les données de mon processus

Dans le code

- Volume distribué sur les processus
- Chaque processus : responsable d'un plan



Stride 1,1,nb_procs, Start 1 × 1 × ra

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

- 1 Buts
- 2 Notions de base et Vocabulaire
- 3 Exemple de base: travail sur un dataset simple
- 4 Fonctionnalités avancées
- 5 Étude de cas : données CEA**
- 6 Conclusion

Présentation du problème

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

Situation initiale

- Machines de calcul : cluster
- 1 diagnostic par pas de temps
- Temps important consacré aux E/S pendant le calcul (NFS)
- Volume de données important (jusqu'à quelques TO pour une simulation importante)
- Compression efficace (ratio de l'ordre de 10)

Nécessités

- Prévoir le stockage local pendant le calcul
- Mécanismes de ré-assemblage des fichiers

Présentation du problème

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

Situation initiale

- Machines de calcul : cluster
- 1 diagnostic par pas de temps
- Temps important consacré aux E/S pendant le calcul (NFS)
- Volume de données important (jusqu'à quelques TO pour une simulation importante)
- Compression efficace (ratio de l'ordre de 10)

Nécessités

- Prévoir le stockage local pendant le calcul
- Mécanismes de ré-assemblage des fichiers

Méthode utilisée

- Tous les fichiers sont écrits en hdf5 + compression
- Utilisation des **liens** entre fichiers de hdf5
- Nombre de processeurs variable suivant les runs
- Lien : publication des groupes d'un autre fichier dans un groupe G
- Accès à ce fichier externe via G + nom datasets

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

À la relecture

- Création d'un fichier **squelette**
- Accès aux datasets depuis le fichier squelette
- Hdf5 se charge de l'accès physique
- Décomposition de domaine à intégrer dans l'appli de post-traitement
- Permet de limiter l'espace mémoire nécessaire

Buts

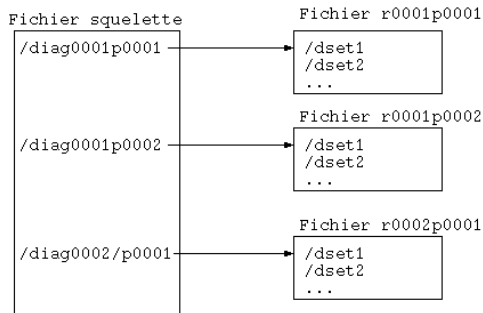
Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion



Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

- 1 Buts
- 2 Notions de base et Vocabulaire
- 3 Exemple de base: travail sur un dataset simple
- 4 Fonctionnalités avancées
- 5 Étude de cas : données CEA
- 6 Conclusion**

Buts

Notions de
base et
Vocabulaire

Exemple de
base: travail
sur un dataset
simple

Fonctionnalités
avancées

Étude de cas :
données CEA

Conclusion

Apports de HDF5

- API de sauvegarde couvrant de larges besoins : sélection, compression, données structurées
- Apport de la structuration des données
- Lecture sans connaissance à priori du contenu du fichier
- Apprentissage nécessaire pour description des données

Autres formats

- Fits : format de données issu de l'astronomie, orienté images multi-spectrales Pas de définition de types structurés
- NetCdf 4 utilisera hdf5 pour le stockage. Version actuelle : 3.6 Nombreuses applications à base de ce format. Orienté tableaux
- XML/XSIL : structuré mais représentation texte. Projets d'association formats binaires et XML, réf à des fichiers binaires en xml