

H5P_SET_ALL_COLL_METADATA_OPS

[Expand all](#) [Collapse all](#)

- [Jump to ...](#)
- [Summary](#)
- [Description](#)
- [Example](#)
- [Switch language ...](#)
- [C](#)
- [C++](#)
- [FORTRAN](#)
- [JAVA](#)

[Summary](#)
[Description](#)
[Example](#)
[JAVA](#)
[FORTRAN](#)
[C++](#)
[C](#)

H5P_SET_ALL_COLL_METADATA_OPS

Sets metadata I/O mode for read operations to collective or independent (default)

Procedure:

H5P_SET_ALL_COLL_METADATA_OPS (accpl_id, is_collective)

Signature:

```
herr_t H5Pset_all_coll_metadata_ops(  
    hid_t accpl_id,  
    hbool_t is_collective  
)
```

Fortran Interface: h5pset_all_coll_metadata_ops_f

Signature:

```
SUBROUTINE h5pset_all_coll_metadata_ops_f(plist_id, is_collective, hdferr)  
    INTEGER(HID_T) , INTENT(IN)  :: plist_id  
    LOGICAL, INTENT(IN)          :: is_collective  
    INTEGER, INTENT(OUT)         :: hdferr
```

Inputs:

plist_id - File access property list identifier.
is_collective - Indicates if metadata writes are collective or not.

Outputs:

hdferr - Returns 0 if successful and -1 if fails.

Parameters:

<code>hid_t accpl_id</code>	IN: File, group, dataset, datatype, link, or attribute access property list identifier
<code>hbool_t is_collective</code>	IN: Boolean value indicating whether metadata reads are collective (<code>TRUE</code>) or independent (<code>FALSE</code>) <i>Default mode: Independent (FALSE)</i>

Description:

`H5P_SET_ALL_COLL_METADATA_OPS` sets the metadata I/O mode for read operations in the access property list `accpl`.

When engaging in parallel I/O, all metadata write operations must be collective. If `is_collective` is `TRUE`, this property specifies that the HDF5 library will perform all metadata read operations collectively; if `is_collective` is `FALSE`, such operations may be performed independently.

Users must be aware that several HDF5 operations can potentially issue metadata reads. These include opening a dataset, datatype, or group; reading an attribute; or issuing a *get info* call such as getting information for a group with `H5F_GET_INFO`. Collective I/O requirements must be kept in mind when issuing such calls in the context of parallel I/O.

If this property is set to true on a file access property list that is used in creating or opening a file, then the HDF5 library will assume that all metadata read operations issued on that file identifier will be issued collectively from all ranks irrespective of the individual setting of a particular operation. If this assumption is not adhered to, corruption will be introduced in the metadata cache and HDF5's behavior will be undefined.

Alternatively, a user may wish to avoid setting this property globally on the file access property list, and individually set it on particular object access property lists (dataset, group, link, datatype, attribute access property lists) for certain operations. This will indicate that only the operations issued with such an access property list will be called collectively and other operations may potentially be called independently. There are, however, several HDF5 operations that can issue metadata reads but have no property list in their function signatures to allow passing the collective requirement property. For those operations, the only option is to set the global collective requirement property on the file access property list; otherwise the metadata reads that can be triggered from those operations will be done independently by each process.

Functions that do not accommodate an access property list but that might issue metadata reads are listed in [Functions with No Access Property List Parameter that May Generate Metadata Reads](#).

As noted above, corruption will be introduced into the metadata cache and HDF5 library behavior will be undefined when both of the following conditions exist:

- A file is created or opened with a file access property list in which the collective metadata I/O property is set to `TRUE`.
- Any function is called that triggers an independent metadata read while the file remains open with that file access property list.

An approach that avoids this corruption risk is described above.

See Also:

- [H5P_GET_ALL_COLL_METADATA_OPS](#)
- [H5P_GET_COLL_METADATA_WRITE](#)
- [H5P_SET_ALL_COLL_METADATA_OPS](#)
- [H5P_SET_COLL_METADATA_WRITE](#)
- [Functions with No Access Property List Parameter that May Generate Metadata Reads](#)

Returns:

Returns a non-negative value if successful; otherwise returns a negative value.

Example:

Coming Soon!

History:

Release	Change
1.10.0	C function and Fortran wrapper introduced with this release.

--- Last Modified: August 12, 2019 | 09:31 AM