

# H5O\_ENABLE\_MDC\_FLUSHES

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

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

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

## H5O\_ENABLE\_MDC\_FLUSHES

Enables flushing of dirty metadata entries from a file's metadata cache

### Procedure:

H5O\_ENABLE\_MDC\_FLUSHES(object\_id)

### Signature:

```
herr_t H5Oenable_mdc_flushes( hid_t object_id )
```

### Parameters:

<code>hid_t object_id</code>	IN: Identifier of the object that will have flushes re-enabled; may be a group, named datatype, or dataset identifier
------------------------------	---

### Description:

The H5O\_/H5F\_ ENABLE/DISABLE\_MDC\_FLUSHES and associated H5X\_FLUSH functions can be used to control the flushing of entries from a file's metadata cache.

This function allows an object or cache's dirty metadata entries to be flushed from the cache by the usual cache eviction/flush policy.

Metadata cache entries can be controlled at both the individual HDF5 object level (datasets, groups, committed datatypes) and the entire metadata cache level.

HDF5 objects include datasets, groups, and committed datatypes. Only `hid_t` identifiers that represent these objects can be passed to

the function.

Passing in an `hid_t` identifier that represents any other HDF5 entity is considered an error.

It is an error to pass an HDF5 file identifier (obtained from `H5F_OPEN` or `H5F_CREATE`) to this function. Use `H5F_ENABLE_MDC_FLUSHES` instead.

Using this function on an object that has not had flushes disabled is considered an error. The state of an object can be determined with `H5O_ARE_FLUSHES_DISABLED`.

Individual objects can be returned to the default flush algorithm with this function after `H5F_DISABLE_MDC_FLUSHES` has been used to globally prevent flushes.

An object will be returned to the default flush algorithm when it is closed.

All objects will be returned to the default flush algorithm when the file is closed.

An object's entries will not necessarily be flushed as a result of calling this function.

### Returns:

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

### Example:

Coming Soon!

### History:

Release	Change
1.10.0	C function introduced with this release.

--- Last Modified: April 25, 2019 | 01:19 PM