Intel® Optane™ Memory pinning SDK

Reference Guide

For the Intel® Rapid Storage Technology (Intel® RST) Release Version 16.7.x with Intel® Optane™ Memory Support

Revision 1.0

You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning Intel products described herein. You agree to grant Intel a non-exclusive, royalty-free license to any patent claim thereafter drafted which includes subject matter disclosed herein.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at Intel.com, or from the OEM or retailer.

No computer system can be absolutely secure. Intel does not assume any liability for lost or stolen data or systems or any damages resulting from such losses.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at intel.com, or from the OEM or retailer.

All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest Intel product specifications and roadmaps.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-5484725 or visit www.intel.com/design/literature.htm.

By using this document, in addition to any agreements you have with Intel, you accept the terms set forth below.

Contact your local Intel sales office or your distributor to obtain the latest specifications before placing your product order.

Contents

§ §

Revision History

Revision Number	Description	Revision Date
1.0	Initial Revision	October2018

§ §

1. About this Document

This document will include Intel® Optane™ Memory File Cache API, which allows software vendors to improve their application performance by placing most frequently accessed files into Intel® Optane™ Memory. The provided API allows application to define and change the working set of files and fetch ahead defined set of files into the Intel® Optane™ Memory.

2. Intel® Optane™ Memory Pinning API Usage

Typical usage of Intel® Optane™ memory API usage will start from initialization phases in which availability of pinning API and API version is checked. For more information about initialization phase see section 2.1. If initialization phase is successful files for which fast access is required can be pinned and fetched into Intel® Optane™ Memory. This process is described in section 2.2. At the end of the application lifetime it is recommended to unpin files. This will allow to faster release Intel® Optane™ Memory for other applications.

2.1 API Initialization

Initialization phase would typically consists of two steps.
The first step shall be call to OptaneGetPinningStatus in order to check if pinning API is available on the platform.
The second step is call to OptaneGetPinningApiVersion in order to retrieve API version provided by Optane.dll and API version supported by platform (installed driver). If API version is appropriate (see OptaneGetPinningApiVersion for more information) then initialization phase is completed successfully.

2.2 Defining and Fetching Current Working Set

The set of files which application needs to accelerate in the near future can be called current working set. While application is running the current working set may change dynamically and it can defined/redefined using pin and unpin operations. Application shall pin/unpin files that shall be added to/removed from the current working set. The pinning and unpinning operations are performed using OptanePinFiles and OptaneUnpinFiles methods respectively. While defining current working set application may call these methods multiple times in arbitrary order. However the both calls take list of files and it is recommended to group the files in pin and unpin list and this way reduce this way the number of API calls.
Once all needed files are pinned/unpinned the new current working set is defined. In order to prefetch current working set into Intel® Optane™ Memory, application shall execute OptaneTriggerMigration method.
The following sequence diagram presents scenario where application defines its current working and redefines it when such a need appears. Please note that every time the working set is changed OptaneTriggerMigration is called to fetch appropriate files into Intel® Optane™ Memory. It is highly recommended to keep the current working set up to date especially unpin files that will

be no longer needed in foreseeable future. Such a behavior allows to better manage content of Intel® Optane™ Memory.

2.3 API Deinitialization

In order to quickly release resources related with cached files it is highly recommended to unpin all files from the current working set when application is closing.

3. Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

4. File Index

4.1 File List

Here is a list of all documented files with brief descriptions:

5. Class Documentation

5.1 OPTANE_FILE Struct Reference

Public Attributes

- **OPTANERETCODE fileError**
  Will be set to error if file operation failed, otherwise OPTANE_SUCCESS.
- const wchar_t * **path**
  File path.

The documentation for this struct was generated from the following file:

- SSE_Storage_NGSA/Implementation/source/common/OptaneApiCommon/**OptaneTypes.h**

5.2 OPTANE_FILECACHESTATUS Struct Reference

Public Attributes

- unsigned __int64 **fileCacheSize**
- unsigned __int64 **usedSpace**
- unsigned __int64 **nrCachedFiles**
- unsigned __int64 **nrUserPinnedFiles**
  Number of all files pinned by the user.
- unsigned __int64 **nrUserCachedFiles**
  Number of files pinned by the user and located in the file cache.
- unsigned __int64 **sizeUserCachedFiles**
  Size [B] of files pinned by user and located in the file cache.
- unsigned __int64 **sizeUserUnCachedFiles**
  Size [B] of files pinned by the user and located on the slow drive.
- unsigned __int64 **nrAppPinnedFiles**
  Number of files pinned by software vendor Applications and user pinned applications.
- unsigned __int64 **nrAppCachedFiles**
- unsigned __int64 **sizeAppCachedFiles**
- unsigned __int64 **sizeAppUnCachedFiles**
- unsigned __int64 **nrAutoCachedFiles**
- unsigned __int64 **sizeAutoCacheFiles**
- unsigned __int64 **sizeSystemCacheFiles**

Member Data Documentation

unsigned __int64 OPTANE_FILECACHESTATUS::fileCacheSize

Size [B] of the file cache (size of the fast drive minus block cache size)

unsigned __int64 OPTANE_FILECACHESTATUS::nrAppCachedFiles

Number of files pinned by software vendor Applications and user pinned applications located in the file cache

unsigned __int64 OPTANE_FILECACHESTATUS::nrAutoCachedFiles

Number of files migrated to the file cache as a result of Optane algorithms and polices

unsigned __int64 OPTANE_FILECACHESTATUS::nrCachedFiles

Number of files in the file cache (including partially cached files)

unsigned __int64 OPTANE_FILECACHESTATUS::sizeAppCachedFiles

Size [B] of files pinned by software vendor Applications and user pinned applications located in the file cache

unsigned __int64 OPTANE_FILECACHESTATUS::sizeAppUnCachedFiles

Size [B] of files pinned by software vendor Applications and user pinned applications located on the slow drive

unsigned __int64 OPTANE_FILECACHESTATUS::sizeAutoCacheFiles

Size [B] of files migrated to the file cache as a result of Optane algorithms and polices

unsigned __int64 OPTANE_FILECACHESTATUS::sizeSystemCacheFiles

Size [B] of system files migrated to the file cache as a result of NGSA algorithms and polices

unsigned __int64 OPTANE_FILECACHESTATUS::usedSpace

Total size [B] of space that is occupied by files in the file cache (including system files)

The documentation for this struct was generated from the following file:

- SSE_Storage_NGSA/Implementation/source/common/OptaneApiCommon/**OptaneTypes.h**

5.3 OPTANE_FILELIST Struct Reference

Public Attributes

- unsigned int **size**
  Number of files to pin or unpin.
- **OPTANE_FILE** * **data**
  List of paths to pin or unpin with per-file errors.

The documentation for this struct was generated from the following file:

- SSE_Storage_NGSA/Implementation/source/common/OptaneApiCommon/**OptaneTypes.h**

5.4 OPTANE_FILEPINSTATUS Struct Reference

Public Attributes

- **OPTANERETCODE pathError**
  Operation return code for file pointed by provided path.
- const wchar_t * **path**
  Full path to file.
- **OPTANE_PINNING_STATE cachingState**
  File caching status.
- **OPTANE_CACHING_STATUS locationState**
  File location status.

The documentation for this struct was generated from the following file:

- SSE_Storage_NGSA/Implementation/source/common/OptaneApiCommon/**OptaneTypes.h**

5.5 OPTANE_FILEPINSTATUSLIST Struct Reference

Public Attributes

- unsigned int **size**
  Number of files/entries provided in listOfFileStatus array.
- **OPTANE_FILEPINSTATUS** * **listOfFileStatus**
  Pointer to array of file pinning status structures.

The documentation for this struct was generated from the following file:

- SSE_Storage_NGSA/Implementation/source/common/OptaneApiCommon/**OptaneTypes.h**

5.6 OPTANE_PINNINGAPIVERSION Struct Reference

Public Attributes

- unsigned long **major**
- unsigned long **minor**
- unsigned long **platformMajor**
  Pinning API version supported by platform ï¿½ major number.
- unsigned long **platformMinor**
  Pinning API version supported by platform ï¿½ minor number.

Member Data Documentation

unsigned long OPTANE_PINNINGAPIVERSION::major

Optane Pinning API version supported by DLL ï¿½ major number. No backwards-compatible API changes. Removing obsolete function or data type for example.

unsigned long OPTANE_PINNINGAPIVERSION::minor

Optane Pinning API version supported by DLL ï¿½ minor number. New functionality added in a backward compatible manner. Bug fixes, adding new function or data type but still supporting old one.

The documentation for this struct was generated from the following file:

- SSE_Storage_NGSA/Implementation/source/common/OptaneApiCommon/**OptaneTypes.h**

5.7 OPTANE_PINNINGINFO Struct Reference

Public Attributes

- **OPTANE_PINNING_CAPABILITY pinningStatus**
  Status of pinning (if it is avaible)
- wchar_t **optaneVolumeName** [OPTANE_VOL_NAME_SZ]

Member Data Documentation

wchar_t OPTANE_PINNINGINFO::optaneVolumeName[OPTANE_VOL_NAME_SZ]

Optane volume name. This parameter has valid value when Optane pinning state is OPTANE_PINNING_ACTIVE.

The documentation for this struct was generated from the following file:

- SSE_Storage_NGSA/Implementation/source/common/OptaneApiCommon/**OptaneTypes.h**

6. File Documentation

SSE_Storage_NGSA/Implementation/source/common/OptaneApiCommon/OptaneTypes.h File Reference

Classes

- struct **OPTANE_PINNINGINFO**
- struct **OPTANE_FILE**
- struct **OPTANE_FILELIST**
- struct **OPTANE_FILEPINSTATUS**
- struct **OPTANE_FILEPINSTATUSLIST**
- struct **OPTANE_FILECACHESTATUS**
- struct **OPTANE_PINNINGAPIVERSION**

Macros

- #define **OPTANE_VOL_NAME_SZ** 8
- #define **OPTANE_SUPPORTED_APPS_DIR** L"SupportedApps"
- #define **OPTANE_SUPPORTED_APPS_DIR_LOCATION** L"%CommonProgramFiles%\\Intel\\Intel Optane Memory\\SupportedApps"

Typedefs

- typedef struct **OPTANE_PINNINGAPIVERSION OPTANE_PINNINGAPIVERSION**

Enumerations

- enum **OPTANERETCODE** { **OPTANE_SUCCESS**, **OPTANE_INTERNAL_ERROR**, **OPTANE_PARSE_ERROR**, **OPTANE_WRONG_CHAR**, **OPTANE_FILE_NOT_FOUND**, **OPTANE_DIRECTORY_NOT_FOUND**, **OPTANE_APP_NOT_FOUND**, **OPTANE_WRONG_ACCESS_RIGHTS**, **OPTANE_TMP_FILE_CREATION_ERROR**, **OPTANE_FILE_CACHE_FULL**, **OPTANE_FILE_CACHE_ALMOST_FULL**, **OPTANE_NO_SPACE_IN_CACHE**, **OPTANE_MORE_DATA**, **OPTANE_WRONG_LOCATION**, **OPTANE_INVALID_PARAMETER**, **OPTANE_NOT_READY**, **OPTANE_ACCESS_DENIED**, **OPTANE_INDIVIDUAL_FILE_ERROR**, **OPTANE_INVALID_VERSION**, **OPTANE_NOT_SUPPORTED** }
- enum **OPTANE_PINNING_CAPABILITY** { **OPTANE_PINNING_INACTIVE**, **OPTANE_PINNING_ACTIVE** }
- enum **OPTANE_PINNING_STATE** { **OPTANE_PINNED**, **OPTANE_NORMAL** }
- enum **OPTANE_CACHING_STATUS** { **OPTANE_UNCACHED**, **OPTANE_PARTIALLY_CACHED**, **OPTANE_CACHED** }

Detailed Description

Contains definition of externally available PinApi structures and enums

Enumeration Type Documentation

enum OPTANE_CACHING_STATUS

Enumerator:

OPTANE_UNCACHED	File is located on the slow drive.

OPTANE_PARTIALLY_CACHED	Part of the file is located on the fast drive and the remaining part is on the slow drive.
OPTANE_CACHED	File is located on the fast drive.

enum OPTANE_PINNING_CAPABILITY

Enumerator:

OPTANE_PINNING_INACTIVE	Pinning can be used.
OPTANE_PINNING_ACTIVE	Pinning cannot be used.

enum OPTANE_PINNING_STATE

Enumerator:

OPTANE_PINNED	File is pinned. File is in cache or will be migrated into cache.
OPTANE_NORMAL	File is not pinned. Optane algorithms and polices are applied to this file.

enum OPTANERETCODE

Enumerator:

OPTANE_SUCCESS	Operation completed successfully.
OPTANE_INTERNAL_ERROR	RST Optane component internal error.
OPTANE_PARSE_ERROR	Path/file name cannot be parsed correctly.
OPTANE_WRONG_CHAR	Path or file name contains a forbidden character.
OPTANE_FILE_NOT_FOUND	File not found.
OPTANE_DIRECTORY_NOT_FOUND	Directory not found.
OPTANE_APP_NOT_FOUND	Application not found.
OPTANE_WRONG_ACCESS_RIGHTS	Access was denied to file/directory/object.

OPTANE_TMP_FILE_CREATION_ERROR	Large temporary file for software vendor application cannot be created.
OPTANE_FILE_CACHE_FULL	File Cache is full of user pinned files.
OPTANE_FILE_CACHE_ALMOST_FULL	Predefined percentage of the File Cache is full of user pinned files.
OPTANE_NO_SPACE_IN_CACHE	Not enough space in the File Cache to satisfy pinning request.
OPTANE_MORE_DATA	If the allocated buffer is not large enough to hold the returned data.
OPTANE_WRONG_LOCATION	Path/file name point outside the fast and the slow drive that are forming the Optane logical volume
OPTANE_INVALID_PARAMETER	Pinning API function was called with wrong parameters.
OPTANE_NOT_READY	Command could not execute as Optane binaries are in a wrong state (for instance initializing, closing)
OPTANE_ACCESS_DENIED	Access to a resource is denied.
OPTANE_INDIVIDUAL_FILE_ERROR	Request executed properly but not all files were processed succesfully.
OPTANE_INVALID_VERSION	Incompatible versions of platform pinning API and DLL pinnining API.
OPTANE_NOT_SUPPORTED	Not supported architecture or platform.

SSE_Storage_NGSA/Implementation/source/OptaneApi/Headers/Optane.h File Reference

```
#include "OptaneTypes.h"
```

Macros

- #define **OPTANE_PIN_API_MAJ_VERSION** 1

- #define **OPTANE_PIN_API_MIN_VERSION** 2
- #define **PIN_API** __declspec(dllimport)

Typedefs

- typedef **OPTANERETCODE**(* **OptaneGetPinningStatusFunc**) (**OPTANE_PINNINGINFO** *status)
- typedef **OPTANERETCODE**(* **OptanePinFilesFunc**) (**OPTANE_FILELIST** list)
- typedef **OPTANERETCODE**(* **OptaneUnPinFilesFunc**) (**OPTANE_FILELIST** list)
- typedef **OPTANERETCODE**(* **OptaneGetPinStatusFilesFunc**) (**OPTANE_FILEPINSTATUSLIST** list)
- typedef **OPTANERETCODE**(* **OptaneTriggerMigrationFunc**) (void)
- typedef **OPTANERETCODE**(* **OptaneGetFileCacheStatusFunc**) (**OPTANE_FILECACHESTATUS** *fileCacheStatus)
- typedef **OPTANERETCODE**(* **OptaneGetPinningApiVersionFunc**) (**OPTANE_PINNINGAPIVERSION** *pinningApiVersion)

Functions

- PIN_API **OPTANERETCODE OptaneGetPinningStatus** (**OPTANE_PINNINGINFO** *status)
  Method provides the current state of Optane pinning.
- PIN_API **OPTANERETCODE OptanePinFiles** (**OPTANE_FILELIST** list)
  Method serves to soft pin files to the NGSA file cache.
- PIN_API **OPTANERETCODE OptaneUnpinFiles** (**OPTANE_FILELIST** list)
  Method serves to unpin files from the NGSA file cache.
- PIN_API **OPTANERETCODE OptaneGetPinStatusFiles** (**OPTANE_FILEPINSTATUSLIST** list)
  Method gets pin status of a single file or of a list of files.
- PIN_API **OPTANERETCODE OptaneTriggerMigration** (void)
  Method forces a migration procedure. It triggers the NGSA migration task.
- PIN_API **OPTANERETCODE OptaneGetFileCacheStatus** (**OPTANE_FILECACHESTATUS** *fileCacheStatus)
  Method gets the file cache status.
- PIN_API **OPTANERETCODE OptaneGetPinningApiVersion** (**OPTANE_PINNINGAPIVERSION** *pinningApiVersion)
  Method returns pinning API version.

Detailed Description

Contains definition of externally available PinApi methods

Function Documentation

PIN_API OPTANERETCODE OptaneGetFileCacheStatus (OPTANE_FILECACHESTATUS * *fileCacheStatus*)

Method gets the file cache status.

Parameters:

	**FileCacheStatus*	Pointer to an allocated area for the File Cache Status data.

Return values:

SUCCESS	When file cache status was returned successfully.
INVALID_PARAMETER	When provided **OPTANE_FILECACHESTATUS** *fileCacheStatus is a nullptr.
OPTANE_INTERNAL_ERROR	When the error is caused by one of the RST Optane component's internal issue.

PIN_API OPTANERETCODE OptaneGetPinningApiVersion (OPTANE_PINNINGAPIVERSION * *pinningApiVersion*)

Method returns pinning API version.

Parameters:

	**PinningApiVersion*	Pointer to allocated area for Pinning API version data.

Return values:

SUCCESS	When pinning api version was returned successfully.
INVALID_PARAMETER	When provided **OPTANE_PINNINGAPIVERSION** *pinningApiVersion is a nullptr.

PIN_API OPTANERETCODE OptaneGetPinningStatus (OPTANE_PINNINGINFO * *status*)

Method provides the current state of Optane pinning.

Parameters:

	**status*	Pointer to a Pinning Status information structure.

Return values:

SUCCESS	When status is returned successfully.
INVALID_PARAMETER	When the provided **OPTANE_PINNINGINFO** *status is a nullptr.
OPTANE_INTERNAL_ERROR	When the error is caused by one of the RST Optane component's internal issue.

PIN_API OPTANERETCODE OptaneGetPinStatusFiles (OPTANE_FILEPINSTATUSLIST *list*)

Method gets pin status of a single file or of a list of files.

Parameters:

[in|out]	list: Structure that contains path or paths to files with space for pin status of each file.

Return values:

SUCCESS	When status was returned successfully.
INVALID_PARAMETER	When the provided pointer OPTANE_FILEPINSTATUS* listOfFileStatus is a nullptr.
OPTANE_INDIVIDUAL_FILE_ERROR	When there are some file errors.
OPTANE_INTERNAL_ERROR	When the error is caused by one of the RST Optane components' internal issue.

PIN_API OPTANERETCODE OptanePinFiles (OPTANE_FILELIST *list*)

Method serves to soft pin files to the NGSA file cache.

Parameters:

[in|out]	list: Structure that contains path or paths to pinned files with error codes.

Return values:

SUCCESS	When the file was pinned successfully .
INVALID_PARAMETER	When the provided pointer to the list of paths is a nullptr.
OPTANE_INTERNAL_ERROR	When the error is caused by one of the RST Optane component's internal issue.

PIN_API OPTANERETCODE OptaneTriggerMigration (void)

Method forces a migration procedure. It triggers the NGSA migration task.

Return values:

SUCCESS	When migration trigger request was sent successfully.
OPTANE_NOT_READY	When the NGSA service is still busy.
OPTANE_INTERNAL_ERROR	When the error is caused by one of the RST Optane component's internal issue.

PIN_API OPTANERETCODE OptaneUnpinFiles (OPTANE_FILELIST *list*)

Method serves to unpin files from the NGSA file cache.

Parameters:

[in|out]	list: Structure that contains path or paths to pinned files with error codes.

Return values:

SUCCESS	When the file was unpinned successfully .
INVALID_PARAMETER	When the provided pointer to the list of paths is a nullptr.
OPTANE_INTERNAL_ERROR	When the error is caused by one of the RST Optane component's internal issue.

§ §