```
1    /* -*- Mode: C++; tab-width: 2; indent-tabs-mode: nil; c-basic-offset: 2 -*-
2     *
3     * The contents of this file are subject to the Netscape Public License
4     * Version 1.0 (the "NPL"); you may not use this file except in
5     * compliance with the NPL.  You may obtain a copy of the NPL at
6     * http://www.mozilla.org/NPL/
7     *
8     * Software distributed under the NPL is distributed on an "AS IS" basis,
9     * WITHOUT WARRANTY OF ANY KIND, either express or implied. See the NPL
10    * for the specific language governing rights and limitations under the
11    * NPL.
12    *
13    * The Initial Developer of this code under the NPL is Netscape
14    * Communications Corporation.  Portions created by Netscape are
15    * Copyright (C) 1998 Netscape Communications Corporation.  All Rights
16    * Reserved.
17    */
18
19   /*
20    * nsDiskModule. The disk cache module that stores the cache objects
21    * on the disk.
22    *
23    * Gagan Saksena 02/03/98
24    *
25    */
26   #ifndef _nsDiskModule_H_
27   #define _nsDiskModule_H_
28
29   #include "nsCacheModule.h"
30   #include "nsCachePref.h"
31   #include "mcom_db.h"
32
33   class nsDiskModule : public nsCacheModule
34   {
35
36   public:
37       nsDiskModule(const PRUint32 size = nsCachePref::GetInstance()->DiskCacheSize());
38       ~nsDiskModule();
39
40       // Cache module interface
41       PRBool          AddObject(nsCacheObject* io_pObject);
42
43       PRUint32        AverageSize(void) const;
44
45       PRBool          Contains(nsCacheObject* io_pObject) const;
46       PRBool          Contains(const char* i_url) const;
47
48       void            GarbageCollect(void);
49
50       nsCacheObject*  GetObject(const PRUint32 i_index) const;
51       nsCacheObject*  GetObject(const char* i_url) const;
52
```

```
55      PRBool              Remove(const char* i_url);
56      PRBool              Remove(nsCacheObject* pObject);
57      PRBool              Remove(const PRUint32 i_index);
58
59      // To do cleanup set size to zero. Else initialize disk cache
60      void                SetSize(const PRUint32 i_size);
61
62      PRBool              Revalidate(void);
63
64  protected:
65
66      PRBool              ReduceSizeTo(const PRUint32 i_NewSize);
67
68  private:
69      enum sync_frequency
70      {
71          EVERYTIME,
72          IDLE,
73          NEVER
74      } m_Sync;
75
76      PRBool          InitDB(void);
77
78      nsDiskModule(const nsDiskModule& dm);
79      nsDiskModule& operator=(const nsDiskModule& dm);
80
81      DB* m_pDB;
82  };
83
84  inline
85  PRUint32 nsDiskModule::AverageSize(void) const
86  {
87      MonitorLocker ml((nsDiskModule*)this);
88      if (Entries()>0)
89      {
90          return (PRUint32)(m_SizeInUse/m_Entries);
91      }
92      return 0;
93  }
94  #endif
95
```