ProcessInfoRec   structure

#include <Processes.h>

```
typedef struct ProcessInfoRec {                 Size    Offset  Description
    unsigned long           processInfoLength;  4       0       length of record
    StringPtr               processName;        4       4       name of process
    ProcessSerialNumber     processNumber;      8       8       psn of the process
    unsigned long           processType;        4       16      file type of app file
    OSType                  processSignature;   4       20      signature of app file
    unsigned long           processMode;        4       24      'SIZE' resource flags
    Ptr                     processLocation;    4       28      address of partition
    unsigned long           processSize;        4       32      partition size
    unsigned long           processFreeMem;     4       36      free bytes in heap
    ProcessSerialNumber     processLauncher;    8       40      process that launched
    unsigned long           processLaunchDate;  4       48      time when launched
    unsigned long           processActiveTime;  4       52      accumulated CPU time
    FSSpecPtr               processAppSpec;     4       56      location of the file

} ProcessInfoRec;                               60
```

Notes: You specify the values for three fields of the process information record:
processInfoLength , processName and processAppSpec . You must either set
the processName and processAppSpec fields to NIL or set these fields to
point to storage that you have allocated for them. The
GetProcessInformation function returns information in all other fields
of the process information record.

The processInfoLength field is the number of bytes in the process
information record. For compatibility, you should specify the length of the
record in this field. The name returned in the processName field is the name
of the application or desk accessory. For applications, this field contains the
name of the application as designated by the user at the time the application
was opened. For example, for foreground applications, the processName field
is the name as it appears in the Application menu. For desk accessories, the
processName field contains the name of the 'DRVR' resource. You must
specify NIL in the processName field if you do not want the application name
or the desk accessory name returned. Otherwise, you should allocate at least
32 bytes of storage for the string pointed to by the processName field. Note
that the processName field specifies the name of either the application or
the 'DRVR' resource, whereas the processAppSpec field specifies the
location of the file.

The processNumber field specifies the process serial number. The process
serial number is a 64-bit number; the meaning of these bits is internal to
the **Process Manager**. You should not interpret the value of the process
serial number.

The processType field indicates the file type of the application, generally
'APPL' for applications, and 'appe' for background-only applications
launched at startup. If the process is a desk accessory, the processType field
is the type of the file containing the 'DRVR' resource.

The processSignature field indicates the signature of the file containing the
application or the 'DRVR' resource (for example, the signature of the
TeachText application is 'ttxt').

The processMode field indicates whether the process is an application or desk accessory. For applications, this field also returns information specified in the application's 'SIZE' resource. This information is returned as flags.You can refer to these flags by using these constants.

modeDeskAccessory

modeMultiLaunch

modeNeedSuspendResume

modeCanBackground

modeDoesActivateOnFGSwitch

modeOnlyBackground

modeGetFrontClicks

modeGetAppDiedMsg

mode32BitCompatible

modeHighLevelEventAware

modeLocalAndRemoteHLEvents

modeStationeryAware

modeUseTextEditServices

The processLocation field is the beginning address of the application partition.
The processSize field is the number of bytes in the application's partition (including the heap, stack, and A5 world).

The processFreeMem field is the number of free bytes in the application's heap. The processLauncher field is the process serial number of the process that launched the application or desk accessory. If the original launcher of the process is no longer open, this field contains the constant kNoProcess.

The processLaunchDate field contains the value of the Ticks global variable at the time that the process was launched.

The processActiveTime field represents the accumulated time, in ticks, during which the process has used the CPU, including both foreground and background processing time.

The processAppSpec field specifies the address of a file specification record that stores the location of the file containing the application or 'DRVR' resource. You should specify NIL in the processAppSpec field if you do not want the FSSpec record of the file returned.