PGetRequest Receive a request sent by a **PSendRequest** call

#include <AppleTalk.h> **AppleTalk Manager**

```
OSErr      PGetRequest(thePBptr, async);
ATPPBPtr   thePBptr;    pointer to an ATPParamBlock structure
Boolean    async;       0=await completion; 1=immediate return
           returns      Error Code; 0=no error
```

PGetRequest sets up the mechanism to receive a request sent by a **PSendRequest** call.

thePBptr iis a pointer to an ATPParamBlock structure.

Out-In	Name	Type	Size	Offset	Description
←	userData	long	4	18	user bytes
→	csCode	short	2	26	always getRequest
→	atpSocket	char	1	28	socket number
←	atpFlags	char	1	29	control information
←	addrBlock	AddrBlock	4	30	destination socket address
↔	reqLength	short	2	34	request size in bytes
→	reqPointer	Ptr	4	36	pointer to request data
←	bitMap	Ptr	4	40	bitmap
←	transID	short	2	48	transaction ID

async is a Boolean value. Use FALSE for normal (synchronous) operation or TRUE to enqueue the request and resume control immediately. See Async I/O.

Returns: an operating system Error Code. It will be one of:

noErr	(0)	No error
badATPSkt	(-1099)	Bad responding socket

Notes: userData returns the four user bytes from the request. atpSocket contains the socket number of the socket that should listen for a request. The internet address of the socket from which the request was sent is returned in addrBlock. reqLength and reqPointer indicate the size (in bytes) and location of a buffer to store the incoming request. The actual size of the request is returned in reqLength. The transaction bitmap and transaction ID will be returned in bitMap and transID. The exactly-once flag in atpFlags will be set if the request is part of an exactly-once transaction.

To send a request to another socket and get a response, call **PSendRequest**. The call terminates when either an entire response is received or a specified retry timeout interval elapses. To open a socket for the purpose of responding to requests, call **POpenATPSkt**. Then call **PGetRequest** to receive a request; when a request is received, the call is completed. After receiving and servicing a request, call **PSendResponse** to return response information. If you cannot or do not want to send the entire response all at once, make a **PSendResponse** call to send some of the response, and then call **PAddResponse** later to send the remainder of the response. To close a socket opened for the purpose of sending responses, call **PCloseATPSkt**.