

## MicroKernel Database Engine Status Codes

This section describes status codes that the MicroKernel returns.

### 1: The operation parameter is invalid

The specified operation does not exist or is not valid.

### 2: The application encountered an I/O error

This status code typically indicates a corrupt file, an error while reading from or writing to the disk. One of the following has occurred:

The file is damaged, and you must recover it. Refer to the Pervasive.SQL Advanced Operations Guide for more information on recovering files.

In NetWare, at least one MicroKernel data file is flagged as Shareable. All MicroKernel data files should be flagged as Non-Shareable. The MicroKernel is the only user that accesses the files (on behalf of many users). As the only user, the MicroKernel can control the integrity of the data files. If you flag your MicroKernel data files as Shareable, data corruption can occur.

For pre-v6.0 data files, there is a large pre-image file inside a transaction, and there is not enough disk space for a write to the pre-image file.

For pre-v6.0 data files, there is one pre-image file for multiple data files. For example, if you name the data files CUSTOMER.ONE and CUSTOMER.TWO, both files have pre-image files named CUSTOMER.PRE.

For pre-v6.0 data files that are larger than 768 MB, there is a conflict among locking mechanisms. The file has not been corrupted. Your application can retry the operation until the conflict is resolved (when the competing application releases the lock your application requires).

A pre-v6.0 Btrieve engine attempted to open a v6.x or later MicroKernel file. With Btrieve for Windows NT Server Edition v6.15.445, 32 bit Windows application may return Status 2 or "MKDE Terminated with Service Specific Error 0" after running an application for an extended period of time.

There was a conflict with the assignment of token handles and permissions. You may receive status 2 or corruption on very busy SMP boxes, when a user is deleted from the Btrieve Monitor and the user immediately reopens the files. There is an FTF for Pervasive.SQL v.7.0 Windows NT and Service Pack 3 for Pervasive.SQL 2000i Windows NT.

### 3: The file is not open

The operation cannot execute because the file is not open. The application must perform a successful Open operation before the MicroKernel can process any other operations. The MicroKernel also returns this status code if the application passed an invalid position block for the file, or if the application passed a position block with a client ID other than the client ID used to open the file.

### 4: The application cannot find the key value

The MicroKernel cannot find the specified key value in the index path. When you receive this status code on an Update or Delete operation, it usually means that the file is damaged, and you must recreate it. Occasionally, a corrupt key can cause this status code. Drop the key, then add it again.

The MicroKernel returns this status code when an application performs a Get Equal operation to search on field type char. It is caused by a mismatch of two fields at the char level. To resolve, fill the KeyBuffer with the same fill char type as the field.

**5: The record has a key field containing a duplicate key value**

The MicroKernel cannot add or update a record because the record has a key field that contains a duplicate key value for an index that does not allow duplicate values. The MicroKernel also returns this status code when it cannot create an index that does not allow duplicate key values because a duplicate key value already exists.

**6: The key number parameter is invalid**

The value stored in the key number parameter is not valid for the file being accessed. The key number must correspond to one of the keys defined for the file. Valid key numbers are 0 through 118.

**7: The key number has changed**

The key number parameter changed before a Get Next, Get Next Extended, Get Previous, or Get Previous Extended operation. The operation requires the same key number parameter as the previous operation, because the MicroKernel uses positioning information relative to the previous key number.

In a related situation, the MicroKernel returns this status code when an application performs a Delete or Update operation immediately following a Get operation. If the application changes the value of the key number in the Delete or Update operation (from the value used with the preceding Get operation), the MicroKernel deletes or updates the record as requested and does not return this status code, at least not at this point. However, the MicroKernel does return this status code on the very first Get Next, Get Next Extended, Get Previous, or Get Previous Extended operation performed after the deletion or update, even if that Get operation uses the same key value the application passed to the Delete or Update operation.

If you need to change key numbers between consecutive Get Next, Get Next Extended, Get Previous, or Get Previous Extended operations (or in Delete or Update operations as described in the preceding paragraph), use a Get Position operation followed by a Get Direct/Record operation to reestablish positioning for the new index path.

**8: The current positioning is invalid**

You must establish the current position in order to update or delete a record. Perform a Get or Step operation to establish the current position. The MicroKernel also returns this status code if the application passed an invalid position block for the file.

**9: The operation encountered the end-of-file**

The MicroKernel returns this status code in the following situations:

The operation encountered an end-of-file boundary or tried to read past a file boundary (end-of-file or start-of-file).

In a Get Next Extended, Get Previous Extended, Step Next Extended, or Step Previous Extended operation, the number of records satisfying the filtering condition is less than the number of specified records to be returned, and the reject count or filter limit has not been reached.

When reading a file in ascending order according to an index path, the MicroKernel has already returned the last record in that index path. When reading a file in descending order according to an index path, the MicroKernel has already returned the first record in the index path.

When using the Get By Percentage operation, either the value supplied for the percentage is too high-it exceeds 10,000 decimal (0x2710)-or the file contains no records.

When using the Get operation using ActiveX Data Control, this error will occur only after the application is compiled and deployed. This error will not occur at design time during development. The error results from a missing MSDADC.DLL on the deployment machine. Make sure to include this file (MSDADC.DLL) in your installation script. This file is a Microsoft MDAC (Microsoft Data Access Component) file.

#### **10: The key field is not modifiable**

During an Update operation, the application attempted to modify a key field that is cannot be modified by definition.

#### **11: The specified filename is invalid**

The MicroKernel returns this status code in the following situations:

The specified filename does not conform to file naming conventions or the pathname is invalid. Make sure the filename or pathname is valid for the environment.

If operating in the client/server environment:

The application attempted to open a file that has .^^^ as its extension. This extension is reserved for the MicroKernel to use during continuous operation. (Only server engines can use continuous operation.)

The data buffer for a Begin or End continuous operation is not set up correctly.

You attempted to load a remote file when your Communications Requester settings of Local and Requester are incorrectly set to On and Off, respectively. To resolve this condition, go to the Access option in the Client configuration settings and change the Requester setting to On. For more information about Configuration settings, refer to the Pervasive.SQL Advanced Operations Guide.

You attempted to open a local file with a Workgroup engine. Go to the directory where the file you attempted to open resides. Check to see if the ~pvsw~.loc in that directory is flagged read only. If it is, change it to read-write.

If you are using the dynamic locator file in your Workgroup engine:

Make sure the name of the second-level locator file specified in your first-level locator file does not have the same name as an existing directory. Also, make sure they are all on the same drive.

Make sure the second-level locator file specified in your first-level locator file can be accessed by the engine.

Make sure you have the same directory level of drive mapping across all the Workgroup engine sharing dynamic locator feature.

If you are accessing files on a DOS client:

A NET START FULL for the Windows for Workgroups workstation was not used when it was booted into DOS. Use a NET START FULL to get a full redirector in the DOS client mode. BREQNT.EXE requires a full redirection. At the DOS prompt type "net ver" <Enter>. Here is the list of required components for a DOS workstation to connect to a Windows NT server:

LSL.COM LAN Card Driver

IPXODI.COM

IFSHLP.SYS

NET START FULL

These can be loaded high, using emm386. NET START FULL will load in upper memory. NETX is required ONLY if you are connecting to a NetWare server.

You attempted to open a file with a long filename on NSS volumes. The MicroKernel queries the volumes using OS calls and then loads the appropriate drivers for the associated name spaces it find for the volumes. In this case, the MicroKernel was being loaded before mounting the volumes so it did not find the requirement for long filename support by the NSS volumes.

Issue the MGRstart or Bstart command after loading the volumes. An example would be:

```
LOAD NSS
MOUNT ALL
SYS:ETC\INITSYS.NCF
MGRSTART or BSTART
```

If you are working in the Microsoft Terminal Server environment:

Approximately 5 users can work in a Windows application on 2 different Terminal Servers that are connected to a primary Windows NT 4.0 server. If you have attempted to run on top of this limit, you will receive status codes 11 and 35. For the recovery solution for this instance, refer to Microsoft Knowledgebase Article Q190162, "Terminal Server and the 2048 Open File Limitation."

## **12: The MicroKernel cannot find the specified file**

Check that the file exists and that you specified the correct file.

Check the key buffer parameter to make sure the pathname is terminated with a blank or a binary zero.

When accessing a file on a server, ensure that you have FILE SCAN rights to the directory in which the file resides. Occasionally, a corrupt key can cause this status code. Drop the key, then add it again.

In Pervasive.SQL 2000i NetWare and Novell NetWare 4.x environments, check for files with the hidden file attribute at the Operating System level and remove the hidden file attribute.

This does not occur with NetWare 5.

An ODBC connection may receive this code on NetWare 4.x with the Pervasive.SQL 2000 Workgroup Engine. This status code did not appear until Pervasive.SQL 2000 Service Pack 2a. It is caused by Netware 4.x reporting long directory in a different format than NetWare 3.x, NetWare 5.x, or the Win32 Environment.

Novell has a fix for this problem in a file named Mixmod6.exe. Search for file at Novell's web site (<http://www.novell.com>).

This error code may also be returned when the file DBNames.CFG has been removed (for example, by the Pervasive.SQL cleanup utility) and old Data Source Names (DSNs) which reference that file are not removed from the ODBC. You may have a Status 12 returned and see X\$FILE.DDF for the file name in your DDFs .

If the file.ddf is examined with a Btrieve utility, the location name for the dictionary files will be x\$file.ddf, x\$field.ddf, x\$index.ddf, instead of the standard file.ddf, field.ddf, index.ddf. An unsupported DDF Creation utility called DDL Services (DDL SVCS.DLL) created the DDFs. DDL Services has a known bug that causes the system table to be populated with incorrect data. To resolve this error, recreate the DDFs using a newer tool (DDF Ease or SQL Scope for Pervasive.SQL v7.0 or SQL Data Manager/Pervasive Control Center with Pervasive.SQL 2000).

You may get Status 12 when a file with a filename or path with embedded spaces is

opened on Windows 9x, Windows NT or Windows 2000. For long NetWare file names, support is available only in the MicroKernel and not in NLM utilities such as BUTIL.NLM. Btrieve data files can be accessed using long names but long names cannot be used for other files. For example, in the NLM command:  
BUTIL -CREATE <outputFile> <descriptionFile>

Since MKDE manipulates the <outputFile> Btrieve data file, it can have a long name. However, the <descriptionFile> can only have a short name (such as BUTIL.NLM) since it does not accept long names. The solution is to enable "Embedded Spaces".

To enable Embedded Spaces in Pervasive.SQL v7 Service Pack 3 +:

Choose Start | Run.

Type Regedit and click OK. The Registry Editor opens.

Browse to HKL\_Machine | Software | Pervasive Software | Btrieve Requester | Version 7 | Settings:Embedded Spaces

Select Yes.

Exit the configuration utility and restart the engines for the new settings to take effect.

To Enable Embedded Spaces in Pervasive.SQL 2000i:

Click Start | Programs | Pervasive | Pervasive Control Center.

Expand Pervasive.SQL 2000i Engines and expand the desired database.

Expand Configuration, then expand Client, and click on Application Characteristics.

In the right hand frame, double-click on Embedded Spaces.

Select On.

Exit the configuration utility and restart the engines for the new settings to take effect.

### **13: The MicroKernel could not open the extension file for an extended file**

The MicroKernel could not open the extension file for an extended file that the application tried to open. An extended file can consist of a base file and up to 31 extension files. Extension files must remain in the same volume and directory as their base files. The MicroKernel returns this status code if you delete, move, or rename the extension files.

### **14: The MicroKernel cannot create or open the pre-image file**

The MicroKernel uses pre-image files only for pre-v6.0 data files.

The MicroKernel returns this status code in one of the following situations:

The MicroKernel cannot create a new pre-image file because the disk directory is full. The MicroKernel must be able to create a pre-image file.

The MicroKernel cannot open the pre-image file to restore file integrity. If the pre-image file is erased or damaged, the MicroKernel cannot restore the file integrity. Refer to the Pervasive.SQL Advanced Operations Guide for more information about recovering damaged files.

The workstation MicroKernel cannot assign a handle to the pre-image file because the MicroKernel was not started by a user with access rights to the pre-image file. The file structure of a pre-image file created by this MicroKernel is different from the file structure of a pre-image file created by a v5.x MicroKernel. If you have an

extraneous .PRE file in v5.x format, the MicroKernel returns this status code when you try to open the data file to which the .PRE file belongs.

### **15: The application encountered an I/O error during pre-imaging**

The MicroKernel uses pre-image files only for pre-v6.0 data files.

The pre-image file is damaged and the integrity of the data file cannot be ensured. Refer to the Pervasive.SQL Advanced Operations Guide for more information about recovering damaged files.

The disk is full. Erase any unnecessary files.

### **16: The application encountered an expansion error**

This status code is obsolete in MicroKernel versions 6.0 and later.

The MicroKernel encountered an error while writing the directory structure to disk prior to creating the expanded file partition. Either the MicroKernel cannot close the file, or a new page was added to the file and the MicroKernel cannot close and reopen the file to update the directory structure. Check for a disk hardware failure.

### **18: The disk is full**

The MicroKernel can return this status code in the following situations:

The disk is full and the MicroKernel cannot expand the file to accommodate additional records. Erase any unnecessary files.

There is not enough space to append a new page to the data file.

The pre-image file is out of disk space. If your files are in pre-v6.0 format and you are in a transaction, the pre-image file size increases for the duration of the transaction. If you receive this status code, either reduce the number of operations in the transaction, or obtain more disk space.

For files located on a NetWare server, the NetWare owner name for the file is no longer valid, and your application tried to insert or update records in the file, thus causing the file to expand. In this case, the MicroKernel returns this status code when it needs to add a page to the file, regardless of how much disk space is available. To check for an owner name, use the NetWare utility NDIR. To add an owner name, use either FILER (a NetWare text utility) or the NetWare Administrator graphical utility.

In some environments, you can restrict the amount of disk space available to each user. This status code indicates that the application attempted to expand a data file beyond the amount of disk space allocated to the file owner.

You tried to read or modify a file which was not closed properly after a disk full error. Make sure that every application using the file at the time of the disk full error closed the file successfully.

If a client connected to a Pervasive.SQL server encounters this status code, other clients performing read-only operations from the same disk may also receive a non-zero status.

### **19: The application encountered an unrecoverable error**

To ensure file integrity, recover the file as described in the Pervasive.SQL User's Guide.

## **20: The MicroKernel or Btrieve Requester is inactive**

If you are running an application in a client/server environment:

Make sure the Btrieve requester is loaded.

Verify that the IPX/SPX or TCP/IP protocol is properly installed at the client machine and that no two machines on the network have the same Internal Network Number.

Make sure the Local Engine Usage option is enabled by default in Windows environments. If you do not have a local MicroKernel available or do not wish to access local data, disable this option.

If you are running an application in a workstation/workgroup environment, make sure the MicroKernel is loaded.

If you are running an application in a client/server environment and also need to access files located on a local drive:

Make sure the Btrieve Requester is loaded.

Make sure you enable both Local Engine Usage and Server Engine Usage.

Make sure a local MicroKernel is available and loaded.

If you are operating in a DOS server environment:

Pervasive.SQL 2000i installs BTRBOX95 by default. As long as this is installed no other requester can be used.

If you want to use BREQNT, BREQUEST, or BREQTCP, you must remove BTRBOX95 and make sure the proper requesters are loaded. Refer to Getting Started with Pervasive.SQL for more information.

If you want to use BTRBOX95:

Change directories to \pvsw\clients\dos\windosbox and run the setup utility. This loads the appropriate file for clients running Windows 9x or Windows NT. The setup program creates and places a README.TXT file in the \pvsw\W32DOXBOX directory. After installation, reboot the client.

For Windows NT users: open a command prompt and run a DOS Btrieve application.

For Windows 9x users:

Run the BTRBOX95.exe. A minimized dialog box appears, indicating that BTRBOX95 support is active. If you close this dialog box, it will unload BTRBOX95. You only need to run BTRBOX95.exe once even though you have multiple DOS sessions open. Open a command prompt and run your DOS Btrieve application.

If you are operating in a NetWare environment:

Make sure the Microkernel and (if applicable) the Btrieve Requester are loaded before generating any requests.

Make sure the appropriate communication modules are loaded at the server.

If you are operating in a Windows NT server environment:

Make sure the MicroKernel is started before generating any requests.

Make sure the Windows NT DLLs are in your path.

Make sure the appropriate communications modules are loaded at the server.

## **21: The key buffer parameter is too short**

The key buffer parameter is not long enough to accommodate the key field for the requested index path. Verify that the length of the key buffer equals the defined length of the key specified in the key number parameter.

## **22: The data buffer parameter is too short**

The data buffer parameter specified by the application was not large enough to accommodate either the minimum length of the record for an Insert or Update operation, or the entire record length for a Get or Step operation. Also, the data

buffer may not be large enough to accommodate the length of data required for operations such as Create, Create Index, Stat, Get By Percentage, Find Percentage, or Version.

For Get or Step operations, the MicroKernel returns as much data as it can and this status code, indicating that it cannot return the entire record.

For an Insert operation, the MicroKernel does not insert the record if the data buffer is shorter than the fixed-length portion of the record.

For an Update operation, if the data buffer is too short to contain the fixed-length portion of a record, the MicroKernel does not update the record.

For the Create, Stat, and Create Index operations, the data buffer is not long enough to contain all the file specifications, the key specifications, and (if specified) the Alternate Collating Sequence (ACS) definition.

For the Get by Percentage or Find Percentage operation, the data buffer length is less than 4 bytes.

For the Version operation, the data buffer length is less than 5 bytes.

The data buffer parameter is too short when access table with more than 60 field using ActiveX. Download the latest ActiveX controls from the Pervasive web site ([http://www.pervasive.com/support/updates/upd\\_sdk.tml#activex](http://www.pervasive.com/support/updates/upd_sdk.tml#activex))

A corrupt file may be indicated if the file allows variable-length records and you receive this status code on a Get or Step operation. In such a corrupt file, you can receive Status Code 54 when you use Get or Step operations to read other records. Recover the file according to the instructions in the Pervasive.SQL User's Guide.

### **23: The position block parameter is not 128 bytes long**

This status code is obsolete in Btrieve language interface versions 6.0 and later.

The position block parameter must be exactly 128 bytes long.

### **24: The page size or data buffer size is invalid**

The MicroKernel returns this status code in one of the following situations:

The page size you specified when creating a file is invalid. The page size must be a multiple of 512 bytes and cannot exceed 4096 bytes.

During a Create operation, the page size is the first file specification the MicroKernel checks. If you receive this status code at this point, it can indicate an invalid data buffer parameter.

### **25: The application cannot create the specified file**

The MicroKernel returns this status code in one of the following situations:

If an application attempted to create a data file, the disk directory or the disk itself may be full.

If an application tried to create a file over an existing file, the existing file is open or the operating system will not allow the operation for another reason (such as when a file is flagged transactional in NetWare).

In an attempt to create a Btrieve file over existing Btrieve file, this status will be returned. The keybuffer on the Btrieve create operation API (opcode 14) is set properly to create a file over an existing file.

This problem may be caused by Antivirus software (such as Innoculan)

One solution is to disable the Antivirus software. Contact the third party vendor for additional information on configuring the Antivirus software to eliminate scanning specific data files.

**26: The number of keys specified is invalid**

The number of keys specified for the page size is invalid. The number of key segments must be within the following limits:

Table 1-6 Maximum Number of Key Segments

Page Size 512 1024 1536 2048 2560 3072 3584 4096

Maximum Number Key Segments 8 23 24 54 54 54 54 119

If the MicroKernel is configured to create files in v5.x format, the maximum number of key segments is 8 for files using 512 byte page sizes; the maximum number of key segments for all other v5.x files is 24.

For a key-only file, this status code is also returned if the number of keys specified is not one, or the available linked keys is not zero (only one key is allowed for a key-only file.)

Pervasive.SQL 2000i nullable columns: Because each indexed nullable column with true null support requires an index consisting of 2 segments, you cannot have more than 59 indexed nullable columns in a table (or indexed nullable true null fields in a Btrieve file). This limit is smaller for smaller page sizes (1/2 the values noted above, rounded down). Any file created with Pervasive.SQL 2000i, with file create mode set to 7.x, and TRUENULLCREATE set to the default value of On, has true null support. Files created using an earlier file format, or with Pervasive.SQL 7, or with TRUENULLCREATE set to Off, do not have true null support and do not have this limitation.

**27: The key position is invalid**

The specified key field position is less than 1 or exceeds the defined record length for the file. Either the key position is greater than the record length or the key position plus the key length exceeds the record length.

**28: The record length is invalid**

The specified record length (plus overhead for duplicates, record usage count, variable record pointers, key pointers, and blank truncation information) must be less than or equal to the page size minus 6 bytes, and greater than or equal to 6 bytes.

For key-only files, the maximum record length is 253 bytes (255 bytes for a pre-v6.0 file).

You have tried to create a file with a record length greater than 4086 bytes. The Btrieve API Create(14) Call will returned this status code. Due to keeping the fixed length portion of a record on one page in memory, the fixed length portion of any record may not exceed 4078 bytes (4086 - 8), to encompass the record and overhead for the record.

A program may define variable length portions of a record. These variable length portions are not restricted to remain on any one page in memory.

Thus, if a record length is needing to be 5000 bytes:

one can create two separate files and join them with JOINS in the queries  
OR

one can create a fixed length portion of the record, not to exceed 4078 bytes, and place the additional columns/fields in a variable length portion.  
OR

one can select compression when creating the file. This will allow one to have a large size because the compression selected files reside on variable pages as opposed to fixed length pages.

### **29: The key length is invalid**

The MicroKernel returns this status code in one of the following situations:

You have not specified a key length that is greater than 0 but does not exceed 255 bytes. The length of a key segment must agree with its key type if the key type implies a length (for example, an integer key must have a length evenly divisible by two). Each key page in the file must be large enough to hold at least eight keys. If the page size is too small to accommodate eight occurrences of the specified key length (plus overhead), you must increase the file page size or decrease the key length.

An additional byte of storage space is needed for the null indicator for the column. This error occurs through a SQL CREATE INDEX statement, or through the creation of a SQL PRIMARY KEY or FOREIGN KEY, if the index, or key, references a null CHAR column of 255 characters (or VARCHAR of 254). This additional byte causes the actual length of the index to be one byte longer, or 256 bytes. To resolve the error, reduce the size of the column or create the column as NOT NULL and try again. For a foreign key, if you decrease the size of the column, you must decrease both the referencing column and the referenced column.

You have tried to create a Btrieve file with more than one index. This may occur in the Btrieve v6.15 Windows95/NT environment. The problem is reproducible developing 32-bit application using Microsoft Visual Basic and is caused by long variables aligned on even word boundaries. The solution is to not use the long variable in the KeySpec type. Use a four (4) byte string instead of a long variable in the type definition. Then convert the string into a long.

### **30: The file specified is not a MicroKernel file**

This status code is returned in one of the following situations:

The MicroKernel did not create the file, or a pre-v3.x MicroKernel created it. While using an earlier version of Btrieve, you opened a file created by a later version that has a format incompatible with the earlier version.

The first page of the file may be damaged. Use a backup copy of your data file. If you receive this status code and you suspect that the header page of the source file is damaged, recover the file as described in the Pervasive.SQL User's Guide.

You have attempted to access a valid Btrieve file. This status code is returned when old engines access newer file formats. A likely scenario is that data created by a new server engine is later used by an earlier workstation engine. Status 30 can be reported if the file format is newer than the MicroKernel engine attempting to open it. Particularly, accessing a 7.x file with a 6.x engine causes this error.

NOTE: Previously, accessing a 6.x file with a 5.x engine returned Status 2: "the application encountered an I/O error"

**31: The file is already extended**

This status code is obsolete in MicroKernel versions 6.0 and later.

An application tried to extend a file that had already been extended; you can only extend a file once.

**32: The file cannot be extended**

The MicroKernel must create an extension file to accommodate a file which is growing larger than the operating system file size limit. However the MicroKernel encounters an error from the operating system when it tries to create and open the new extension file. Possible causes for receiving this status code include the following: the directory is full, the disk is full, or the MicroKernel has not been granted sufficient rights by the operating system.

**33: The MicroKernel cannot unload**

In the DOS environment, The MicroKernel returns this status code for the following reasons:

You attempted to unload the MicroKernel when you have loaded another terminate and stay resident (TSR) program after you loaded the MicroKernel. Unload the other TSR before unloading the MicroKernel.

You attempted to unload the MicroKernel from a 32-bit application that uses the BSTUB interface with the DOS/4G extender.

**34: The specified extension name is invalid**

This status code is obsolete in MicroKernel versions 6.0 and later.

An application specified an invalid filename for the extended partition. Check the validity of the filename.

**35: The application encountered a directory error**

Either a Get Directory operation specified a drive that does not exist, or a Set Directory operation specified an invalid pathname. Check the validity of both the drive and the pathname.

**37: Another transaction is active**

The application issued a Begin Transaction (19 or 1019) operation while another transaction was active by the same user or task; the active transaction can be nested or non-nested. This status code often indicates a problem in nested transactions within your application.

**38: The MicroKernel encountered a transaction control file I/O error**

This status code is obsolete in MicroKernel versions 7.0 and later.

The MicroKernel tried to write to the transaction control file. Possible causes for receiving this status code are that the disk is full, the disk is write protected, the transaction control file (BTRIEVE.TRN) that is created when you load the MicroKernel has been deleted, or the transaction control file is flagged read-only or is corrupt.

**39: A Begin Transaction operation must precede an End/Abort Transaction operation**

The application issued an End Transaction (20), or Abort Transaction (21) operation without a corresponding Begin Transaction (19 or 1019) operation. Make sure that each End or Abort Transaction operation in your program is executed only after a successful Begin Transaction operation.

**40: The file access request exceeds the maximum number of files allowed**

This status code is obsolete in MicroKernel versions 6.0 and later.

The application tried to access more than the maximum number of files allowed within a transaction. You set the maximum number of different files that you can access during a logical transaction when you configure the MicroKernel.

**41: The MicroKernel does not allow the attempted operation**

The MicroKernel returns this status code for the following reasons:

The application tried to perform an operation that is not allowed under these operating conditions. The MicroKernel does not allow some operations under certain operating conditions. For example, the MicroKernel returns this status code if the application attempts to perform a Step operation on a key-only file.

If using a server engine, the key number parameter of a continuous operation MicroKernel call is not valid.

The MicroKernel prohibits certain operations during transactions because they have too great an effect on the file or on performance. These operations include Set Owner, Clear Owner, Create Index, and Drop Index.

**42: A file previously opened in Accelerated mode was not closed**

This status code is obsolete in MicroKernel versions 6.0 and later.

The MicroKernel returns this status code for the following reasons:

Either the application tried to open a v5.x data file that was previously accessed in Accelerated mode by a v5.x MicroKernel and never successfully closed, or the application tried to open a file for which a v6.0 or later MicroKernel encountered an unrecoverable error during a Set or Clear Owner operation. The file integrity cannot be ensured. Refer to the Pervasive.SQL Advanced Operations Guide for information about recovering damaged files.

Your application tried to open a file in MicroKernel v5.x format using a v5.x MicroKernel; however, that same file was previously accessed by a v6.0 or later MicroKernel, which failed to close the file successfully and left a pre-image file on the disk. Version 5.x MicroKernels cannot read pre-image files created in v6.0 or later format.

**43: The specified record address is invalid**

The MicroKernel returns this status code for the following reasons:

The record address specified for a Get Direct operation is invalid. Either the address is outside the file boundaries, or it is not on a record boundary within or on a data page, or the record at the specified address has been deleted. For a Get Direct operation, specify the 4-byte address obtained by a Get Position operation.

If the records' file is in v5.x format, this status code can indicate a file access

conflict. For example, task 1 has a file locked in an exclusive transaction. Task 2 is reading records from the same file and tries to update a record that the transaction inserted. If task 2 reads the record and then task 1 aborts the transaction, task 2 receives this status code when issuing the Update operation.

For a Find Percentage operation that is seeking a percentage based on a record's physical location within the file, the specified record address is invalid.

The file may be corrupt, and you must recover it. Refer to the Pervasive.SQL Advanced Operations Guide for information about recovering damaged files.

#### **44: The specified key path is invalid**

The application tried to use the Get Direct/Record operation to establish positioning on an index path for a key whose value is null in the corresponding record. The MicroKernel cannot establish positioning based on a null key value.

#### **45: The specified key flags are invalid**

The key flags specification on a Create operation is inconsistent. If a key has multiple segments, the duplicate, modifiable, and null attributes should be the same for each segment in the key. Also, you cannot use the null or manual key attributes in a key-only file. The MicroKernel also returns this status code if the application attempted to specify a different Alternate Collating Sequence (ACS) for two or more segments of a segmented key.

#### **46: Access to the requested file is denied**

The MicroKernel returns this status code for the following reasons:

The application opened a file in Read-only mode and tried to perform a Write operation on that file.

The application attempted to perform a Write operation on a file that is flagged read-only by the operating system.

When the application opened the file, it did not correctly specify the owner name required for updates.

(Workstations Only) If a workstation engine user or task opens a file that a client machine has opened using a server MicroKernel, the MicroKernel returns this status code if the workstation engine user attempts to write to the file.

#### **47: The number of files opened exceeds the maximum allowed**

This status code is obsolete in MicroKernel versions 6.0 and later.

Pre-v6.0 workstation MicroKernels return this status code when the number of files opened in Accelerated mode exceeded the number of buffers available in the MicroKernel cache. When a file is opened in Accelerated mode, the MicroKernel reserves one of its cache buffers for the file. It always reserves five empty buffers for index manipulation. Reconfigure Btrieve with both a smaller /P configuration option (to allocate more buffers) and a larger /M option (to increase the cache allocation).

#### **48: The alternate collating sequence definition is invalid**

The MicroKernel returns this status code for the following reasons:

The first byte of an Alternate Collating Sequence (ACS) definition (the identification byte) does not contain the hexadecimal value AC (for user-defined ACSs), AD (for locale-specific ACSs), or AE (for international sorting rules support). Make sure that the first byte contains the appropriate value.

You set the Create File Version option set to v5.x, and you attempted to create a file that contains a key with a locale-specific ACS. Pre-v6.0 files do not support locale-specific ACSs.

#### **49: The extended key type is invalid**

The MicroKernel returns this status code for the following reasons:

You tried to create a file or an index with an invalid extended key type.

You tried to assign an Alternate Collating Sequence (ACS) to a BINARY key or key segment. You can assign an ACS only to a STRING, LSTRING, or ZSTRING key type. You defined an index requiring an ACS, but no ACS definition exists either in the file or in the key definition passed in the data buffer.

You attempted to create a key segment with both the Case Insensitivity and the ACS flags set, and the MicroKernel is configured to create files in v5.x format. This combination is invalid for v5.x files.

You set the Create File Version value to v5.x, and you attempted to create a file with a NUMERICSA or NUMERICSTS key. Pre-v6.x files do not support these key types.

OR

You set the Create File Version value to v6.x, and you attempted to use one of the new Pervasive.SQL 2000i data types, such as CURRENCY or TIMESTAMP. Pre-v7.x files do not support these key types. Increase the setting for this component.

To increase the Create File Version setting:

Click Start | Programs | Pervasive | Pervasive Control Center.

Expand Pervasive.SQL 2000i Engines and expand the desired database.

Expand Configuration, then expand Server, and click on Compatibility.

In the right hand frame, double-click on Create File Version.

Adjust Current with a higher file version.

Exit the configuration utility and restart the engines for the new settings to take effect.

#### **50: The file owner is already set**

The application tried to perform a Set Owner operation on a file that already has an owner. Use the Clear Owner operation to remove the previous owner before specifying a new one.

#### **51: The owner name is invalid**

The MicroKernel returns this status code for the following reasons:

If the application received this status code from a Set Owner operation, the owner names specified in the key buffer and data buffer do not match.

If this status code occurred during an Open operation or a DROP TABLE statement, the application attempted to open a file that has an owner name assigned to it. The application must specify the correct owner name in the data buffer. Ensure that the owner name is null-terminated in the data buffer and that the data buffer length is set long enough to include the owner name plus the null terminator.

#### **52: An error occurred while writing to the cache**

This status code is obsolete in MicroKernel versions 6.0 and later.

While trying to make a cache buffer available, the MicroKernel attempted to write data to a disk from a file that was previously opened in Accelerated mode. The operating system returned an I/O error during the write. This generally indicates a hardware memory problem. Unload and reload Btrieve before you continue.

**53: The language interface version is invalid**

An application tried to access a file containing variable-length records with a language interface from Btrieve v3.15 or earlier.

**54: The variable-length portion of the record is corrupt**

During a Get or Step operation, the MicroKernel could not read all or part of the variable-length portion of a record. The MicroKernel returns as much data as possible to the application. This status code usually indicates that one or more pages used to store variable-length records are corrupt. Check the data buffer length the MicroKernel returns to see how much of the record was returned. Recover the damaged file as described in the Pervasive.SQL User's Guide.

**55: The application specified an invalid attribute for an AUTOINCREMENT key**

The data field indexed by an AUTOINCREMENT key can be part of a different segmented key only if the key number of the AUTOINCREMENT key is less than the key number of the new segmented key and the new data type flag referencing the field is not AUTOINCREMENT

**56: An index is incomplete**

An index can be damaged if a Create Index operation (31) or a Drop Index operation (32) is interrupted before it runs to completion. Perform a Drop Index operation to completely remove the damaged index from the file, then rebuild the index with the Create Index operation, if desired.

**57: An expanded memory error occurred**

This status code is obsolete in MicroKernel versions 6.0 and later.

Btrieve for DOS returns this status code if it receives an error from the Expanded Memory Manager. This status code usually means that the MicroKernel was unable to save or restore the memory mapping register context, indicating an incompatibility with another application that uses expanded memory.

**58: The compression buffer length is too short**

This status code is obsolete in Pervasive.SQL 2000i.

**59: The specified file already exists**

During a Create operation, the application specified -1 in the key number parameter and the name of an existing file in the key buffer parameter. To overwrite the existing file, remove the -1 from the key number parameter. To preserve the existing file, alter the filename specified in the key buffer parameter.

**60: The specified reject count has been reached**

The MicroKernel rejected the number of records specified by the reject count before a Get Next Extended, Get Previous Extended, Step Next Extended, or Step Previous Extended operation found the requested number of records that satisfy the filtering

condition. Check the first two bytes returned in the data buffer for the number of records that were retrieved.

#### **61: The work space is too small**

The Get Next Extended, Get Previous Extended, Step Next Extended, and Step Previous Extended operations use a buffer as work space. This status code indicates that the work space (set by default to 16 KB) is not large enough to hold the filtering data buffer structure and the largest record to be received. You will receive Status Code 0 if the work space is large enough to hold the filter/extraction expression and enough of the record to include all of the fields to be filtered or extracted.

#### **62: The descriptor is incorrect**

This status code is returned in the following situations:

The descriptor (data buffer structure), which is passed for a Get Next Extended, Get Previous Extended, Step Next Extended, or Step Previous Extended operation, is incorrect. The descriptor length (the first two bytes of the data buffer) on the extended operation call must be the exact length of the descriptor. This requirement does not apply to the data buffer length option, which can still be declared longer than necessary.

On a Stat Extended operation, the signature field in the data buffer is not set to 0x74537845, the subfunction field is not set to 0x00000001, or the namespace field is not set to 0x00000000.

On a Get Direct/Chunk or Update Chunk operation, the descriptor structure in the data buffer is incorrect, or is inconsistent either internally or with respect to the data buffer length.

ActiveX control's buffers are not cleared and reallocated. Use the Init method to clear and reallocate the control's buffers before the use of any extended operations in the code. In addition, if you are using AutoMode, it is necessary to establish logical position (GetLast, GetFirst, GetEqual, etc.) before making the call to Init.

#### **63: The data buffer parameter specified on an Insert Extended operation is invalid**

An Insert Extended operation provided an invalid buffer. Either the buffer length is less than 5 bytes, or the number of records specified is 0. Correct the buffer length or the number of records.

#### **64: The filter limit has been reached**

The MicroKernel returns this status code for the following reasons:

During a Get Next Extended, Get Previous Extended, Step Next Extended, or Step Previous Extended operation, a rejected record was reached; no other record can satisfy the given filtering condition, going in the direction that the operation specified. This is applicable only if the first segment of the key that the key number specified is also used as the first term of the filtering field.

The number of records to be retrieved is greater than the number of records present in the file that satisfy the filter condition. This option is specified in the data buffer of the extended operation.

#### **65: The field offset is incorrect**

The field offset in the extractor of a Get Next Extended, Get Previous Extended, Step Next Extended, or Step Previous Extended operation is invalid based on the length of

the retrieved record. Make sure that the field offset is a valid value (from 0 through the record length minus 1).

#### **66: The maximum number of open databases has been exceeded**

This status code is obsolete in Pervasive.SQL 2000i.

The MicroKernel tried to open files bound to too many MicroKernel databases. To avoid receiving this status code, you must set a higher value for the number of databases that the MicroKernel can open.

To increase the value for Maximum Databases:

Click Start | Programs | Pervasive | Pervasive Control Center.

Expand Pervasive.SQL 2000i Engines and expand the desired database.

Expand Configuration, then expand Server, and click on Access.

In the right hand frame, double-click on Maximum Databases.

Increase Current to accommodate the number of databases you want the MicroKernel to be able to open at one time. (Note: The MicroKernel requires approximately 800 bytes of memory for each open database.)

Exit the configuration utility and restart the engines for the new settings to take effect.

Refer to the Pervasive.SQL Advanced Operations Guide for more information about bound files.

#### **67: The MicroKernel cannot open the SQL data dictionary files**

The MicroKernel returns this status code for the following reasons:

An application attempted to use a data file that is bound to a the MicroKernel database, but the MicroKernel could not open one of the MicroKernel data dictionary files (FILE.DDF or, if the file has RI definitions, RELATE.DDF) or the configuration file (DBNAMES.CFG).

You attempted to create a file with the Replace option, and a bound MicroKernel data file with the same name and location already exists. However, the MicroKernel could not open the MicroKernel data dictionary file FILE.DDF, or the configuration file (DBNAMES.CFG).

If the data file has RI definitions, the DBNAMES.CFG file must be in the location specified in the DBNames Configuration Location option in the server configuration settings. Also, ensure that FILE.DDF and RELATE.DDF (if the file has RI definitions) are in the locations specified by the Working Directory option in the server configuration settings.

#### **68: The MicroKernel cannot perform the RI Delete Cascade operation**

The MicroKernel cannot enforce the Delete Cascade rule on a file under RI control because the record that the application attempted to delete has more than 16 levels of descendants. Delete records from the lower levels, and then try again to delete the record that the application was attempting to delete initially. Refer to the Pervasive.SQL Advanced Operations Guide for more information about RI.

#### **69: The Delete operation specified a file that is damaged**

The application encountered an error while the MicroKernel was attempting to enforce the Delete Cascade rule in response to a Delete operation. This status code indicates that the related file has been damaged and must be recreated. Refer to the

Pervasive.SQL Advanced Operations Guide for more information about RI and the Delete Cascade rule.

### **71: There is a violation of the RI definitions**

If you attempted an Insert operation on a file under RI control, a foreign key value in the record to be inserted does not have a corresponding primary key in the referenced file.

If you are performing an Update operation, there are two possible causes for this status code:

You attempted to change the value of a primary key.

You attempted to change the value of a foreign key to a value that does not exist for the defined primary key.

If you attempted a Delete operation, the restrict rule is enforced, and a primary key value in the record you are trying to delete references a foreign key in the referenced file. Refer to the Pervasive.SQL Advanced Operations Guide for more information about RI.

### **72: The MicroKernel cannot open the RI referenced file**

The referenced file cannot be found at the location specified by FILE.DDF and DBNAMES.CFG. Be sure that the referenced file is in one of the data file locations that the DBNAMES.CFG file specifies for the named database.

If the DBNAMES.CFG file is defined on a server, verify that the file location does not contain a drive letter.

If the DBNAMES.CFG file is defined for a workstation engine, make sure that the drive letters are the same (and map to the same locations) as specified in DBNAMES.CFG.

Refer to the Pervasive.SQL Advanced Operations Guide for more information about RI.

### **73: The RI definition is out of sync**

The MicroKernel returns this status code for the following reasons:

You tried to open a data file that is bound to a MicroKernel database, and the database to which the file is bound was not found in the DBNAMES.CFG file.

You tried to open a data file with RI (Referential Integrity) definitions that are bound to a MicroKernel database, and the table to which the file is bound was not found in the database FILE.DDF file, or the table location and filename do not match the file location and filename as configured in the DBNAMES.CFG or FILE.DDF file.

You attempted to modify a bound file, and the RI definition for that file disagrees with the definition in the RELATE.DDF file.

You attempted an Insert, Delete, or Update operation that would change a foreign key, if the file related to this file is out of sync (an attempt to open or modify the related file would have returned this same status code).

You attempted to create a file with the Replace option, and a bound MicroKernel data file with the same name and location already exists. However, the MicroKernel detected that the existing bound file was out of sync (that is, an attempt to open the existing file would have returned this same status code).

Check the RI constraints on your database. For information about how to do this, refer to the Pervasive.SQL User's Guide.

**74: The MicroKernel aborted the transaction**

This status code is obsolete in MicroKernel versions 6.0 and later.

This is an informative status code. A NetWare-based MicroKernel replaced an End Transaction operation with an Abort Transaction operation after detecting an error for a Transaction Tracking System (TTS) file inside the transaction. The MicroKernel then executed the Abort Transaction operation.

**76: There is a conflict on the referenced file**

An application attempted to perform an Update, Insert, or Delete operation on an RI-controlled file that references another file. The application cannot open the referenced file for RI checking because it is already open in Exclusive mode. Wait until the referenced file is closed or is opened in a mode other than Exclusive, and then retry the operation. Refer to the Pervasive.SQL Advanced Operations Guide for more information about RI.

**77: The application encountered a wait error**

This status code is obsolete in MicroKernel versions 7.0 and later.

This is an informative status code. You must retry the operation yourself; the MicroKernel does not automatically retry the operation. A client/server MicroKernel returns this status code in one of the following situations:

The application specified a wait lock bias for an operation, but another user has locked the requested resource.

The application is currently processing a wait transaction and tried to access a file that another user has locked.

When you are using the Btrieve Requester to access the MicroKernel, the Requester waits and retries if a requested resource is locked. When a server-based application is accessing the MicroKernel and the requested resource is locked, a wait is also required. In this case, the MicroKernel is expected to perform the wait. Because this would occupy the MicroKernel and lock out other users who might be trying to release the requested resource, the MicroKernel does not perform the wait. Instead, it returns this status code, and the server-based application must retry later.

**78: The MicroKernel detected a deadlock condition**

The application should clear all resources by aborting, ending the transaction, or releasing all record locks before proceeding. This breaks the deadlock, allowing other applications to access the resources for which they are waiting.

**79: A programming error occurred**

This status code is obsolete in MicroKernel versions 7.0 and later.

There is a malfunction that the MicroKernel cannot specifically detect or from which the MicroKernel cannot recover. Retry the operation. If the error persists, there may be system corruption; try to clear the system by rebooting, and then try the operation again.

**80: The MicroKernel encountered a record-level conflict**

The MicroKernel did not perform the Update or Delete operation because of a record-level conflict. For example, station A reads a record, station B reads the same record

and updates it, and then station A attempts to update the record. The application should reread the record prior to resending an Update or Delete operation. Alternatively, the application can employ record locks to avoid conflicts.

In key-only files, you receive this status code if the record is moved in the file b-tree after being read and before being updated or deleted. A record can move as a result of other records being inserted, updated, or deleted.

#### **81: The MicroKernel encountered a lock error**

The MicroKernel returns this status code in one of the following situations:

The application tried to unlock a record that is locked with a multiple record lock, but the record position stored in the data buffer does not correspond to any record locked in the associated file.

The application tried to unlock a single-record lock with a multiple-record lock or vice-versa.

#### **82: The MicroKernel lost positioning**

When performing a Get Next or Get Previous operation on a key with duplicates, the application tried to retrieve a record that was deleted or whose key value was modified by another application. Use a Get Equal or a Get Direct/Record operation to re-establish positioning. (See Status Code "44: The specified key path is invalid" on page 1-72 for a related positioning problem.)

#### **83: The MicroKernel attempted to update or delete a record that was read outside the transaction**

This status code is obsolete in MicroKernel versions 7.0 and later.

The application tried to update or delete a record within a transaction, but it did not read the record within the transaction. The application must read the record within the transaction before attempting to modify the data.

#### **84: The record or page is locked**

An Insert, Update, or Delete operation attempted to lock an index page to insert or delete a key value. Have your application check for this status code and retry the operation if the status code is returned.

When using a NetWare server engine, you can receive this status code when running an application on a Win95 client if the NetWare operating system runs out of record locks. To solve this, increase the "maximum record locks per connection" and, if necessary, the "maximum record locks" (system wide limit) on the NetWare server.

The application tried one of the following:

Applied a no-wait lock on a record that is currently locked by another application

Tried to access a file in a no-wait transaction while another application holds one or more active record locks in that file

Tried to update or delete a record locked by another application.

The application can use either of the following recovery methods:

Retry the operation until it is successful. This can be the simplest and quickest solution for a network with light to moderate use.

Use the wait option (+100/+300) instead of the no-wait option (in versions that support the wait option).

### **85: The file is locked**

The MicroKernel returns this status code in one of the following situations:

The workstation MicroKernel has a file open, and client machine that has the Requester loaded tries to open the same file via the server MicroKernel. The server MicroKernel cannot open the file because it cannot obtain exclusive access. The client machine that has the Requester loaded receives this status code.

In a workstation engine environment, the MicroKernel can return this status code on an Open, Insert, Update, or Delete operation for a file under heavy usage by multiple users or tasks. The MicroKernel must momentarily have exclusive access to the file during these operations, and it retries the operation several times before returning this status code. In this case, the application can reissue the operation. In addition, you can reconfigure the workstation MicroKernel with a lower Operation Bundle Limit and Initiation Time Limit to reduce the amount of time the MicroKernel keeps a lock on the file. Refer to the Pervasive.SQL Advanced Operations Guide for more information on how to do this.

While one user has a file locked in an exclusive transaction, another user attempts to lock all or part of that file.

When opened by a MicroKernel, a file is in transition into Continuous Operation mode. Retrying eventually works.

When opened by a MicroKernel, two data files have the same filename but different extensions (for example, INVOICE.HDR and INVOICE.DET). One file is open and in Continuous Operation mode, causing the MicroKernel to generate a delta file (for example, INVOICE.^^). The MicroKernel returns this status code when you attempt to open the second file. For this reason, we recommend naming your files with completely different names, not just reusing the same name with different extensions.

When opened by a Windows NT server MicroKernel using Microsoft File and Print Services for NetWare on behalf of a Win16 Windows client, the file was also opened simultaneously by a Win32 Windows NT or Windows 9x machine. This causes the server MicroKernel to open the same physical file using two different paths.

Without any pattern of occurrence, you may receive a status 85 when the file is closed because Anti-Virus software opens and locks the file to scan causing the next database operation to fail. To resolve, set the Anti-Virus software to not scan Pervasive SQL data files. Consult your Anti-Virus software manual for instructions on how to exclude files.

### **86: The file table is full**

This status code is obsolete in Pervasive.SQL 2000i. This setting is managed dynamically by the engine.

The number of different files that the MicroKernel can open at the same time has reached its maximum limit. To avoid receiving this status code, increase the value for the Maximum Open Files configuration option.

To adjust the value for Maximum Open Files:

Click Start | Programs | Pervasive | Pervasive Control Center.

Expand Pervasive.SQL 2000i Engines and expand the desired database.

Expand Configuration, then expand Server, and click on Access.

In the right hand frame, double-click on Maximum Open Files.

Increase Current to increase the number of different files the MicroKernel can open

at one time. (Note: The default is 10000 files and the maximum is 64000 files.) Exit the configuration utility and restart the engines for the new settings to take effect.

### **87: The handle table is full**

This status code is obsolete in Pervasive.SQL 2000i. This setting is managed dynamically by the engine.

You have either attempted to open more handles than the MicroKernel is configured to support, or the MicroKernel attempted to open more files than the operating system allows.

To configure the MicroKernel to allow more handles:

Click Start | Programs | Pervasive | Pervasive Control Center.

Expand Pervasive.SQL 2000i Engines and expand the desired database.

Expand Configuration, then expand Server, and click on Access.

In the right hand frame, double-click on Logical File Handles.

Increase Current to accommodate your request. (Note: The default is 100000 file handles.)

Exit the configuration utility and restart the engines for the new settings to take effect.

To configure your operating system to allow more handles, refer to your operating system documentation.

It is helpful to know the following details regarding the MicroKernel requirements for handles from the operating system. When the same file is opened multiple times, the MicroKernel uses only one operating system handle. However, if the file is in v6.x or later format and the file is shared via MEFS mode, the MicroKernel opens a second handle for the associated .LCK file.

If the file is in v5.x format, the MicroKernel might request a second handle, for the .PRE file. Also, if the file (in any format) is placed in Continuous Operation mode, the MicroKernel requests another handle for the delta file. If the file is extended, the MicroKernel requests an operating system handle for each of the extension files.

In the Btrieve v6.15 DOS or Microsoft Windows NT 4.0 environments, you may receive this status code when opening the 16th file in a DOS application running under Windows NT. There may be two solutions:

Btrieve File handle configuration may be set incorrectly in BTI.CFG. Check BTI.CFG for file handle setting (/h: and /f:) and increase those values.

Check the file= setting in CONFIG.NT. This file is in the WINNT\SYSTEM32 directory and raise the values.

NOTE: default value is 20.

### **88: The application encountered an incompatible mode error**

The MicroKernel returns this status code in one of the following situations:

If an application opens a file in Exclusive mode, all other applications receive this status code when they try to open the same file in any mode.

If an application opens a file in any mode other than Exclusive, all other applications receive this status code when they try to open the same file in Exclusive mode.

While using the MicroKernel Continuous Operation mode:

You attempted to remove a file from continuous operation, but the file is not in Continuous Operation mode.

You attempted to remove a file from continuous operation, but a different client placed the file into continuous operation.

You attempted to include two files in continuous operation that have the same name but different extensions.

You attempted to include a file in continuous operation, but the file is already in continuous operation.

### **89: A name error occurred**

This status code is obsolete in MicroKernel versions 5.0 and later.

BSERVER was loaded before you specified the short name to which the device was redirected. You must specify all short names that you want to share with the NET SHARE command before you start BSERVER.

### **90: The redirected device table is full**

This status code is obsolete in MicroKernel versions 6.0 and later.

The DOS Requester redirection table or server routing table is full. This occurs if you attach to additional servers or map additional drives after loading the Requester. Reload the Requester, specifying a larger value for either the Number of File Servers (/S) option or the Number of Mapped Drives (/R) option. This status code also occurs if you detach from a server and attach to a different server. Once a client has attached to a server, the Requester does not remove its name from the server routing table.

For a list of all available BRequest commands, type /? at the command line.

### **91: The application encountered a server error**

The MicroKernel returns this status code in one of the following situations:

The Requester cannot establish a session with the server. Either the client/server MicroKernel is not loaded or the server is not active.

The SPX drivers are not installed or are outdated.

The MicroKernel has reached the maximum limit for the number of sessions it can open at one time. To avoid receiving this status code, increase the value for the Number of Sessions configuration option.

To increase the Number of Sessions:

Click Start | Programs | Pervasive | Pervasive Control Center.

Expand Pervasive.SQL 2000i Engines and expand the desired database.

Expand Configuration, then expand Server, and click on Access.

In the right hand frame, double-click on Number of Sessions.

Increase Current to allow more sessions. (Note: The default is 1000 sessions; approximately 32 kilobytes of memory are required for each session.)

Exit the configuration utility and restart the engines for the new settings to take effect.

An application specified a path for a file and did not include the volume name in the path.

The MicroKernel Router has not been loaded, and the following situation has occurred: an application that uses both the MicroKernel Router and the MicroKernel to make remote calls (and which therefore includes the server and volume name when performing an Open operation) has attempted to open a remote file. Because the MicroKernel Router does not interpret the server name, the MicroKernel attempts to do so but cannot.

A communication or network addressing problem exists in your network environment, so the MicroKernel requests never reach their destination server address. Ensure that your client and server network components are up to date and certified for your network environment.

There is a conflict using the DOS BREQNT requester with the SPX protocol. To resolve this error:

Change directories to \pvsw\clients\dos\windosbox and run the setup utility. This loads the appropriate file for clients running Windows 9x or Windows NT. The setup program creates and places a README.TXT file in the \pvsw\W32DOXBOX directory. After installation, reboot the client.

For Windows NT users: open a command prompt and run a DOS Btrieve application.

For Windows 9x users:

Run the BTRBOX95.exe. A minimized dialog box appears, indicating that BTRBOX95 support is active. If you close this dialog box, it will unload BTRBOX95. You only need to run BTRBOX95.exe once even though you have multiple DOS sessions open. Open a command prompt and run your DOS Btrieve application.

For NetWare servers only:

The Maximum Packet Receive Size configuration option is inappropriate for your environment. For example, the setting should be 1500 for an Ethernet LAN or 4096 for a Token-Ring LAN.

To adjust the Maximum Packet Receive Size:

Open the sys:system/BTI.CFG file.

Locate the Btrieve Communications Manager option.

Ensure that the Maximum Packet Receive Size value is appropriate for your environment.

NetWare users may also receive this status code in the following situations:

The user count limit has been exceeded. Either close a session or upgrade your user count. For more information about purchasing and installing additive user counts, refer to the Pervasive.SQL User's Guide.

Ensure that the NDS network number is the same as the Internal Network Number viewed by BINDER.EXE output.

You ran BUTIL.NLM to roll forward a file using a log filename other than the default, and your BLOG.CFG file did not contain a correct entry such as "\dir\file.ext=vol:\dir\log.ext".

## **92: The transaction table is full**

This status code is obsolete in MicroKernel versions 7.0 and later.

The application exceeded the maximum number of active transactions. Use the configuration utility to specify a higher value for the Number of Transactions configuration option.

## **93: The record lock types are incompatible**

The application tried to mix single-record locks (+100/+200) and multiple-record

locks (+300/+400) in the same file at the same time. You must release all locks of one type before you can execute a lock of the other type.

#### **94: The application encountered a permission error**

The MicroKernel returns this status code in the following situations:

The application tried to open or create a file in a directory without the proper privileges. The MicroKernel does not override the network privileges assigned to users.

The designated server is in the server routing table, but your particular client is not logged into that server.

The system data source name (DSN) on the server has an error in the pathname to the data files.

A NetWare application tried to access a file using NetWare Runtime support with the given username. Specifically, one of the following situations exists regarding the supplied username:

The user is not a valid user on the NetWare Runtime server.

The user does not have the appropriate rights to access the file.

The username is ADMIN or SUPERVISOR. For security reasons, the MicroKernel does not enable you to use ADMIN or SUPERVISOR as a username when enabling NetWare Runtime support.

User is logged in to NetWare server with grace login (i.e., password has expired and a set number of grace logins are left). Change the user's password to a valid password (no grace login) and reconnect to the server with the valid password.

When using the Win32 Requester from a Windows NT/9x client machine to a NetWare server, you must use the same username for logging in to both the client machine and the NetWare server. You cannot be logged in to NetWare as SUPERVISOR or ADMINISTRATOR.

When using the Win32 Requester from a Windows NT/9x client machine using NetWare emulation to a Windows NT server, the server cannot use Microsoft File and Print Services for NetWare. This causes the requester to attempt authentication as though the server were a NetWare Runtime server.

It is recommended that you keep the default Requester setting of Yes on FPNW servers running Pervasive.SQL. You may receive a Status Code 94 if you change this setting to No when you are running the Btrieve Interface locally on the FPNW server and are using a local FPNW drive mapping or local FPNW UNC path.

You may also receive this status if you have Client for NetWare Networks installed on a Windows only network where no NetWare servers present. Remove the Client for NetWare Networks from the Windows 95 workstation.

#### **95: The session is no longer valid**

The server MicroKernel returns this status code for one of the following reasons:

The previously established session is no longer active due to an error at the client machine, at the server, or on the network. Verify that the client machine is still attached to the server, and then unload and reload the Btrieve Requester.

If you are using the SPX protocol:

The server MicroKernel has reached the maximum number of SPX sessions. Use the Monitor utility to check this statistic. To avoid receiving this status code, increase the

value for the Number of Sessions configuration option.  
To increase the Number of Sessions:

Click Start | Programs | Pervasive | Pervasive Control Center.  
Expand Pervasive.SQL 2000i Engines and expand the desired database.  
Expand Configuration, then expand Server, and click on Access.  
In the right hand frame, double-click on Number of Sessions.  
Increase Current to allow more sessions. (Note: The default is 1000 sessions;  
approximately 32 kilobytes of memory are required for each session.)  
Exit the configuration utility and restart the engines for the new settings to take  
effect.

This may be a time delay problem if the client machine does not receive a response  
back from the server in an appropriate time frame or after an appropriate number of  
retries. Refer to your network configuration documentation for information about  
increasing timeout and retry parameters. This is often necessary in a WAN  
environment or a LAN configuration with heavy network traffic.

For NetWare servers:

Verify that the Maximum Packet Receive Size configuration option is appropriate for  
your environment. For example, the setting should be 1500 for an Ethernet LAN.  
To adjust the Maximum Packet Receive Size:

Open the sys:system/BTI.CFG file.

Locate the Btrieve Communications Manager option.

Ensure that the Maximum Packet Receive Size value is appropriate for your  
environment.

Ensure that the SPX timeout parameters are set as follows in both the client machine  
NET.CFG file and the server SPXCONFIG.NLM file:

SPX VERIFY TIMEOUT=54

SPX LISTEN TIMEOUT=108

SPX ABORT TIMEOUT=540

These three values must have a 1:2:10 ratio. You can increase these values to at  
most three times the default. If you continue to receive this status code after  
increasing these values, the problem is most likely not related to these settings.

For Windows NT servers, verify that the Maximum Packet Size registry setting is 576  
decimal or 240h:

The path to the MaxPktSize registry setting is:

HKEY\_LOCAL\_MACHINE\System\currentControlSet\

Services\NwlnkIPX\NetConfig\MaxPktSize.

If you continue to receive this status code after increasing the network timeout  
parameters, this status code usually indicates a problem with network  
communications. Verify that you have up to date network cards and drivers; for  
example, incompatible LAN card drivers can also cause this status code to occur.  
Consult your LAN administrator for network communication troubleshooting.

After installing Windows NT Service Pack 3 or 4, SPX requesters may fail and return  
this status code (95), Status 97, or Status 91. Refer to the Microsoft Knowledgebase  
article Q170517 for more information regarding this cause.

Choose Start | Run.

Type Regedit and click OK. The Registry Editor opens.

Make the following CHANGES to the server's registry using Regedit:

HKEY\_Local\_Machine | System | CurrentControlSet | Services | NwLnkIpx |  
<Network Card> | MaxPktSize = 240 Hex.

HKEY\_Local\_Machine | System | CurrentControlSet |  
Services | NwLnkIpx | <Network Card> | NetworkNumber = <Non-Zero Value>

ADD the following registry entry:

HKEY\_Local\_Machine | System | CurrentControlSet |  
Services | LanManServer | Parameters | MinClientBufferSize regdword = 500  
decimal.

### **96: A communications environment error occurred**

The MicroKernel returns this status code for the following reasons:

You tried to attach to the MicroKernel on a server, but the SPX connection table or the MicroKernel client table is full. To avoid receiving this error, increase the value for the Number of Sessions option.

To increase the Number of Sessions:

Click Start | Programs | Pervasive | Pervasive Control Center.

Expand Pervasive.SQL 2000i Engines and expand the desired database.

Expand Configuration, then expand Server, and click on Access.

In the right hand frame, double-click on Number of Sessions.

Increase Current to allow more sessions. (Note: The default is 1000 sessions; approximately 32 kilobytes of memory are required for each session.)

Exit the configuration utility and restart the engines for the new settings to take effect.

An application that calls the MicroKernel can return this status code if the DBNAMES.CFG file contains a named database definition specifying a data location on a different server.

You may also receive this status code in an environment with multiple servers with Btrieve running on them. When doing a Begin Transaction using BREQUEST 6.15, the requester checks its list of all known attached Btrieve file servers. Then it attempts to create an SPX session so that it can check the server to see if it is configured for transactions. This implies that all the servers have to be configured for at least the same number of remote sessions as well as being configured for the same number of transactions.

### **97: The data buffer is too small**

The application either tried to read or write a record that is longer than the current allowed settings for the MicroKernel or the Btrieve Requester, as follows:

For an Update, Insert, or Create operation, the application receives this status code if the data buffer length it specifies for the record exceeds the message buffer length.

For a Get, Step, or Stat operation, the application receives this status code if the message buffer is shorter than the length of the data the MicroKernel would return, regardless of the data buffer length specified in the application.

For a Get Chunk or Update Chunk operation, the total size of the retrieved or updated chunk exceeds the message buffer length.

To increase the Communication Buffer Size:

Click Start | Programs | Pervasive | Pervasive Control Center.

Expand Pervasive.SQL 2000i Engines and expand the desired database.

Expand Configuration, then expand Server, and click on Communication Buffer Size.

In the right hand frame, double-click on Communication Buffer Size.

Increase Current. (Note: The maximum value is 65153KB.)

Exit the configuration utility and restart the engines for the new settings to take effect.

DOS Requesters only: Reload the Btrieve Requester and specify a higher value for the message buffer size. This is done using the

/D parameter which is documented in Getting Started with Pervasive.SQL.

For Windows NT servers, verify that the Maximum Packet Size registry setting is 576 decimal or 240h. The path to the MaxPktSize registry setting is

HKEY\_LOCAL\_MACHINE\System\currentControlSet\

Services\NwlnkIPX\NetConfig\MaxPktSize.

For more information about configuring the message buffer size, refer to the Pervasive.SQL Advanced Operations Guide.

### **98: The MicroKernel detected an internal transaction error**

For MicroKernel v6.0 and earlier: a NetWare-based MicroKernel detected an error while executing the operation on a NetWare Transaction Tracking System (TTS) file. The application can perform only an Abort Transaction operation at this point.

### **99: The Btrieve Requester is unable to access the NetWare Runtime server**

This status codes is returned in the following situations:

You enabled NetWare Runtime server support and the Requester either detected no existing connection or could not find a valid login username. SUPERVISOR and ADMIN are not valid usernames, even if supplied with the correct password. If the Requester cannot find a login username other than SUPERVISOR or ADMIN, there is no valid name to pass.

You tried to access a server with the NetWare Runtime server support option disabled and you do not have an existing connection to that server.

When accessing Windows NT 4.0 Server from a client machine, you may receive this status code if the user moved from a NetWare server to a Windows NT Server. To resolve, remove the "SatEntry" (referencing the NetWare Server) in BTI.ini (for 16-bit applications) and/or the Registry (for 32-bit applications) under Communications Requester.

NOTE: The "SatEntry" is a line that resembles "SatEntry1=xxxxxxx-01,2,3"

### **100: No cache buffers are available**

This indicates that the MicroKernel has used all the cache buffers it allocated at load time. The common approach to this problem is to increase Cache Allocation in the configuration options and then reload the MicroKernel.

To increase Cache Allocation:

Click Start | Programs | Pervasive | Pervasive Control Center.

Expand Pervasive.SQL 2000i Engines and expand the desired database.

Expand Configuration, then expand Server, and click on Performance Tuning.

In the right hand frame, double-click on Cache Allocation Size.

Increase Current by entering Kilobytes of memory to allocate cache. Note: The default is 1024KB (= 1MB).

Exit the configuration utility and restart the engines for the new settings to take effect.

**101: Insufficient operating system memory is available**

This indicates that there is not enough operating system memory available to perform the requested operation. To fix this problem, perform one or more of the following:

Go to Performance tuning in the server configuration and decrease the value for the Cache Allocation configuration option.

Add memory to the server.

For more information, see Status Code "1002: A memory allocation error occurred" on page 1-109, a similar status code that the Windows-based MicroKernel returns in the same situation.

**102: Insufficient stack space is available**

This indicates that the MicroKernel has run out of stack space. To increase the amount of stack space available to your application, re-link the application, setting the stack size to a higher value. The MicroKernel returns this status code only to Windows-based applications that call WBTRCALL.DLL, or applications that call the Btrieve interface on the local server.

**103: The chunk offset is too big**

The MicroKernel returns this status code in one of the following situations:

A Get Direct/Chunk operation specified an offset beyond the end of the record, either explicitly or using the next-in-record bias to the subfunction value. Unless the MicroKernel returns this status code while processing the first chunk, the operation was partially successful. Check the data buffer length parameter immediately after the call to see how much data was retrieved (and therefore how many chunks).

An Update Chunk operation specified an offset that is more than one byte beyond the end of the record. This status code indicates that the MicroKernel has made no changes to the record.

An Update Chunk operation with an Append subfunction causes a record length to exceed its operating system file size limit. The MicroKernel has made no changes to the record.

**104: The MicroKernel does not recognize the locale**

During a Create or Create Index operation, the operating system was not able to return a collation table for the country ID and code page specified. Ensure that the application specified the locale's country ID and code page correctly and that the operating system is configured to support the country ID and code page.

**105: The file cannot be created with Variable-tail Allocation Tables (VATs)**

An application tried to create a file with Variable-tail Allocation Tables (VATs) but without variable-length records (a precondition for files to use VATs). This status code applies to key-only files as well as to regular data files.

**106: The MicroKernel cannot perform a Get Next Chunk operation**

An application called the Get Direct/Chunk operation to retrieve a chunk from a record and used the next-in-record bias on the descriptor subfunction. However, after the application established its positioning in the record (and prior to this call), the target record was deleted.

**107: The application attempted to perform a chunk operation on a pre-v6.0 file**

An application tried to use either a Get Direct/Chunk operation or an Update Chunk operation on a file in pre-v6.0 format.

**109: An unknown error was encountered either creating or accessing a semaphore**

The Windows NT platform of the workstation MicroKernel attempted an operation using incompatible versions of the DLLs. Shut down the MicroKernel and make sure that you are using the most recent versions of the DLLs.

**110: The MicroKernel cannot access the archival logging configuration file**

The archival logging configuration file (BLOG.CFG) contains entries for the data files on the drive for which you want to perform archival logging. The MicroKernel returns this status code for the following reasons:

The MicroKernel cannot find the BLOG.CFG file. Ensure that the file is in the \BLOG directory in a real root directory of the physical drive that contains data files you want to log. (That is, do not use a mapped root directory.) If your files are on multiple volumes, you must create a \BLOG directory on each volume.

The MicroKernel cannot open the BLOG.CFG file. Either the file is locked or it does not exist.

The MicroKernel cannot read the BLOG.CFG file. Either the file does not use the correct format or it is corrupt. Refer to the Pervasive.SQL Advanced Operations Guide for information about the format of the BLOG.CFG file.

**111: The specified filename was not found in the archival logging configuration file**

The MicroKernel cannot find the specified file in the BLOG.CFG file. The file must be specified in the BLOG.CFG file on the same physical drive. By default, the MicroKernel names the archival log file the same as the logged file, but with a .LOG extension. However, you can specify a different filename for the archival log file in the BLOG.CFG file. Ensure that the BLOG.CFG file indicates the correct filename for the archival log and ensure that the archival log file exists.

**112: The specified file is in use by another client**

Before the MicroKernel can perform a roll forward, the file must be in the same state it was in when it was last backed up. If another client changes the file, you must restore the file again before rolling forward.

**113: The MicroKernel cannot find the archival log for the specified file**

The MicroKernel cannot find the archival log file associated with the specified file. By default, the MicroKernel names the archival log file the same as the logged file, but with a .LOG extension. However, you can specify a different filename for the archival log file in the BLOG.CFG file. Ensure that the BLOG.CFG file indicates the correct filename for the archival log and ensure that the archival log file exists.

**114: The archival log for the specified file is invalid**

The archival log associated with the specified file is not a valid archival log file. By default, the MicroKernel names the archival log file the same as the logged file, but with a .LOG extension. However, you can specify a different filename for the archival

log file in the BLOG.CFG file. Ensure that the BLOG.CFG file indicates the correct filename for the archival log and ensure that the archival log file exists.

### **115: The MicroKernel cannot access the archival logging dump file**

The MicroKernel cannot access the archival logging dump file for one of the following reasons:

The filename indicated for dumping entries in an archival log is not a valid filename. Be sure this filename does not contain a volume specification. The dump file is created on the same volume as the log file.

The caller does not have access rights to the dump file.

The MicroKernel cannot open the file because another user has opened the file using an exclusive operating system lock.

### **116: The file is owned by another Microkernel engine acting as a Gateway**

The MicroKernel cannot contact the engine running on the gateway computer even though it can read the locator file. This might occur the following reasons:

When the two computers are separated by a router so they can both see the server, but cannot see one another. Try the following:

Use the Gateway Locator utility to identify the owner of the gateway.

Use either the InstallScout or SmartScout utility to connect to that computer.

You may have attempted to open a file with two different workgroup engines that are mapped to the files using different share names. The MicroKernel attempts to correct this, but cannot do so in all cases. Make sure each computer is mapping to the same share name.

### **130: The MicroKernel ran out of system locks**

This status code is obsolete in MicroKernel versions 6.15 and later.

This status code can indicate a temporary condition in which no system locks are currently available. The following are example cases:

A single client is performing a very large transaction, in which thousands of records are being modified.

Many clients are performing large transactions concurrently.

A client can receive this status code whether or not it is in a transaction. In some cases, a client can simply retry the failed operation. If other clients have released system locks in the interim, the retried operation may succeed. If a client in a transaction receives this status code, end or abort the transaction. If the transaction is very large, consider breaking it into multiple, smaller transactions. You can also use the Setup utility to lower the number of system locks devoted to explicit locking. To do so, lower the values assigned to the Number of Locks and/or Number of Sessions configuration options.

### **132: The file has reached its size limit**

The MicroKernel returns this status code in one of the following situations:

The file size reached its 64G limit.

An operation attempted to allocate more than 16,777,216 pages to a data file.

A data file has remained in continuous operation for a lengthy period of time, causing

its delta file to exceed the operating system limit.

A pre-v6.15 data file has reached the operating system file size limit.

**133: More than 5 concurrent users attempted to access the same data file**

In the Pervasive.SQL 2000i SDK for a workstation environment, you attempted to access a data file with more than five MicroKernels at the same time. The Pervasive.SQL 2000i SDK for a workstation environment limits the number of concurrent users of a file to five engines.

**134: The MicroKernel cannot read the International Sorting Rule**

The MicroKernel returns this status code for the following reasons:

The ISR is not found in the COLLATE.CFG file.

The COLLATE.CFG file is missing or corrupt.

The MicroKernel cannot read the ISR from the COLLATE.CFG file.

**135: The specified ISR table is corrupt or otherwise invalid**

The MicroKernel found a readable COLLATE.CFG file, but the specific International Sorting Rule table is invalid.

**136: The MicroKernel cannot find the specified Alternate Collating Sequence in the file**

The MicroKernel returns this status code in the following situations:

You tried to create an index that uses an Alternate Collating Sequence (ACS), but the MicroKernel cannot locate an ACS with the specified name in the file.

You called a Step Next Extended, Get Next Extended, Step Previous Extended, or Get Previous Extended operation and specified an ACS name, but the MicroKernel cannot locate an ACS with the specified name in the file.

**138: The NULL indicator position is invalid**

In order to ensure accessibility to your data from all of the Pervasive.SQL access methods, the NULL indicator segment (NIS) must appear immediately before the data segment that the NIS indicates.

A NIS cannot be indicated by another NIS.

**139: The MicroKernel has detected an unacceptable value in the key number**

Certain operations either use, or reserve the use of, the key number parameter as a subfunction number, rather than as a means to specify a file index to be used with the operation. (Note: This is also done in the GetEqual operation). This status code is returned if an application does not specify a valid subfunction number (via the key number parameter) to one of these operations:

You issued a Begin Transaction operation with an invalid key number.

You issued an End Transaction operation with an invalid key number.

You issued an Abort Transaction operation with an invalid key number.

You issued a Start Extended operation with an invalid key number.

**140: The savepoint ID is invalid**

The application supplied an invalid savepoint ID in a nested transaction operation. Ensure that you specify valid savepoint IDs.

**143: The MicroKernel cannot allow unauthorized access to files in a secure MicroKernel database**

You attempted to open a data file bound to a MicroKernel database that has security enabled. The MicroKernel does not allow access to such files, except through the MicroKernel.

The MicroKernel also returns this status code if you are not using the MicroKernel and all of the following are true:

You attempt to create a file with the Replace option.

A bound MicroKernel data file with the same name and location already exists.

The database to which the existing file is bound has security enabled.

**146: Duplicate system key**

The same key number was generated by two different threads generating system keys.

**147: The log segment is missing**

The MicroKernel cannot find a log segment that is necessary for rolling at least one file forward.

**148: A roll forward error occurred**

The MicroKernel encountered an error while rolling a file forward. Depending on the operating system, the MicroKernel reports an error message as follows:

The NetWare MicroKernel displays the message on the server system console and writes the message to the Pervasive Event Log (PVSU.LOG), which resides in SYS:\SYSTEM.

The Windows 95 and Windows NT workstation MicroKernel displays the message in the console message window and writes the message to the Pervasive Event Log (PVSU.LOG), which is located in the WINDOWS\SYSTEM or WINNT\SYSTEM32 directory.

The Windows NT server MicroKernel does not display a message, but writes the message to the Pervasive Event Log (PVSU.LOG) in the WINNT\SYSTEM32 directory.

**149: SQL Trigger**

While using the Btrieve API to alter database tables or entries, the system encountered SQL restrictions placed on the database by the SQL layer.

**161: The maximum number of user count licenses has been reached**

You attempted to open another session when you were at the limit of your user count license. Either close a session or upgrade your user count. For more information about purchasing and installing additive user counts, refer to Getting Started with Pervasive.SQL. You can use the Monitor utility to determine which users currently have connections to Pervasive.SQL. For information about the Monitor utility, refer to the Pervasive.SQL Advanced Operations Guide.

**162: The client table is full.** Try increasing the number of Active Clients in the MicroKernel Database Engine System Resource setting

This status code is obsolete in Pervasive.SQL 2000i. This setting is managed dynamically by the engine.

You may receive this status code due to one of the following:

You have run out of memory.  
Your number of Active Clients has exceeded 64K.  
Your setting for Active Clients in the Configuration utility is set too low, causing Pervasive.SQL to exhaust its client table space.  
To increase your Active Clients setting:

Click Start | Programs | Pervasive | Pervasive Control Center.  
Expand Pervasive.SQL 2000i Engines and expand the desired database.  
Expand Configuration, then expand Server, and click on Access.  
In the right hand frame, double-click on Active Clients.  
Increase Current to the number of clients that access the server engine at any one time.  
Exit the configuration utility and restart the engines for the new settings to take effect.

### **163: The NULL indicator cannot be the last segment**

The NULL indicator segment (NIS) cannot be the last segment of the key descriptor.

### **3000: The MicroKernel router encountered a memory allocation error**

The MicroKernel router could not internally allocate memory. Check the Pervasive Event Log (PVSW.LOG) for more information.

### **3001: Local access is unavailable to the MicroKernel router**

A call to the MicroKernel failed. This is most likely the result of an incorrect configuration. For example, if the MicroKernel is unavailable because of configuration settings and the file to be opened is local, then the MicroKernel router returns this status code.

### **3002: The MicroKernel router resource DLL is unavailable**

The MicroKernel resource DLL is either missing or incompatible with the current version of the router. If this happens, the MicroKernel reverts to its default settings and continues to run. Check the Pervasive Event Log (PVSW.LOG) for more information.

### **3003: The MicroKernel router detected an incompatible network component**

The networking services component is not compatible with this version of the MicroKernel router. The most likely cause is that the Networking services DLL has been replaced by an older version. Check the Pervasive Event Log (PVSW.LOG) for more information.

### **3004: The MicroKernel is not responding to the MicroKernel router**

The MicroKernel is not responding to requests from the MicroKernel router. Verify that the MicroKernel is running.

### **3005: The MicroKernel router encountered an operating system error**

The MicroKernel router encountered an unexpected error from the operating system, such as a shared memory failure. Check the Pervasive Event Log (PVSW.LOG) for more information.

### **3006: The MicroKernel router detected an invalid session**

The session information contained in a position block is invalid. This status code occurs for one of the following reasons:

- The application is trying to use a position block for a file that resides on an engine and for which the connection has been previously terminated via a B\_STOP (25) or B\_RESET (28) operation.
- The application is trying to use a position block for a file that resides on a remote server and that encountered an abnormal network error on a previous operation.

### **3008: Invalid configuration for MicroKernel router**

This status code results from an invalid configuration. For example, if the Local, Remote, and Thunk options have all been set to Off, the router has no valid path to take. If this happens, set one of these options to On, depending on your setup.

### **3009: NETinterface.DLL is not loaded**

The MicroKernel router could not find NETAPI.DLL.

### **3010: NetWare API DLLs are not loaded**

The MicroKernel router could not find the NetWare API DLLs.

### **3011: Thunk not possible - the Win32 MicroKernel router could not be found or is incompatible**

This status code is only relevant to the Win16 MicroKernel router. You receive this status code when the Use Thunk option is set to On and the thunk path is not accessible. Possible causes include:

- The Win32 MicroKernel Router DLL does not exist or is not compatible.
- The Win32 routines failed to initialize.

Check the Pervasive Event Log (PVSW.LOG) for more information.

### **3012: Local engine is not accessible to the MicroKernel router**

Access to the local engine is not possible because it is not loaded or could not be launched. You can receive this status code if you try to access a local file on a client and you do not have a MicroKernel workstation engine installed or if you try to access a local file on a server and the MicroKernel server engine is not running.

If you have only a server engine installed and your Pervasive Event Log (PVSW.LOG) contains Status Code 3012 warning entries, perform the following steps:

To Turn Off Local Engine support (for clients and workstations only):

- 1 Click Start | Programs | Pervasive | Pervasive Control Center.
- 2 Expand Pervasive.SQL 2000i Engines and expand the desired database.
- 3 Expand Configuration, then expand Client, and click on Access.
- 4 In the right hand frame, double-click on Local, change the Current value to Off, and click OK.
- 5 In the right-hand frame, double-click on Requester, change the Current value to Off, and click OK.
- 6 In the right-hand frame, double-click Target Engine, change Current to Server only, and click OK.
- 7 In the left-hand frame, expand Client-16 bit and click on Access.
- 8 In the right-hand frame, double-click Use Local MicroKernel Engine, change Current value to Off, and click OK.
- 9 Make sure the Use Remote MicroKernel Engine value is set to on and that the remote engine is set to accept remote requests.
- 10 Exit the configuration utility and restart the engines for the new settings to take effect.

### **3013: The remote engine is inaccessible to the MicroKernel router because the networking component is not loaded**

Access to the remote engine is not possible because the MicroKernel router could not initialize the networking component. Possible causes include:

- The Remote option is incorrectly set to Off.
- The required networking component could not be found or is not compatible.

### **3014: The MicroKernel router cannot find an engine**

The MicroKernel router could not complete the operation because it did not find an engine (local or remote) that could process the operation. Possible causes include:

- The target network operating system is not available.
- The target engine is not available.

- In a Windows 3.x environment, you are using a shared drive name that contains a space. To resolve the condition, remove the space from the shared drive name. For example, if you connect to a server named "D Drive", change the share name to "D\_Drive".

### **3015: The MicroKernel router encountered an initialization error**

An unexpected error occurred during the initialization of the MicroKernel router. Check the Pervasive Event Log (PVSW.LOG) for more information.

### **3016: The MicroKernel router encountered an internal error**

The MicroKernel router encountered an internal error. Check the Pervasive Event Log (PVSW.LOG) for more information.

### **3017: Data buffer of the local engine is too small**

The requester has more data to send the MicroKernel than the MicroKernel buffers can handle. This status code is only relevant to an engine running locally and does not apply to client/server environments. Some possible solutions include:

- Specify higher values for the Communications Buffer Size and the MicroKernel Communication Buffer Size parameters in the Configuration buffer size option.
- Use smaller buffers in your operations.

### **3018: The file is already closed**

The MicroKernel router is shut down and is not accepting any more requests.

### **3019: The MicroKernel router encountered a semaphore error**

A semaphore error occurred while trying to establish contact with the local engine. Check the Pervasive Event Log (PVSW.LOG) for more information.

### **3020: An error occurred while loading the MicroKernel**

You receive this status code when an error occurs while loading the MicroKernel or when access to the server and client machine shared memory objects are unavailable. Check the Pervasive Event Log (PVSW.LOG) for more information.

### **3021: The MicroKernel router received a badly formatted data packet**

The MicroKernel router rejected the response from the engine because it was badly formatted. Check the Pervasive Event Log (PVSW.LOG) for more information.

### **3022: The MicroKernel router could not send the request to the remote engine because the specified data buffer length resulted in a data packet object that is too large**

A request to the MicroKernel router specified a data buffer length that resulted in a data packet size greater than 64 KB, which is the maximum. This can occur even if the data buffer length for the request is smaller than 64 KB because the MicroKernel router adds some packet overhead as it forms the data packet. Check the Pervasive Event Log (PVSU.LOG) for more information.

### **3023: The task table of the MicroKernel router is full**

An internal table used by the MicroKernel router to manage tasks is full. This error should only occur in Win16 environments and indicates that a large number of applications are using the Win16 MicroKernel router at the same time.

### **3025: Cannot load IDS client requester**

The MicroKernel router could not load the IDS client requester. The IDS client requester is W3ICRxxx.DLL, where xxx is a revision number (100, 101, etc.). The IDS client requester needs to be located in a directory listed in your PERSASIVE\_PATH or PATH environment variable.

### **3026: Incompatible IDS client requester**

The MicroKernel router found an incompatible IDS client requester. The IDS client requester was found, but is a version that cannot be used by the MicroKernel router, or the file is corrupt.

### **3027: Connection failure to IDS target server**

The MicroKernel router could not connect to the IDS target server.

Be sure that IDS is running on the target server. If you are connecting to the IDS target by a host name, ensure that the name is spelled correctly and can be resolved to a TCP/IP address. If you are connecting by a TCP/IP dotted notation address, be sure that the address is correct and accessible from the client. You can use SmartScout as a diagnostic tool.

### **3028: Login failure to IDS target server**

The MicroKernel router could not login to the IDS target server.

Be sure that the user name, password, and database set name used are correct. Some applications may prompt for this information; others may require MicroKernel Router configuration parameters to be set. See Getting Started with Pervasive.SQL for more information regarding MicroKernel Router configuration parameters.

### **3029: IDS Not Allowed**

This code will be returned if the path indicates an IDS request and either:

- the key number on the open request indicates 'local only'; OR

- the configuration setting for Use IDS is set to Off.

### **3030: Remote access not allowed**

This code is returned if the path is remote, the configuration setting for Local has been set to Off, and either:

- the key number on the open request indicates 'local only'; OR
- the configuration setting for Requester is set to Off.

### **3103: Server name not found by Pervasive Network Services Layer**

The search for a target server name was unable to resolve an address after searching NDS, bindery, named pipes, and DNS. Some possible causes include:

- No common communication protocol is available between the server and the client. Check the Supported protocols setting within the Communication protocols option for both the client and server. Make sure there is a common protocol (either SPX or TCP/IP) between the server and the requester.

- For Windows NT/2000:

- The MicroKernel engine not running.

- Server only: Named pipes are used in these two operating systems to resolve server names. The networking at the operating system level was unable to find the target named pipe.

- For NetWare:

- The Server Addressing Protocol (SAP) filtered out the server name. SAP is used in NetWare 3.x and can be used in version 4.x. This type of filtering is common in WAN environments.

- NWCALLS.DLL and/or NWIPXSPX.DLL is missing when using the Win16 client. This is especially likely if the Win32 clients are functioning but the Win16 clients do not. These are NetWare system files that can be obtained from Novell web site (<http://www.novell.com>).

- For NetWare using TCP/IP: The server name is not in DNS.

- For Novell NetWare 4.11 and Microsoft Windows 98 installs of Pervasive.SQL 2000 Client:

- When the Pervasive System Analyzer tries to perform the first communication/functionality test, a status 3103 is returned. This status code may be caused by having no supported protocols set in the Pervasive.SQL 2000 client configuration utility. If Microsoft SPX is added as a supported protocol, the installation should complete without error.

### **3104: The Pervasive Network Services Layer encountered a permission error**

For Windows NT only: Permission to access the target named pipe is denied. If you receive this status code:

- Verify that you have access to the inter process communication mechanism.
- Ensure that your passwords are identical if you have the same user name on two systems.

### **3105: No available transport protocol for the Pervasive Network Services Layer**

No transport protocol that is common to both the target server engine and clients is available. For example, this status code could be caused by a client using SPX when the server engine only has TCP/IP available. Check the Supported protocols setting within the Communication protocols option for both the client and server. For more information on Communication protocols, refer to the Pervasive.SQL Advanced Operations Guide.

### **3106: The Pervasive Network Services Layer encountered a connection failure**

The Pervasive Network Services Layer was able to establish a transport connection at the client side, but the connection attempt at the target side failed. Some possible causes are:

- The MicroKernel is not running on the server.
- The network is overloaded.
- The connection path is invalid.
- You have more than one mapped drive to the same server.
- You are trying to access an engine on a Windows NT server and the Accept Remote Requests setting of the Btrieve Communications Manager on that server is set to Off. See the Pervasive.SQL Advanced Operations Guide for more information.

Check the Pervasive Event Log (PVSU.LOG) for more information if you receive this status code.

### **3107: The Pervasive Network Services Layer is out of memory**

The Pervasive Network Services Layer has insufficient memory to continue. Verify that you are not in a low memory condition.

### **3108: The Pervasive Network Services Layer detected an invalid session**

The application attempted to use a network session that was not recognized by the Pervasive Network Services Layer. If the error persists, contact Technical Support at Pervasive Software.

### **3110: The network layer is not connected**

The application attempted to use a network connection that was no longer active. This happens when the session with the Pervasive Network Services Layer is still valid, but was dropped by the network. Stop and restart the application.

### **3111: Failure during send to the target server**

The Pervasive Network Services Layer attempted to send an application request to the target server and encountered a network specific error from the target. Check the Pervasive Event Log (PVSW.LOG) for more information.

You may receive this error when attempting to connect to a Workgroup engine on Windows 95 if the machine has not been upgraded to Winsock 2. If you are not running Winsock 2 on your Windows 95 computer with Workgroup engine installed, you should download the WinSock 2 update from Microsoft:

### **3112: Failure during receive from the target server**

The Pervasive Network Services Layer attempted to receive data from the target server and encountered a network specific error. Check the Pervasive Event Log (PVSW.LOG) for more information.

You may receive this error when attempting to connect to a Workgroup engine on Windows 95 if the machine has not been upgraded to Winsock 2. If you are not running Winsock 2 on your Windows 95 computer with Workgroup engine installed, you should download the WinSock 2 update from Microsoft.

### **3114: The routing table of the Pervasive Network Services Layer is full**

The Pervasive Network Services Layer Server Routing Table is full. This table normally grows dynamically as needed. Verify that you are not in a low memory condition.

### **3115: Pervasive Network Services Layer encountered a transport failure**

The Pervasive Network Services Layer encountered a WinSock (Win32, Win16) error while creating the local transport endpoint on the client side. Check the Pervasive Event Log (PVSW.LOG) for more information.

### **3119: No authentication context is available**

NSL was unable to return the access context on the specified session for the supplied object. There are several reasons why NSL could not obtain the context and it varies depending on the transport type.

- NamePipe - It is possible that NSL was able to locate the server via DNS but when it tried to make a NamePipe call to request authentication of the client and the Name Pipe communications failed. Ensure that NamePipes is configured correctly for the network for both the client and server. See the Pervasive.SQL Advanced Operations Guide for more information on NamePipes configuration.

- Novell NetWare Client - NSL asks the NetWare Client if this workstation has access to the database on the NetWare server. Ensure the user is configured to access the database file on the server. See the Pervasive.SQL Advanced Operations Guide for more information on NetWare user configuration.

- NetWare - You can configure this to use RunTime Server Support. NSL will create a context using one of the following:

- the userID and password configured in the NetWare RTSS setting
- the userID and password already used on this workstation to connect to any server
- the workstation userID and password.

If you want to use this mechanism, RTSS must be enabled. See the Pervasive.SQL Advanced Operations Guide for more information on enabling RTSS and userID and password configuration.

#### **3124: Pervasive Network Services Layer task table is full**

For each user of the Network Services Layer, a task entry is created. If the maximum number of tasks has already been reached (512 and not configurable), this error is returned. Close any applications using Pervasive.SQL that are no longer needed.

#### **3125: The Pervasive Network Services Layer was unable to resolve the Novell Directory Services name into a NetWare server name**

The Pervasive Network Services Layer was unable to resolve the Novell Directory Services name into a NetWare server name. This occurs on Windows 95/98/ME workstations with the Microsoft Service for Novell Directory Services in cases where the system is unable to load the Windows 95 Requester support modules. Ensure that the Requester Win95 support modules are installed and available, and that NWCALLS.DLL is available on the Windows 95 workstation.

#### **3126: The Pervasive Network Services Layer was unable to resolve the given filename into a valid path**

Check that the given filename is a valid file format and path.

#### **3127: Failure during send to the IDS target server**

The IDS client requester attempted to send data to the IDS target server and encountered a TCP/IP specific error. Stop and restart the application. If the problem persists, consult your network administrator.

### **3128: Failure during receive from the IDS target server**

The IDS client requester attempted to receive data from the IDS target server and encountered a TCP/IP specific error. Stop and restart the application. If the problem persists, consult your network administrator.

### **3131: Pervasive Network Services Layer encountered an error while reconnecting**

This status code means that AutoReconnect was unsuccessful. The following messages will appear in PVSW.LOG:

- "3131: Reconnect failed. Client and Server Message IDs out of synchronization." The client and server did not agree on the current context. Because the integrity of the last request cannot be verified, the connection was terminated.
- "3131: Reconnect not attempted, AutoReconnect not enabled on either client or server." In this case, either the client, server or both has AutoReconnect disabled or your server version does not support AutoReconnect (pre-Pervasive.SQL 2000i). The reconnection attempt will not be made.
- "3131: Reconnection attempt timed out." This means that the client was unable to obtain a new connection to the server within the AutoReconnect timeout period.
- "3131: Reconnection negotiations failed after re-establishing network connection." This means that the client was successful in making a new network connection to the server, but communications failed during the Pervasive AutoReconnect negotiations.