NORTH PLAINS SYSTEMS INC.

TeleScope Design Documentation

Cross-Platform File Specifications

Revision 1.0

July 2, 1996

Table of Contents

1. Overview

As a software solution which operates on both Macintosh and Windows platforms, TeleScope has a requirement not only for storing the locations of files on both systems, but having a Macintosh able to locate and retrieve a Windows file and vice-versa. This is a difficult and complex problem, which has necessitated the creation of a platform-independent method of locating a file, and a format for storing this data into the TeleScope database in such a way that both Windows and Macintosh client software can access and understand it.

Additionally, the underlying network layer must be removed from consideration, because of the wide variety of platforms and file servers which need to be supported. Thus, the desired solution will allow use of NFS/Share, Novell, or any other network layer configuration which permits both Macintoshes and Windows machines to connect to shared network devices.

This is done using a combination of the *file_alias* and *file_location* fields in the *editorial* table, and the contents of the *share_mappings* table of the TeleScope database. Between them, the contents of these items are enough to allow both Macintosh and Windows computers to locate and access files on either platform, across a network.

2. File Alias

In the *editorial* table, the *file_alias* field is used by the Macintosh client to locate files. The contents of this binary field are a Macintosh Alias Record. On import, the Macintosh client will place an alias to the imported file into this field. Since the contents of a Macintosh Alias Record are unpublished and proprietary to Apple, the Windows client will not be able to use this field. Also, the Macintosh client cannot rely on the existence of this field, as the Windows client will leave the field null on import. Thus, this field is nothing more than a convenience for the Macintosh client. If it exists, the Macintosh client will use it to locate the file. If not, the Macintosh client must use the *file_location* field, as described in the next section.

3. File Location

In the *editorial* table, the *file_location* field is used by the Windows client to locate files, and by the Macintosh client when the *file_alias* field is not available. On import, the Windows client stores null into the *file_alias* field, and the file's location into the *file_location* field (as described below). The Macintosh client will store an alias to the file in the *file_alias* field, and also store appropriate data into the *file_location* field.

The *file_location* field is a 512-character field, in which the text is formatted as a series of 5 fields, each separated by an (ascii 0x03) separator character, as follows:

```
<machine name>0x03<share name>0x03<removable flag>0x03<volume name>0x03<path>
```

There are several possible scenarios for describing the locations of files, and each scenario places different information into the *file_location* field.

Shared File

If the file is shared (i.e. if it is located under a directory which is a Windows share), then the following information is placed into *file_location*:

Machine Name	Contains the name of the machine on which the file resides.
Share Name	Contains the name of the share under which the file is located.
Removable Flag	Contains no characters.
Volume Name	Contains no characters.
Path	Contains the path from the share directory to the file. This string must begin with a backslash character ('\'), and must not contain a drive letter.

File on Removable Media

If the file is on a removable media device (such as floppy, CD ROM, or Optical), then the following information is placed into *file_location*:

Machine Name	Contains no characters.
Share Name	Contains no characters.
Removable Flag	Contains "R" to indicate a removable device.
Volume Name	Contains the name of the removable volume.

Path	Contains the path to the file from the root of the removable volume, including the drive letter (usually “a:” for floppies, or “d:” for other removable volumes).

Local File, Not Shared

This is the degenerate case, where the file cannot be shared across the network (i.e. it is not located under a Windows share), and it is not on a removable volume. In this case, the file is only accessible to the Windows machine which imported it into the database. The following information is placed into *file_location*:

Machine Name	Contains the name of the machine which imported the file.
Share Name	Contains no characters.
Removable Flag	Contains no characters.
Volume Name	Contains the name of the volume on which the file is located.
Path	Contains the path to the file from the root of the volume, including the drive letter.

4. Share Mappings

The *share_mappings* table contains the information used by the Macintosh client to locate files imported by the Windows client. Since the *share_mappings* table contains network-wide connection information, the contents of this table are created and maintained by the System Administrator using the TeleScope Director software. The Windows client ignores this table, since all the information required to locate a file under Windows is contained in the *file_location* field.

The *share_mappings* table contains the following fields:

Name	Data Type	Description
share_name	Char(64)	Contains the name of the Windows share as used by the Windows client in the *file_location* field.
machine_name	Char(64)	Contains the name of the machine, as used by the Windows client in the *file_location* field.
appleshare_alias	Varbin	Contains an AppleShare Alias Record to an AppleShare volume or folder which matches the Windows share. This assumes that the Windows share is located on a server or machine which accessible both to the Windows networking architecture and to AppleShare. If this is not the case, then files will not be interchangeable across platforms.

The contents of this table determine how the Macintosh client locates files which were imported by a Windows client, by mapping between Windows share and machine names and AppleShare shared volumes and folders.

Thus, if the Macintosh client needs to get to the original file, it first checks the *file_alias* field for an alias record. If it finds one, then it can use this data and ignore the *file_location* field. If not, it needs to parse the *file_location* field for the file, to get the share name and machine name from the data. Given these two names, it can search the *share_mappings* table for an equivalent share, and retrieve the appropriate AppleShare alias. It then resolves this alias using standard Macintosh OS calls, and can use the path from the *file_location* field to locate the file.

When importing, the Macintosh client can do the reverse of this process, inserting the proper information into the *file_location* field given that it knows the locations of the files it is importing, and can compare against the *share_mappings* table to determine the share name and machine name for the files (if they are located under one of the shares in the *share_mappings* table).

A similar process is followed for removable volumes, with the Macintosh client inserting the appropriate information into the *file_location* field for files on removable devices (as described in section 3). As a courtesy when importing from removable volumes, the Macintosh client will determine if the file is located on a diskette or on another removable volume, and set the drive letter in the file path of the *file_location* field to “a:” for diskettes, and “d:” for other removable volumes. This gives the Windows client a semi-intelligent starting point when asking the user to insert removable volumes.