Developing File System Mini-Filters for Windows®

Overview

This class provides a hands-on lab for those that need to build a file system filter driver using the "mini-filter" architecture. The focus is on covering key issues in developing file system mini-filters that are typically important concerns in most file system filters. With a "real-world" type emphasis, the participant in the seminar is expected to spend much of the time adding real functionality to a prototype filter in order to explore these issues and look at pragmatic solutions to these specific issues.

The class emphasizes lab time, with short presentations followed by a more extended lab session. Students should expect to leave with a substantially improved understanding of the issues involved in developing such drivers.

Seminar Formats

This seminar is available as a 5 day lecture with labs, which can be customized with additional or different topics for presentation at your location. Please contact an OSR Seminar Coordinator for details on arranging an on-site seminar.

Target Audience

This course is targeted specifically at developers needing to build a file system min-filter driver, or those that wish to explore more issues involved in developing such drivers in a live system interactive environment.

Note: This is not an introductory/beginner class. Participants will be doing active development and investigating issues in the development of mini-filters. The bulk of this seminar focuses on student lab sessions, with discussions focused on specific issues and potential solutions to those issues. The class is intense and covers considerable material in a relatively short period of time. Students should be prepared for a very intense week of in-depth discussion and hands-on lab time.

Prerequisites

Participants are expected to have taken OSR's *Developing File Systems for Windows* seminar. If you have not attended the Developing File Systems for Windows seminar but feel you have enough real experience that covers the course topics of the seminar, we will take that into consideration prior to registration. Seriously folks, any mini-filter developer must first understand the vast complexities and subtle nuances of the file system interface, and standard Windows file systems. This seminar does not cover these topics. It is an advanced course for advanced developers.

Seminar Outline

1. **Introduction to Mini-Filters**
   This introductory session focuses on the basics of mini-filters, including their basic structure, model and a discussion of the basics. The lab will focus on the structure of the initial prototype skeleton mini-filter as well as how to build it.

2. **Installation of Mini-Filters**
   This introductory session will focus on the installation process for mini-filters. Starting with the basic INF file format, we discuss the basics of the INF file as well as how to extend the basic components to add additional parameters. The lab will focus on taking the mini-filter from the previous session, installing it and loading/unloading the mini-filter.

3. **Debugging Mini-Filters**
   This section dives into the pragmatic issues of debugging mini-filters and the various techniques the developer should consider as part of that process. The lab will focus on ensuring the filter in the previous section can be observed under the debugger as well as adding specific debugging information into the driver and observing it in the debugger, including the use of the filter manager kernel debugger extension.

4. **Dealing with Contexts**
   Filter Manager relies upon context to provide a mechanism for tracking mini-filter related state information. This section will discuss the issues and mechanisms for tracking and managing contexts. The lab will focus on augmenting the sample mini-filter to allocate and free context for the various operations.

5. **Local File Systems and Common Naming**
   Few things in Windows file systems create more unexpected confusion than naming issues in Windows. This section will discuss the most common naming scenarios specific to local file systems, including case sensitivity and the issues associated with it. The lab will focus on adding support for these scenarios in our sample file system filter.

6. **Remote File System Naming Issues**
   Building upon the discussion about local file system naming issues, this section switches to looking at the issues that arise when moving to network file systems (with an emphasis on LanManager, but also discussing DAV and NFS. The lab will focus on further augmenting the filter to support these scenarios.

7. **Hard Links**
   Various Windows file systems provide support for "hard links" - multiple names that point to the same file. This section describes these issues. The lab will focus on detecting hard links in both a local and network situation, as well as detecting the existence of such links as well as the creation of new hard links.

8. **File IDs**
   In order to allow rapid lookup of files, several file systems allow the opening of files by ID. This section discusses the issues involved in "open by ID" including the variants of this, as well as techniques for coping with this. The lab will focus on handling open by ID, as well as demonstrating some of the specific issues involved in performing open-by-id.

9. **Alternate Data Streams**
   Some file systems provide a mechanism of supporting multiple data elements ('alternate data streams') that often come as a surprise to those developing file system filter drivers. In addition to handling these, a mini-filter may find that using an ADS as an effective means of associating meta-data with the file. The lab will focus on detecting such opens, as well as using them to associate specific information with the given file and directory.

10. **Managing Reparse Points**
    A reparse point is a special kind of attribute associated with a file or directory that allows return of data during the create operation. This section discusses reparse points, how to use them in a filter driver and how they manifest. The lab will focus on intercepting reparse points and handling them within your reparse point filter.

11. **Dealing with Delete**
    In Windows file systems, delete is an "intention" that is indicated by a flag set in a file system data structure. From the perspective of a file system filter driver this presents some interesting challenges about knowing what happened (e.g., when the file is deleted.) The lab will demonstrate these issues as well as propose some options for a file system filter driver when dealing with deleted files.

12. **Issues in Rename**
    In Windows file systems, rename can be a surprisingly complex operation, bridging create (SL_OPEN_TARGET_DIRECTORY) and set information calls. It can (as a side-effect) optionally delete the target of the rename. The lab will deal with recognizing and handling rename issues in a file system mini-filter.

13. **Interactions Between Delete, Create, Rename and Open By ID**
    Combining the previous topics, this section focuses on the interactions that occur between create, delete, rename, and open by ID. The lab will demonstrate these issues and explore methods for dealing with them.

14. **Handling Read Operations**
    Dealing with I/O operations is common in file system filter drivers. This section deals with various types of normal I/O (paging, user cached, user non-cached.) The lab will focus on intercepting and handling the various types of read I/O in a mini-filter.

15. **Handling Write Operations**
    Building upon the discussion in the previous section we explore issues involved in write operations. The lab focuses on issues associated with write I/O in a mini-filter.

16. **Special Cases in I/O (MDL, Paging Files)**
    Building upon the previous discussion of normal user and paging I/O, this section discusses various special cases in dealing with I/O. The lab then builds upon this by further extending the sample filter driver to handle those cases.

17. **Directories**
    Directories are distinct from files and require special case handling. Detecting directories within your mini-filter, handling them, and enumerating them are all discussed and then demonstrated during the lab session.

18. **Oplocks and Byte Range Locks**
    This section delves into the issues surrounding byte range locks and oplocks. While these two concepts are generally disjoint they do overlap in practice and are thus covered together. The lab discusses how to deal with each of these and explores issues involved in dealing with them and using them in a mini-filter.

19. **File Sharing, Security and Attributes**
    One of the complex issues is the interaction between local security and remote security. This section will discuss all three issues from a pragmatic perspective. The lab builds upon this by adding logic to monitor each of these and then to examine the behavior difference in the remote file systems.

20. **Using User-Mode Services and Mini-Filters**
    A common technique in file system mini-filters is to interact with a user mode service. This section will cover the methods available and issues that need to be addressed. The lab section will augment the filter to interact with a user mode service and demonstrate some potential uses of such an extension.

21. **Participating in Transactions**
    Windows Vista adds support for Transactions in a variety of different ways. This section will discuss transactions, the Kernel Transaction Manager (KTM) and other transactional services, as well as how transactions are viewed by mini-filters. The lab will then modify the mini-filter to explicitly check for and handle transactions, providing further insight into the role they play in the file system environment.

22. Creating a Resource Manager in a Mini-Filter

In addition to coexist with transactions, a mini-filter can be extended to provide a resource manager and thus extend the transactional model to coexist with services that the mini-filter provides. The lab will extend the functionality of the mini-filter in such a way that it can interact with its user mode service within the context of its own transactional support, thus demonstrating the construction of a very simple resource manager.

23. Handling O/S Version Differences

Because the filter manager environment differs from version to version, it is important to understand these differences and how they affect your file system mini-filter. The lab will provide an opportunity to test the sample mini-filter in a variety of these environments and observe their behavior as well as describing techniques for providing functional backwards compatibility.