

Cloverleaf: Programming with Tool Command Language (TCL)

Course Description

This class is a one-week course of intensive education on interfacing technology, methods, and application training. The primary focus of this course is to build upon and extend the knowledge presented in the *Cloverleaf: Developing Cloverleaf Networks* class using Cloverleaf's built-in Tcl extensions.

Course Objective

Upon completion of this course, attendees should be able to write Tcl code to be used in conjunction with the Cloverleaf User Points of Control (UPoCs) provided with the engine to extend the engine's capabilities.

The final day of class is a half day certification exam using the skills you have learned during this course. The exam includes configuring a Cloverleaf network, monitoring message flow within the network, writing Tcl UPoCs, modifying and explaining engine output, use of database routines, etc. Upon completion of this exam, attendees may be certified as Cloverleaf Analysts*.

**Cloverleaf Analyst is the second level of certification for Cloverleaf.*

Who Should Attend?

Programmer Analysts responsible for implementing the Cloverleaf interface engine.

Topics

- Introduction to the internals of Cloverleaf. The review of engine internals will concentrate on message flow inside the product and how the user may control this flow from User Points of Control (UPoC). All UPoCs in Cloverleaf are written in Tcl, the Tool Command Language developed by Dr. John Outerhout.
- Extensive hands-on labs where you are given the opportunity to develop Tcl code, modify messages, impact message flow, and experiment with many facets of UPoCs.
- Because a primary focus of the Level 2 class is engine message flow, a complete architectural review is presented and used as the focus of the class. The review includes details about:
 - use of threads, including scheduling, context switching, CPU yields, etc.,
 - message flow between threads and processes,
 - recovery and error database states and the recovery process,
 - buffer and message management within the engine and Tcl,
 - inter thread queuing and replies, and
 - protocol drivers, protocol-level replies, application-level replies, etc.
- Cloverleaf's extensions to Tcl will be covered in detail. Examples of non-Cloverleaf additions include Extended Tcl (TclX) written by NeoSoft and the Tcl Distributed Programming (Tcl-DP) package written by Cornell University and the University of California at Berkeley. Cloverleaf extensions covered in detail include:
 - Message object intrinsics that allow users of INFOR's Tcl to work with binary data and to interact with messages flowing through the engine. Included in the intrinsics are conversions to and from binary representation and the ability to access and modify message metadata.

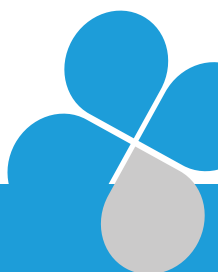


- Message flow "switches" and queue management intrinsics. These give the user extensive control over how and when messages move through the Cloverleaf engine.
 - Date conversion. These intrinsics allow users of INFOR's Tcl to convert between a large number of date and time formats.
 - Translation Pseudo Machine (XPM) intrinsics. XPM is responsible for performing translations within Cloverleaf. User's of INFOR's Tcl may use XPM to perform translations and other actions.
 - HL7, flat and variable record parsing intrinsics. These extensions are available to users of INFOR's Tcl so they may parse, modify, build, or replace complete messages, fields, or sub-fields as messages move through UPOCs or externally from the engine.
 - Local machine monitoring facilities, including CPU load, file system information, etc. A user of INFOR's Tcl may use these extensions to obtain information about the local machine's resources.
- Creation, modification, and understanding of the low-level file formats used by Cloverleaf. Included are fixed, variable, translation and network configuration files.
 - Command line Cloverleaf administrative tools. These are important if you are required to do full or part time remote support of a machine running Cloverleaf.
 - In-depth study of the output generated by the Cloverleaf engine. Detailed examples using the output controls are included. Static and dynamic (message attribute-based) engine output configurations are discussed.

Prerequisites

- Certified Cloverleaf Technician (Cloverleaf:Developing Networks certified).
- At least two months experience using Cloverleaf after becoming a Cloverleaf Technician or at least three months total.
- Programming background with an understanding of basic programming terms, constructs, and concepts.
- Good understanding of Tcl programming concepts and basic Tcl coding before attending the class. To assess your prerequisite skill level, please refer to the [Tcl Prerequisite Qualification](#).
- Knowledge of your operating system, UNIX or NT.
- You should be familiar with the following operating system commands and concepts:

Windows Requirements	UNIX Requirements
identify your current working directory	<i>pwd</i> command for identifying your current working directory
<i>more</i> or <i>type</i> commands for viewing a file	<i>more</i> , <i>pg</i> , or <i>cat</i> commands for viewing a file
<i>copy</i> command for copying files	<i>cp</i> command for copying files
<i>move</i> command for moving files	<i>mv</i> command for moving files
<i>del</i> command for removing files	<i>rm</i> or <i>del</i> command for removing files
<i>dir</i> command for listing the contents of a directory and associated options for formatting the list (i.e. <i>/p</i> for paging the output)	<i>ls</i> command for listing the contents of a directory and associated options for formatting the list (i.e. <i>-l</i> for a detailed listing)



<i>cd</i> command for changing directories	<i>cd</i> command for changing directory locations
<i>Edit</i> command, <i>Notepad</i> , or equivalent editor	<i>vi</i> , <i>emacs</i> , or equivalent editor command for editing files
<i>md</i> command for creating a new directory	<i>mkdir</i> command for creating a new directory
absolute and relative path names	absolute and relative path names
current, parent, and user directories	current, parent, and home directories
file permissions	file permissions
wildcard usage with filenames	wildcard usage with filenames
using Windows Explorer	environment variables
managing processes including background processing using <i>Task Manager</i>	managing processes including background processing
navigating/using the <i>Start</i> menu	Korn shell
using online help	<i>man</i> command for viewing online help

WARNING!

There is no review of these topics in class. Therefore, it is highly recommended that you be comfortable with these topics prior to attending class. Lack of knowledge in these areas results in unsatisfactory comprehension of the materials presented.

*To assess your prerequisite skill level, please refer to the [Operating System Prerequisite Qualification](#).

Suggested Preparation for Class

- Before joining the class, you should study the Tcl part of **Tcl and Tk Practical Programming** by Brent Welch —a book/CD presented in a tutorial fashion, intended for both novices and experts. Please review this book prior to attendance.
- Review the **comp.lang.tcl Tcl Frequently Asked Questions (FAQ)** list. You may obtain a copy of the current FAQ from INFOR or via anonymous ftp on the internet from the host **ftp.aud.alcatel.com**. Look in the **/tcl/docs** directory for the files named **tcl-faq.part[01-05].gz**.

Course Details

- Duration: 5 days.
- *Class Times:* 9:00 a.m.- 4:30 p.m. Monday-Thursday, 8:00 a.m.-2:00 p.m. Friday.

