

Designing Server Platforms for Windows Server 2019 Using C/C++

Installer, Service Process, Pipelines, Networking, Config, PerfMon, WMI, EIF, Patterns, HA/HT, Project

Server platforms consist of a mixture of multiple processes and threads, working in a co-ordinated manner, to provide some service to numerous clients on remote machines. These platforms must be flexible, extensible, configurable, scalable and controllable. A pipeline architecture allows extensible processing of messages. Many techniques are available for flexible inter-process communication. Service processes are the best way to deliver long-lived non-GUI functionality.

If your team consists of senior developers experienced with Windows and C/C++ and your team is assigned the task of developing a high-quality server platform on Windows Server 2019, then this is the ideal course to get all team members up to speed on what is needed.

It covers design concepts, important Windows C APIs, plenty of code samples and a chance to have architectural questions answered. It explores extra features (such as ETW, clean installer, PerfMon) that will distinguish your team's platform from the competition.

Building server platforms for Windows Server 2019 is the logical choice for future-oriented projects.

Contents of One-Day Training Course	
<p>Target Audience System architects and senior software engineers who need to create advanced server platforms for Windows Server 2019 using C/C++.</p> <p>Prerequisites General Windows system programming and especially multithreading experience.</p>	<div style="display: flex; justify-content: space-between;"> <div style="width: 48%;"> <p>Designing Server Platforms Multiple Processes/Threads Central Service (Manager) / Worker Processes / GUI+CLI Admin Processes Variety of threading architectures</p> <p>Service Process SCM-service code interaction Install+config of service “Log on as a service” security right Controlling worker processes</p> <p>Platform Installation How best to install server apps Installer formats and server extensions</p> <p>Platform Configuration Rich config choices Web garden/web-farm layout Config changes without restarting</p> <p>Pipelines Processing paths for messages Sequential and non-sequential steps Structure of pipeline (handlers&modules) Pipeline context</p> <p>Dynamically loading DLLs Dynamically loading DLLs Updating a server's DLLs without having to restart it</p> <p>Networking High performance sockets design Eliminating buffer copying Async I/O & Scatter/Gather I/O</p> <p>Use of Http Server API Same http.sys kernel service as used by IIS</p> </div> <div style="width: 48%;"> <p>Performance Monitor Detecting bottlenecks Tuning performance Developing for Performance Counter</p> <p>WMI Windows Management Instrumentation Management classes Exposing your server through WMI Management by GUI and CLI</p> <p>Event Tracing For Windows (ETW) Tracing architecture Event tracing APIs</p> <p>Designing Platform Security Leveraging Windows' security features Defense in depth platform security Secure communication with remote clients</p> <p>Design Patterns for Server Platforms Patterns to satisfy competing demands Pooling, tuning, managing Sharing, distributing Extending, scheduling</p> <p>High Availability / High Throughput Hardware for high-availability/throughput ccNUMA, Interconnects SAN, DAS, NAS Clustering concepts Design for high availability/throughput</p> <p>Project Overview of development of a sample server platform for Windows Server 2019</p> </div> </div>