***Tool xem file: C:\Program Files\Microsoft SDKs\Windows\v7.0A\bin***

***E2E tracing***

Acentral feature for monitoring WCF applications is called *end-to-end (E2E)*

*tracing*.

Using E2E tracing, it is possible to follow a sequence of actions across service and

machine boundaries—for example, from request origination on the client

through the business logic invoked by the target service.





End-to-end tracing is the concept where logically related actions in different

areas of applications, and perhaps on different systems altogether,

can be linked to improve our ability to follow specific scenarios through

logged information. This correlation is performed by passing unique identifiers

within and between endpoints of a WCF system.



**Tracing**

**Tracing** is used to record the flow and individual actions of the various components of a distributed application.

**Message**

Another feature**, message logging**, is used to record the contents of the messages from or to clients and services.

Message logging can be configured to capture messages at the service level, the transport level, and to record messages that are malformed. The data captured via message logging can be useful for a variety of situations, from diagnostics to creating audit trails of service utilization.

**Service Level**

Messages logged at this layer are about to enter (on receiving) or leave (on sending) user code. If filters have been defined, only messages that match the filters are logged. Otherwise, all messages at the service level are logged. Infrastructure messages (transactions, peer channel, and security) are also logged at this level, except for Reliable Messaging messages. On streamed messages, only the headers are logged. In addition, secure messages are logged decrypted at this level.

**Transport Level**

Messages logged at this layer are ready to be encoded or decoded for or after transportation on the wire. If filters have been defined, only messages that match the filters are logged. Otherwise, all messages at the transport layer are logged. All infrastructure messages are logged at this layer, including reliable messaging messages. On streamed messages, only the headers are logged. In addition, secure messages are logged as encrypted at this level, except if a secure transport such as HTTPS is used.

**Malformed Level**

Malformed messages are messages that are rejected by the WCF stack at any stage of processing. Malformed messages are logged as-is: encrypted if they are so, with non-proper XML, and so on. **maxSizeOfMessageToLog** defined the size of the message to be logged as CDATA. By default, **maxSizeOfMessageToLog** is equal to 256K. For more information about this attribute, see the Other Options section.



**Performance Counters**





**Windows Management Instrumentation (WMI)**

WCF supports the capability to expose settings and status via Windows Management Instrumentation, or WMI.





Filters also provide a safety feature using the **nodeQuota** attribute, which limits the maximum number of nodes in the XPath DOM that can be examined to match the filter.

<messageLogging logEntireMessage="true"

logMalformedMessages="true"

logMessagesAtServiceLevel="true"

logMessagesAtTransportLevel="true"

maxMessagesToLog="420">

<filters>

<add **nodeQuota="10"** xmlns:soap="http://www.w3.org/2003/05/soap-envelope">

/soap:Envelope/soap:Header

</add>

</filters>

</messageLogging>

Filters cannot be applied to the body of a message. Filters that attempt to manipulate the body of a message are removed from the list of filters. An event is also emitted that indicates this. For example, the following filter would be removed from the filter table.

<add xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">/s:Envelope/s:Body[contains(text(), "Hello")]</add>