kepware® kepserverex®

© 2018，PTC Inc. 保留所有权利。

目录

目录 2
14
KEPServerEX V6 14
简介 15
系统要求 16
组件 16
进程模式 17
接口和连接 18
OPC DA 18
OPC AE 18
OPC UA 19
OPC .NET 20
DDE 20
FastDDE/SuiteLink 21
iFIX 本机接口 21
ThingWorx 本机接口 21
访问管理菜单 23
设置 24
设置 - 管理 24
设置 - 配置 24
设置 - 运行时进程 25
设置 - 运行时选项 26
设置 - 事件日志 27
设置 - ProgID 重定向 29
设置 - 用户管理器 30
设置 - 用户管理器 ThingWorx Interface 用户 34
设置 - 配置 API 服务配置 35
设置 - 证书存储 37
设置 - 服务端口 39
用户界面导航 40
项目属性 43
项目属性 - 常规 43
项目属性 - OPC UA 45
项目属性 - DDE 46
项目属性 - OPC .NET 47
项目属性 - OPC AE 48
项目属性 - FastDDE/SuiteLink 48
项目属性 - iFIX PDB 设置 49
项目属性 - OPC HDA 50

项目属性 - ThingWorx 51
ThingWorx 示例 - 传统模式 55
存储并转发 - 完成率示例 57
存储并转发 - 系统标记 57
服务器选项 60
选项 - 常规 60
选项 - 运行时连接 61
基本组件 62
什么是信道？ 62
通道属性 - 常规 62
通道属性 - 高级 63
通道属性 - 以太网通信 64
通道属性 - 串行通信 64
通道属性 - 以太网封装 66
通道属性 - 通信序列化 66
通道属性 - 网络接口 67
通道属性 - 写入优化 68
?????? 68
什么是设备？ 69
设备属性 - 常规 69
设备属性 - 扫描模式 71
设备属性 - 自动降级 71
设备属性 - 通信参数 72
设备属性 - 以太网封装 72
设备属性 - 标记生成 73
设备属性 - 时间同步 74
设备属性 - 定时 75
设备属性 - 冗余 75
什么是标记？ 76
标记属性 - 常规 76
多标记生成 79
标记属性 - 缩放 82
动态标记 83
静态标记 (用户定义) 84
什么是标记组？ 84
标记组属性 84
什么是别名映射？ 85
别名属性 85
什么是事件日志？ 86
事件日志 86
标记管理 88
CSV 导入和导出 88
系统标记 90

属性标记 100
统计信息标记 101
调制解调器标记 103
通信序列化标记 105
通信管理 107
在服务器项目中使用调制解调器 107
电话簿 108
自动拨号 109
设计项目 111
运行服务器 111
开始新建项目 111
添加和配置信道 112
信道创建向导 112
添加和配置设备 114
设备创建向导 115
添加用户定义标记 (示例) 116
浏览标记 118
生成多个标记 119
添加标记缩放 121
保存项目 122
打开加密的项目 124
测试项目 124
如何... 130
允许桌面交互作用 130
创建和使用别名 131
优化服务器项目 133
正确命名通道、设备、标记和标记组 134
连接到服务器的 DNS/DHCP 设备重新上电时，解决通信问题 134
使用别名来优化项目 135
在服务器上使用 DDE 135
使用动态标记寻址 136
使用以太网封装 136
设备需求轮询 138
配置 API 服务 139
安全性 139
文档 139
配置 API 服务 - 体系结构 139
配置 API 服务 - 并发客户端 140
配置 API 服务 - 内容检索 140
配置 API 服务 - 数据 145
配置 API 服务 - 服务 148

配置 API 服务 - 响应码 150
iFIX 信号调节选项 151
iFIX 应用程序的项目启动 156
存储并转发服务 157
内置诊断 158
OPC 诊断查看器 158
OPC DA 事件 161
OPC UA 服务 166
通信诊断 168
事件日志消息 171
服务器汇总信息 171
未找到 <名称> 设备驱动程序，或者无法加载。 172
无法加载 '<名称>' 驱动程序，因为存在多个副本 ('<名称>' 和 '<name>')。请移除冲突的驱动程序并重新启动应用程序。 172
项目文件无效。 173
未能打开调制解调器线路 '<line>' [TAPI 错误 = <code>]。 173
由于驱动程序级别故障，无法添加通道。 173
由于驱动程序级别故障，无法添加设备。 173
版本不匹配。 173
无效的 XML 文档: 173
无法加载项目 <名称>: 174
无法将项目备份到 '<路径>' [<原因>]。保存操作已中止。请验证目标文件未锁定并且具有读/写权限。要继续保存此项目而不进行备份，请在“工具”|“选项”|“常规”下取消选择备份选项，然后重新保存项目。 174
未找到 <feature name>，或者无法加载。 174
无法保存项目文件 <名称>: 174
设备发现超出最大允许设备数 <count>。请限制发现范围并重试。 174
加载此项目需要 <feature name>。 175
当前语言不支持加载 XML 项目。要加载 XML 项目，请在“服务器管理”中将产品语言选择更改为英语。 175
由于缺少对象，无法加载项目。| 对象 = '<object>'。 175
加载项目时遇到无效的型号。| 设备 = '<设备>'。 175
无法添加设备。此通道中可能存在重复设备。 175
自动生成的标记 '<tag>' 已存在，无法覆盖。 175
无法为设备 '<device>' 生成标记数据库。设备未响应。 175
无法为设备 '<device>' 生成标记数据库: 176
自动生成造成的覆盖太多，已停止发布错误消息。 176
未能添加标记 '<tag>'，因为地址太长。最大地址长度为 <number>。 176
线路 '<line>' 已使用。 176
线路 '<line>' 存在硬件错误。 176
未对线路 '<line>' 的连接提供通信处理。 177
无法在线路 '<line>' 拨号。 177
无法使用通道 '<name>' 上的网络适配器 '<adapter>'。正在使用默认网络适配器。 177
系统正在拒绝对引用设备 '<channel device>' 上的型号类型的更改。 177

TAPI 线路初始化失败: <code>。 177
验证错误位于 '<tag>': <error>。 178
无法加载驱动程序 DLL '<名称>'。 178
验证错误位于 '<tag>': 缩放参数无效。 178
无法对线路 '<line>' 应用调制解调器配置。 178
设备 '<device>' 已自动降级。 178
<Source>: 无效的以太网封装 IP '<address>'。 179
'<product>' 驱动程序当前不支持 XML 持久存储。请使用默认文件格式进行保存。 179
无法加载插件 DLL '<名称>'。 179
为 '<device>' 设备设置的时区是 '<zone>'。对于系统而言，这是无效的时区。正在将时区更改为默认时区 '<zone>'。 180
无法加载驱动程序 DLL '<名称>'。原因: 180
无法加载插件 DLL '<名称>'。原因: 180
电话簿必须至少有一个号码，通道才能自动拨号。| 通道 = '<channel>'。 180
通道需要启用自动拨号和其电话簿中的至少一个号码，以使用共享调制解调器连接。| 通道 = '<通道>'。 180
指定的网络适配器在通道 '%1' 上无效。| 适配器 = '%2'。 181
标记生成请求未创建标记。有关详细信息，请参阅事件日志。 181
TAPI 配置已更改，正在重新初始化... 181
<Product> 设备驱动程序加载成功。 181
正在启动 <名称> 设备驱动程序。 181
正在停止 <名称> 设备驱动程序。 181
正在线路 '<modem>' 拨号 '<number>'。 181
线路 '<modem>' 已断开连接。 181
线路 '<modem>' 的拨号已被用户取消。 182
线路 '<modem>' 已连接，速率 <rate> 波特。 182
远端线路在 '<modem>' 占线。 182
远端线路在 '<modem>' 未应答。 182
'<modem>' 无拨号音。 182
电话号码无效 (<number>)。 182
'<modem>' 的拨号已中止。 182
线路在远程站点的 '<modem>' 断开。 182
线路 '<modem>' 检测到呼入。 182
调制解调器线路开通: '<modem>'。 182
调制解调器线路关闭: '<modem>'。 182
<Product> 设备驱动程序已从内存卸载。 182
线路 '<modem>' 已连接。 183
设备 '<device>' 已启用模拟模式。 183
设备 '<device>' 已禁用模拟模式。 183
正在尝试对设备 '<device>' 自动生成标记。 183
已完成设备 '<device>' 的自动标记生成。 183
正在开始对调制解调器线路 '<modem>' 断开连接。 183
客户端应用程序已对设备 '<device>' 启用自动降级。 183
设备 '<device>' 已启用数据收集。 183
设备 '<device>' 已禁用数据收集。 183

项目不允许对象类型 '<name>'。 183
已将项目 '<名称>' 的备份创建到 '<路径>'。 183
已自动升级设备 '<device>' 以确定是否可以重新建立通信。 184
未能加载库: <名称>。 184
未能读取构建清单资源: <名称>。 184
项目文件由此软件的最新版本创建。 184
客户端应用程序已对设备 '<device>' 禁用自动降级。 184
电话号码优先级已更改。| 电话号码名称 = '<名称>', 更新的优先级 = '<priority>'。 184
设备 '<设备>' 的标记生成结果。| 创建的标记 = <计数>。 184
设备 '<设备>' 的标记生成结果。| 创建的标记 = <计数>, 覆盖的标记 = <计数>。 184
设备 '<设备>' 的标记生成结果。| 创建的标记 = <计数>, 未覆盖的标记 = <计数>。 184
访问对象被拒绝。| 用户 = '<account>', 对象 = '<object path>', 权限 = 184
正在更改运行时操作模式。 184
更改运行时操作模式完成。 185
关闭以执行安装。 185
OPC ProgID 已添加到 ProgID 重定向列表。| ProgID = '<ID>'。 185
OPC ProgID 已从 ProgID 重定向列表移除。| ProgID = '<ID>'。 185
无效的 ProgID 条目已从 ProgID 重定向列表删除。| ProgID = '<ID>'。 185
管理员密码已由当前用户重置。| 管理员名称 = '<名称>', 当前用户 = '<名称>'。 185
用户已从用户组转移。| 用户 = '<名称>', 旧组 = '<名称>', 新组 '<name>'。 185
用户组已创建。| 组 = '<名称>'。 185
用户已添加到用户组。| 用户 = '<名称>', 组 = '<名称>'。 185
用户信息已替换为导入内容。| 导入的文件 = '<absolute file path>'。 185
用户组已重命名。| 旧名称 = '<名称>', 新名称 = '<名称>'。 185
用户组的权限定义已更改。| 组 = '<名称>'。 185
用户已重命名。| 旧名称 = '<名称>', 新名称 = '<名称>'。 186
用户已禁用。| 用户 = '<名称>'。 186
用户组已禁用。| 组 = '<名称>'。 186
用户已启用。| 用户 = '<名称>'。 186
用户组已启用。| 组 = '<名称>'。 186
未能重置管理员密码。| 管理员名称 = '<名称>'。 186
重置管理员密码失败。当前用户不是 Windows 管理员。| 管理员名称 = '<名称>', 当前用户 = '<名称>'。 186
用户密码已更改。| 用户 = '<名称>'。 186
导入 CSV 标记时出现一般故障。 186
尝试连接到运行时失败。| 运行时主机地址 = '<host address>', 用户 = '<name>', 原因 = '<reason>'。 186
用户信息无效或缺失。 186
用户权限不足以替换运行时项目。 186
运行时项目更新失败。 187
未能检索运行时项目。 187
无法替换通道上的设备，因为该设备有活动的引用计数。| 通道 = '<名称>'。 187
未能替换通道上的现有自动生成设备，删除失败。| 通道 = '<名称>'。 187
通道不再有效，可能在等待用户输入时被移除。| 通道 = '<名称>'。 187
未加载设备驱动程序 DLL。 187

未找到或无法加载设备驱动程序。| 驱动程序 = '<名称>'。 187
导入 CSV 数据时出错。\n\n读取标识记录时字段缓冲区溢出。 187
导入 CSV 数据时出错。\n\n无法识别字段名称。| 字段 = '<名称>'。 187
导入 CSV 数据时出错。\n\n字段名称重复。| 字段 = '<名称>'。 187
导入 CSV 数据时出错。\n\n字段标识记录缺失。 187
导入 CSV 记录时出错。\n\n字段缓冲区溢出。| 记录索引 = '<number>'。 187
导入 CSV 记录时出错。\n\n插入失败。| 记录索引 = '<number>', 记录名称 = '<name>'。 188
无法启动应用程序。| 应用程序 = '<路径>', 操作系统错误 = '<code>'。 188
导入 CSV 记录时出错。\n\n“映射到”标记地址对于此项目无效。| 记录索引 = '<number>', 标记地址 = '<address>'。 188
导入 CSV 记录时出错。\n\n别名无效。名称不能包含双引号或以下划线开头。| 记录索引 = '<number>'。 188
无效的 XML 文档: 188
重命名失败。已存在具有该名称的对象。| 建议名称 = '<名称>'。 188
未能启动通道诊断。 188
重命名失败。名称不能包含句点、双引号或以下划线开头。| 建议名称 = '<名称>'。 188
与远程运行时同步失败。 188
导入 CSV 记录时出错。标记名称无效。| 记录索引 = '<number>', 标记名称 = '<name>'。 188
导出 CSV 记录时出错。标记或组名称超出最大名称长度。| 记录索引 = '<number>', 最大名称长度 (字符数) = '<number>'。 188
导入 CSV 记录时出错。地址缺失。| 记录索引 = '<number>'。 189
导入 CSV 记录时出错。标记组名称无效。| 记录索引 = '<index>', 组名称 = '<name>'。 189
由于存在启用的连接，已忽略关闭请求。| 启用的连接 = '<count>'。 189
未能保存嵌入的相关文件。| 文件 = '<路径>'。 189
配置实用程序不能与第三方配置应用程序同时运行。请关闭这两个程序，然后只打开要使用的程序。| 产品 = '<名称>'。 189
正在打开项目。| 项目 = '<名称>'。 189
正在关闭项目。| 项目 = '<名称>'。 189
虚拟网络模式已更改。这会影响所有通道和虚拟网络。有关虚拟网络模式的详细信息，请参阅帮助。| 新模式 = '<mode>'。 189
正在开始对通道进行设备发现。| 通道 = '<名称>'。 189
通道的设备发现已完成。| 通道 = '<名称>', 发现的设备 = '<计数>'。 189
通道的设备发现已取消。| 通道 = '<名称>'。 189
通道的设备发现已取消。| 通道 = '<名称>', 发现的设备 = '<计数>'。 190
无法开始对通道进行设备发现。| 通道 = '<名称>'。 190
正在关闭以执行安装。 190
运行时项目已重置。 190
运行时项目已替换。| 新项目 = '<路径>'。 190
未连接到事件记录器服务。 190
尝试添加项 '<名称>' 失败。 190
未加载设备驱动程序 DLL。 190
无效的项目文件: '<名称>'。 190
无法打开项目文件: '<名称>'。 190
替换项目的请求被拒绝，因为它与正在使用的项目相同: '<名称>'。 190
文件名不能覆盖现有文件: '<name>'。 190
文件名不能为空白。 191

文件名的格式必须是 <subdir>/<name>.{json,opf,sopf} 191
文件名包含一个或多个无效字符。 191
在启用项目文件加密的情况下保存项目文件，因为不支持 .OPF 文件类型。支持的文件类型为 .SOPF 和 .JSON。 191
在禁用项目文件加密的情况下保存项目文件，因为不支持 .SOPF 文件类型。支持的文件类型为 .OPF 和 .JSON。 191
添加对象到 '<名称>' 失败: <原因>。 191
移动对象 '<名称>' 失败: <原因>。 191
更新对象 '<名称>' 失败: <原因>。 191
删除对象 '<名称>' 失败: <原因>。 191
无法加载启动项目 '<名称>': <原因>。 191
未能更新启动项目 '<名称>': <原因>。 191
运行时项目已替换为定义的启动项目。运行时项目将在下次重新启动时从 '<名称>' 恢复。 191
正在忽略用户定义的启动项目，因为配置会话处于活动状态。 192
读取只读项引用 '<名称>' 请求被拒绝。 192
无法写入项 '<名称>'。 192
对项 '<名称>' 的写入请求失败。写入数据类型 '<type>' 无法转换为标记数据类型 '<type>'。 192
对项 '<名称>' 的写入请求失败。缩放写入数据时出错。 192
对项引用 '<名称>' 的写入请求被拒绝，因为其所属的设备被禁用。 192
已成功配置 <名称> 作为系统服务运行。 192
已成功从服务控制管理器数据库移除 <名称>。 192
运行时重新初始化已启动。 192
运行时重新初始化已完成。 192
已更新启动项目 '<名称>'。 192
运行时服务已启动。 192
运行时进程已启动。 193
运行时正在退出。 193
运行时关闭完成。 193
关闭以执行安装。 193
运行时项目从 '<名称>' 替换。 193
应用程序数据目录缺失。 193
运行时项目已另存为 '<名称>'。 193
运行时项目已替换。 193
配置会话由 <名称> (<名称>) 启动。 193
分配到 <名称> 的配置会话已结束。 193
分配到 <名称> 的配置会话已升级为写入权限。 193
分配到 <名称> 的配置会话已降级为只读。 193
应用于配置会话的权限更改已分配到 <名称>。 194
Failed to start Script Engine server. Socket error occurred binding to local port. | Error = <error>, Details = '<information>'. 194
An unhandled exception was thrown from the script. | Function = '<function>', error = '<error>'. 194
Script Engine service stopping. 194
Script Engine service starting. 194
Config API SSL 证书包含错误签名。 194
Config API 无法加载 SSL 证书。 194

无法启动 Config API 服务。绑定到端口时可能出现了问题。 194
Config API SSL 证书已过期。 195
Config API SSL 证书是自签名证书。 195
Configuration API started without SSL on port <port number>. 195
Configuration API started with SSL on port <port number>. 195
OPC .NET 服务器无法启动。有关详细信息，请查看 Windows 应用程序事件日志。另请确保 .NET 3.5 Framework 已安装。| OS 错误 = '<错误原因>'。 195
OPC .NET 服务器无法启动，因为尚未安装。请重新运行安装程序。 195
尝试启动 OPC .NET 服务器超时。请验证服务器是否正在通过使用 OPC .NET Configuration Manager 运行。 195
缺少服务器实例证书 '<证书位置>'。请使用 OPC UA Configuration Manager 重新颁发证书。 195
无法导入服务器实例证书: '<证书位置>'。请使用 OPC UA Configuration Manager 重新颁发证书。 195
UA 服务器证书过期。请使用 OPC UA Configuration Manager 重新颁发证书。 195
侦听客户端连接时发生套接字错误。| 端点 URL = '<端点 URL>'，错误 = <错误代码>，详细信息 = '<说明>'。 195
UA 服务器无法注册到 UA Discovery Server。| 端点 URL: '<端点 url>'。 196
UA 服务器无法从 UA Discovery Server 取消注册。| 端点 URL: '<端点 url>'。 196
UA 服务器成功注册到 UA Discovery Server。| 端点 URL: '<端点 url>'。 196
UA 服务器成功从 UA Discovery Server 取消注册。| 端点 URL: '<端点 url>'。 196
无法启用对此服务器的 iFIX PDB 支持。| OS 错误 = '<错误>'。 196
无法启用对此服务器的 iFIX PDB 支持。| OS 错误 = '<错误>'。 196
ReadProcessed 请求超时。| 运行时间 = <秒> (秒)。 196
ReadAtTime 请求超时。| 运行时间 = <秒> (秒)。 196
尝试添加 DDE 项失败。| 项 = '<项名称>'。 196
DDE 客户端尝试添加主题失败。| 主题 = '<主题>'。 196
无法写入项。| 项 = '<项名称>'。 197
指定的区域无效。无法设置订阅筛选器。| 区域 = '<区域名称>'。 197
指定的源无效。无法设置订阅筛选器。| 源 = '<源名称>'。 197
Config API SSL 证书包含错误签名。 197
Config API 无法加载 SSL 证书。 197
无法启动 Config API 服务。绑定到端口时可能出现了问题。 197
Config API SSL 证书已过期。 197
Config API SSL 证书是自签名证书。 197
Configuration API started without SSL on port <port number>. 197
Configuration API started with SSL on port <port number>. 197
与 ThingWorx 连接失败。| 平台 <主机:端口资源>，错误: <原因>。 197
添加项时出错。| 项名称 = '<项名称>'。 198
无法在平台上触发自动绑定完成事件。 198
与 ThingWorx 连接失败，原因未知。| 平台 = <主机:端口资源>，错误= <错误>。 198
因连接缓冲区中空间不足，一或多个值更改更新丢失。| 丢失更新数 = <计数>。 198
发布项失败。多维数组不受支持。| 项名称 = '%s'。 199
由于磁盘已满，“存储并转发”数据存储无法存储数据。 199
已达到“存储并转发”数据存储大小限制。 199
与 ThingWorx 的连接已关闭。| 平台 = <主机:端口资源>。 199
无法自动绑定属性。| 名称 = '<属性名称>'。 200

无法重启“事物”。| 名称 = '<事物名称>'。 200
写入属性失败。| 属性名称 = '<名称>'，原因 = <原因>。 200
ThingWorx 请求添加项失败。该项已添加。| 项名称 = '<名称>'。 200
ThingWorx 请求移除项失败。该项不存在。| 项名称 = '<名称>'。 201
已将服务器配置为针对每次扫描发送更新，但将一个或多个属性的推送类型设置为仅在值更改时推送。| 计数 = <计数>。 201
将一个或多个属性的推送类型设置为从不向平台推送更新。| 计数 = <计数>。 201
ThingWorx 请求移除项失败。该项已绑定且强制标志为 false。| 项名称 = '<名称>'。 201
写入属性失败。| 事物名称 = '<名称>'，属性名称 = '<名称>'，原因 = <原因>。 201
将属性更新推送到事物时出错。| 事物名称 = '<名称>'。 202
无法连接或附加到“存储并转发”数据存储。正在使用内存存储。| 内存存储大小 (更新数) = <计数>。 202
由于文件 IO 错误或数据存储损坏，“存储并转发”数据存储已重置。 202
无法应用由 ThingWorx Platform 引起的设置更改。权限被拒绝。| 用户 = '<用户名>'。 202
面向 ThingWorx Platform 的配置传输失败。 203
面向 ThingWorx Platform 的配置传输失败。| 原因 = '<原因>' 203
无法删除“存储并转发”数据存储中已存储的更新。 203
来自 ThingWorx Platform 的配置传输失败。 203
来自 ThingWorx Platform 的配置传输失败。| 原因 = '<原因>' 203
已连接至 ThingWorx。| 平台 = <主机:端口资源>，事物名称 = '<名称>'。 203
因平台引起了项目设置更改而正在重新初始化 ThingWorx 连接。 204
因接口关闭或重新初始化而正在放弃待处理的自动绑定。| 计数 = <计数>。 204
已处理一个或多个自动绑定请求。| 计数 = <计数>。 204
因 Configuration API 引起了项目设置更改而正在重新初始化 ThingWorx 连接。 204
已恢复将属性更新推送到事物: 错误条件已解决。| 事物名称 = '<名称>'。 204
来自 ThingWorx 的配置传输已启动。 204
来自 ThingWorx 的配置传输已中止。 204
已初始化“存储并转发”数据存储。| 数据存储位置: '<位置>'。 204
已成功从“存储并转发”数据存储中删除已存储的数据。 205
“存储并转发”模式已更改。| 转发模式 = '<模式>'。 205
已初始化“存储并转发”数据存储。| 转发模式 = '<模式>' | 数据存储位置 = '<位置>'。 205
由于数据存储路径无效，连接到数据存储时出错。| 路径 = '<路径>'。 205
无法启动“存储并转发”服务器。绑定到本地端口时出现 socket 错误。| 错误 = <错误>, 详细信息 = '<信息>'。 205
“存储并转发”服务正在停止。 205
“存储并转发”服务正在启动。 205
连接数据存储时文件损坏，已重新创建数据存储。| 数据存储路径 = '<路径>'。 206
配置更改导致数据存储被覆盖。| 数据存储路径 = '<路径>'。 206
无法连接到现有的数据存储，因为该数据存储由较早版本的服务器创建。已重新创建数据存储。| 数据存储路径 = '<路径>'。 206
由于数据存储路径无效，连接到数据存储时出错。| 路径 = '<路径>'。 206
无法启动“存储并转发”服务器。绑定到本地端口时出现 socket 错误。| 错误 = <错误>, 详细信息 = '<信息>'。 206
“存储并转发”服务正在停止。 207
“存储并转发”服务正在启动。 207
连接数据存储时文件损坏，已重新创建数据存储。| 数据存储路径 = '<路径>'。 207

配置更改导致数据存储被覆盖。| 数据存储路径 = '<路径>'。 207
无法连接到现有的数据存储，因为该数据存储由较早版本的服务器创建。已重新创建数据存储。| 数据存储路径 = '<路径>'。 207
另一应用程序正在使用 Com 端口。| 端口 = '<端口>'。 207
无法使用指定参数配置 com 端口。| 端口 = COM<数字>，OS 错误 = <错误>。 208
驱动程序初始化失败。 208
无法创建串行 I/O 线程。 208
Com 端口不存在。| 端口 = '<端口>'。 208
打开 com 端口时出错。| 端口 = '<端口>', OS 错误 = <错误>。 208
连接失败。无法绑定到适配器。| 适配器 = '<名称>'。 208
Winsock 关闭失败。| OS 错误 = <错误>。 209
Winsock 初始化失败。| OS 错误 = <错误>。 209
要使用此驱动程序，必须安装 Winsock V1.1 或更高版本。 209
绑定到本地端口时发生套接字错误。| 错误 = <错误>，详细信息 = '<信息>'。 209
设备未响应。 209
设备未响应。| ID = '<设备>'。 210
通道上出现串行通信错误。| 错误掩码 = <掩码>。 210
写入标记 <设备名称>.<地址> 时检测到无效的数组大小。 210
无法写入设备上的地址。| 地址 = '<地址>'。 210
驱动程序在处理标记期间可能不会更改此页上的项。 211
设备上的指定地址无效。| 无效地址 = '<地址>'。 211
地址 '<地址>' 在设备 '<名称>' 上无效。 211
驱动程序在处理标记期间可能不会更改此属性。 211
无法写入设备 '<名称>' 上的地址 '<地址>'。 211
连接时发生套接字错误。| 错误 = <错误>，详细信息 = '<信息>'。 212
接收数据时发生套接字错误。| 错误 = <错误>，详细信息 = '<信息>'。 212
发送数据时发生套接字错误。| 错误 = <错误>，详细信息 = '<信息>'。 212
检查可读性时发生套接字错误。| 错误 = <错误>，详细信息 = '<信息>'。 212
检查可写性时发生套接字错误。| 错误 = <错误>，详细信息 = '<信息>'。 212
%s | 213
<名称> 设备驱动程序 '<名称>' 213
功能 '<名称>' 未获许可，因而无法使用。 213
无法加载许可证界面，可能是因为缺失第三方依赖项。仅在“限时”模式下运行。 213
“限时”模式已过期。 213
对于精简版 '<数字>' 许可证，已超出最大设备计数。编辑项目并重新启动服务器。 214
对于精简版 '<数字>' 许可证，已超出最大运行时标记计数。编辑客户端项目并重新启动服务器。 214
在功能 '<名称>' 上，已超出 <最大计数> 的类型 <数字类型 ID> 限制。 214
在功能 '<名称>' 上，已超出 <最大计数> 的 <对象类型名称> 限制。 215
必须启用 FlexNet 许可服务才能处理许可证。启用服务失败将导致“限时”模式启动。 215
<名称> 功能许可证已移除。服务器将进入“限时”模式，除非在宽限期过期之前已恢复许可证。 215
无法访问功能 <名称> 的许可证 [错误=<代码>]，必须重新将其激活。 216
功能 <名称> 存在时间限制，且将于 <日期/时间> 过期。 216
功能 <名称> 存在时间限制，且将于 <日期/时间> 过期。 216
在功能 <名称> 上，已超出对象计数限制。受时间限制的使用将于 <日期/时间> 过期。 216
在 <名称> 上，已超出特征计数限制。受时间限制的使用将于 <日期/时间> 过期。 216

在功能 <名称> 上，受时间限制的使用已过期。 216
对于精简版 '<名称>' 驱动程序许可证，已超出最大驱动程序计数。编辑项目并重新启动服务器。 216
无法添加项目。请求的 <数字> 计数会超过 <最大计数> 的许可证限制。 217
要求组件 <名称> (<版本>) 版本与组件 <名称> (<版本>) 版本相匹配。 217
对于精简版 '<名称>' 驱动程序许可证，已超出最大通道计数。编辑项目并重新启动服务器。 217
%s 此刻已获许可。 218
资源 219
索引 220

KEPServerEX V6

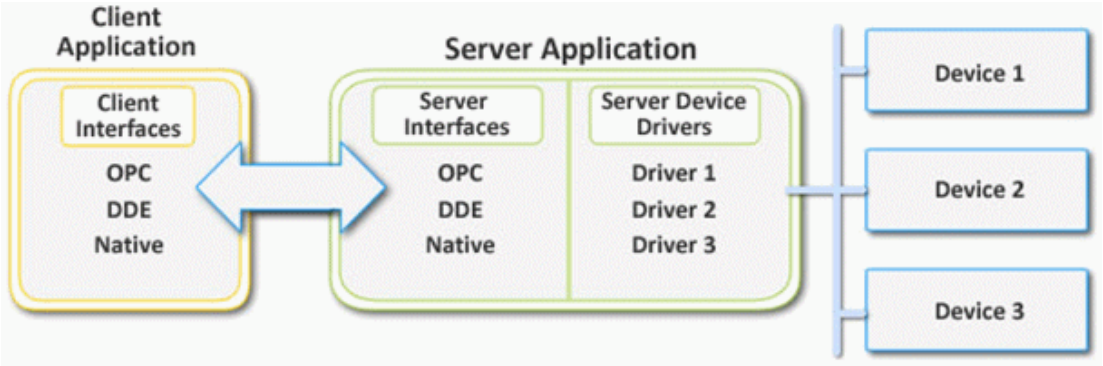
目录

简介
接口和连接
访问管理菜单
导航配置
基本服务器组件
标记管理
通信管理
内置诊断
设计项目
如何...?
事件日志消息

*有关产品许可的详细信息，请参阅 License Utility 帮助文件。要通过服务器"配置"菜单访问帮助文件，请单击**"帮助"| "服务器帮助"| License Utility**。要通过服务器"管理"菜单访问帮助文件，请右键单击"系统托盘"中的 KEPServerEX 图标，然后选择**"帮助"| License Utility**。*

简介

这种基于软件的服务器用于在客户端应用程序、工业设备和系统之间实现精确通信和快速设置，并提供卓越的互用性。该服务器提供了各种插件、设备驱动程序以及组件，可以满足大多数通信需求。插件设计和单一用户界面为基于标准的应用程序和基于非标准的应用程序都提供了本机接口，可实现一致性的访问。



系统要求

该服务器对于软件和硬件有最低系统要求。必须满足这些要求才能使应用程序如期运行。

此应用程序支持以下 Microsoft Windows 操作系统：

- Windows 10 x64 (专业版和企业版)[3]
- Windows 10 x86(专业版和企业版)
- Windows 8.1 x64 (Windows 8、专业版和企业版)[3]
- Windows 8.1 x86(Windows 8、专业版和企业版)
- Windows 8 x64 (Windows 8、专业版和企业版)[3]
- Windows 8 x86(Windows 8、专业版和企业版)
- Windows 7 x64 (专业版、旗舰版和企业版)[3]
- Windows 7 x86(专业版、旗舰版和企业版)
- Windows Server 2016 x64[3]
- Windows Server 2012 x64 R2[3]
- Windows Server 2012 x64[3]
- Windows Server 2008 x64 R2[3]

注意

1. 当安装于 64 位操作系统时，应用程序运行于 Windows 的子系统 WOW64 (Windows-on-Windows 64 位) 中。所有 Windows 64 位版本均包含 WOW64，它在对用户透明的操作系统中表现突出。WOW64 的最低需求如下：
   - 1 GHz 处理器
   - 安装 1 GB 的 RAM (遵从操作系统建议)
   - 180 MB 可用磁盘空间
   - 以太网卡
2. 验证是否已为操作系统安装最新安全更新。
3. 在 32 位兼容模式下运行。

有关较复杂系统的需求和建议的指南，请联系系统工程人员。

组件

服务器将实施客户端/服务器架构。组件包括“配置”、“运行时”、“管理”和“事件日志”。

配置

“配置”是用于修改运行时项目的客户端用户界面。“配置”可由多个用户启动，且支持远程运行时配置。

CSV 导入和导出

此服务器支持以逗号分隔变量 (CSV) 文件的形式导入和导出标记数据。使用 CSV 导入和导出时，将在所需应用程序中快速创建标记。

有关详细信息，请参阅 CSV 导入和导出。

运行时

“运行时”是默认情况下作为服务启动的服务器组件。客户端能够以远程或本地方式连接到“运行时”。

管理

“管理”用于查看和/或修改设置以及启动与用户管理和服务器相关的应用程序。默认情况下，用户帐户登录到操作系统时，将启动“管理”并将其发送到“系统托盘”。

项目

“项目”文件包含信道、设备和标记定义以及首选项和其他已保存设置。

有关详细信息，请参阅设计项目。

事件日志

“事件日志”服务用于收集信息、警告、错误和安全事件。这些事件将被发送到“配置”的“事件日志”窗口中以供查看。

有关详细信息，请参阅什么是事件日志？

另请参阅：基本服务器组件

进程模式

当服务器运行时，可以更改“运行时”进程模式；但是，在客户端连接时这样操作会在短时间内中断连接。操作模式为“系统服务”和“交互式”。

系统服务

默认情况下，已安装服务器并且作为服务运行。选择“系统服务”后，“运行时”不需要用户干预，并将在操作系统处在打开状态时启动。这允许独立用户通过客户端访问服务器。

交互式

选择“交互式”后，“运行时”将一直处于停止状态，直到客户端尝试与其连接。“运行时”启动后，将运行到所有客户端断开连接并关闭之后。如果用户账户从操作系统中注销，“运行时”也会关闭。

注意：可以通过“管理”设置对话框更改“运行时”进程模式，以满足客户端应用程序的需要。

以下条件下需要“系统服务”：

- 当需要在操作系统上运行 iFIX，且 UAC 处于启用状态时。

以下条件下需要“交互式”：

- 当通信界面 (如 DDE) 必须与用户桌面交换信息,而服务器安装在 Windows Vista、Windows Server 2008 或更高版本的操作系统上。

另请参阅：

设置 - 运行时进程

如何... 允许桌面交互作用

接口和连接

此通信服务器同时支持下面列出的客户端/服务器技术。客户端应用程序可以使用其中任何一种技术同时访问服务器中的数据。有关特定接口的详细信息，请从下表中选择一个链接。

OPC DA
OPC AE
OPC UA
OPC .NET
DDE
FastDDE/SuiteLink
iFIX 本机接口
ThingWorx 本机接口

OPC DA

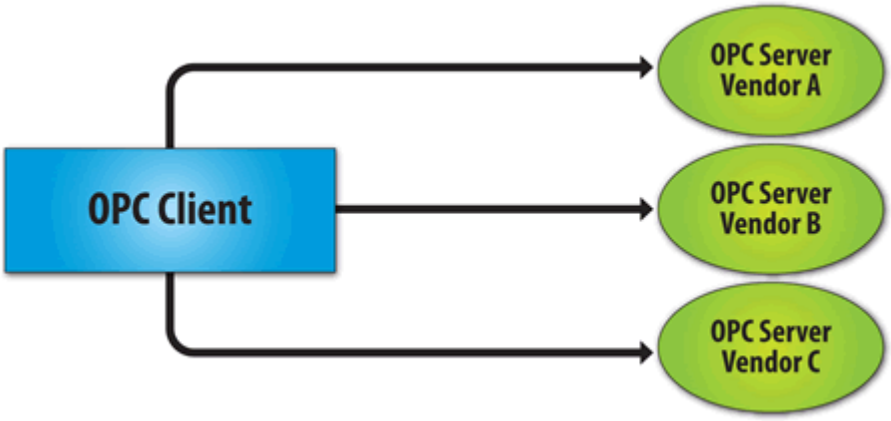
支持的版本

1.0a
2.05a
3.0

概述

"OPC" 代表工业自动化中的“开放生产力和连接”以及支持工业的企业系统。它是一种客户端/服务器技术，其中一个应用程序充当服务器 (提供数据)，另一个应用程序充当客户端 (使用数据)。

OPC 由一系列标准规范组成：OPC 数据访问 (DA) 是最为丰富的标准。OPC DA 是广为接受的工业通信标准，可在多供应商设备和控制应用程序之间进行数据交换，而无需专有限制。OPC 服务器可以在车间 PLC、现场 RTU、HMI 站和桌面 PC 上的软件应用程序之间进行连续的数据通信。OPC 合规性可以实现连续实时通信 (即使硬件和软件来自不同的供应商)。



OPC 数据访问 1.0a 是由 OPC Foundation 于 1996 年开发的原始规范。目前，大部分 OPC 客户端应用程序仍支持 OPC 数据访问 1.0 a，但其增强版 OPC 数据访问 2.0 能更好地利用基础 Microsoft COM 技术。OPC 数据访问 3.0 是 OPC DA 接口的最新版本。

另请参阅：项目属性 - OPC DA

OPC AE

支持的版本

1.0
1.10

概述

OPC 警报和事件是由 OPC Foundation 开发的规范，用于标准化在系统之间共享警报和事件信息的方式。使用该标准，AE 客户端可以接收有关设备安全限制、系统错误和其他异常情况的报警和事件通知。

"简单事件"
简单事件包括事件日志中所显示的服务器事件 (例如，信息、警告、错误和安全事件)。对于 AE 客户端的简单事件，服务器支持以下筛选选项：

- **"事件类型"**简单。
- **"事件类别"**根据服务器定义的类别进行筛选。每个事件都会被分配到一个类别。类别说明如下：
    - **"运行时错误事件"**在事件日志中显示为错误的简单事件。
    - **"运行时警告事件"**在事件日志中显示为警告的简单事件。
    - **"运行信息事件"**在事件日志中显示为信息的简单事件。

"条件事件"
条件事件由服务器条件创建，当前只能通过使用警报和事件插件进行配置。对于 AE 客户端的条件事件，服务器支持以下筛选选项：

1. **"事件"**条件。
2. **"类别"**根据服务器定义的类别进行筛选。每个事件都会被分配到一个类别。类别说明如下：
    - **"水平警报"**由过程水平条件生成的事件。例如，罐液位 > 10。
    - **"偏差警报"**由偏差条件生成的事件。例如，罐液位 ± 10。
    - **"更改率警报"**由更改率条件生成的事件。
3. **"严重性"** 根据严重性级别进行筛选。级别范围介于 0 到 1000 之间；1000 为最严重。每个事件都会被分配到一个严重性。
4. **"区域"**根据过程区域进行筛选，以仅获取该区域的警报和事件。用来组织警报和事件信息的区域。
5. **"源"**根据源进行筛选，以仅获取该源的事件。源是由属于某个区域的源 (例如服务器标记) 所创建的警报和事件区域。

注意：警报和事件插件允许通过服务器标记配置条件。例如，可以通过警报和事件插件来配置温度标记，以便在达到最大值时生成事件。有关警报和事件插件的详细信息，请联系 OPC 供应商。

另请参阅：项目属性 - OPC AE

可选接口
AE 服务器接口不支持以下可选接口：

- **IOPCEventServer::QueryEventAttributes** 该接口用于管理不受服务器支持的事件属性。属性允许将自定义信息添加到事件中 (例如，特殊消息或服务器标记值)。这也适用于 IOPCEventSubscriptionMgt::SelectReturnedAttributes 接口和 IOPCEventSubscriptionMgt::GetReturnedAttributes 接口。
- **IOPCEventServer::TranslateToItemIDs** 此接口允许 AE 客户端获取与事件相关的 OPC DA 项。这是因为在某些情况下，事件与服务器标记的值相关。
- **IOPCEventServer2**：此接口允许客户端启用/禁用区域和源。此接口不受服务器支持，因为它允许一个客户端启用/禁用所有客户端的区域或源。

注意：AE 服务器接口不支持跟踪事件。

OPC UA

支持的版本
1.02 优化二进制 TCP

概述
OPC 统一架构 (UA) 是由 OPC Foundation 在数十个成员组织的协助下共同建立的开放标准。它提供了另外一种将工厂车间数据共享到业务系统的方法 (从车间到顶层)。UA 还提供了一种在不使用 Microsoft DCOM 的情况下将远程客户端安全连接至服务器的方法。它能够通过防火墙和 VPN 连接进行安全连接。UA 服务器的实施支持优化二进制 TCP 和 DA 数据模型。

注意：目前暂不支持通过 HTTP/SOAP web 服务和针对复杂数据的 UA。有关详细信息，请参阅 OPC UA Configuration Manager 帮助文件。

OPC UA 配置文件

OPC UA 是一个多部分规范，用于定义许多称之为“特征”的服务和信息模型。特征将分组到配置文件中，用于说明 UA 服务器或客户端所支持的功能。有关各 OPC UA 配置文件的完整列表和说明，请参阅 http://www.opcfoundation.org/profilereporting/index.htm。

完全支持的 OPC UA 配置文件

- 标准 UA 服务器配置文件
- 核心服务器相关内容
- 数据访问服务器相关内容
- 安全策略 - Basic128Rsa15
- 安全策略 - Basic256
- 安全策略 - None
- UA-TCP UA-SC UA 二进制

部分支持的 OPC UA 配置文件

- 基础服务器行为相关内容

注意：此配置文件不支持安全管理员 - XML 架构。

另请参阅：项目属性 - OPC UA

OPC .NET

支持的版本

1.20.2

概述

OPC .NET 是由 OPC Foundation 提供的一系列 API，它利用 Microsoft 的 .NET 技术，允许 .NET 客户端连接到服务器。此服务器支持 OPC.NET 3.0 WCF，以前称为 OPC Xi。与其他 OPC .NET API 不同，OPC .NET 3.0 采用 Windows Communication Foundation (WCF) 进行连接，在避免出现 DCOM 问题的同时还具备以下优势：

- 通过多个通信约束+- (例如，命名管道、TCP、基本 HTTP 和 Ws HTTP) 进行安全通信。
- 整合了 OPC 典型接口。
- 可实现 Windows 环境的简单开发、配置和部署。

服务器使用 OPC Foundation 提供的 OPC .NET 3.0 WCF Wrapper 的自定义版本增加了 OPC .NET 3.0 支持。封套元素作为 "xi_server_runtime.exe" 系统服务运行。它包括现有服务器的 OPC AE 和 DA 接口，允许 WCF 客户端访问服务器的标记和警报数据。但不支持历史数据访问 (HDA)。

注意：只有在服务器启动且接口启用的情况下，才可启动 OPC .NET 服务。与 OPC DA 不同，客户端无法启动服务器。有关配置的详细信息，请参阅项目属性 - OPC .NET。

要求

要安装和使用 OPC .NET 3.0，在安装服务器之前，计算机上必须存在 Microsoft .NET 3.5。

DDE

支持的格式

CF_Text
XL_Table
高级 DDE

概述

虽然此服务器的主要作用是一个 OPC 服务器，但仍有许多应用程序需要动态数据交换 (DDE) 才能共享数据。因此，服务器将提供访问 DDE 应用程序的权限，这些应用程序支持以下 DDE 格式之一：CF_Text、XL_

Table 和高级 DDE。CF_Text 和 XL_Table 是由 Microsoft 开发的标准 DDE 格式，与所有 DDE 感知应用程序一起使用。高级 DDE 是一种高性能格式，受到许多特定于工业市场的客户端应用程序的支持。

CF_Text 和 XL_Table
DDE 格式 CF_Text 是 Microsoft 定义的标准 DDE 格式。所有 DDE 感知应用程序都支持 CF_Text 格式。XL_Table 是 Microsoft 定义的标准 DDE 格式，可供 Excel 使用。有关 DDE 的详细信息，请参阅如何通过服务器使用 DDE。

高级 DDE
高级 DDE 是 Rockwell Automation 定义的 DDE 格式。如今，所有 Rockwell 客户端应用程序均采用高级 DDE 感知。高级 DDE 是标准 CF_Text 格式的变型，可使更多的数据以更高的速度 (和更好的错误处理) 在应用程序之间传输。

要求
要使 DDE 接口与服务器相连，必须允许运行时与桌面交互。有关详细信息，请参阅如何...允许桌面交互作用。

另请参阅：项目属性 - DDE

FastDDE/SuiteLink

概述
FastDDE 是 Wonderware Corporation 定义的 DDE 格式。它允许更多的应用程序以高于一般 DDE 的速度 (和更好的错误处理) 在应用程序之间传输。SuiteLink 是继 FastDDE 之后的客户端/服务器通信方法。它以 TCP/IP 为基础，在带宽和速度方面均有所改进。所有 Wonderware 客户端应用程序均支持 FastDDE 和 SuiteLink。

注意：Wonderware 连接工具包可用于同时提供 OPC 和 FastDDE/SuiteLink 连接，且无需使用中间桥接软件即可快速访问设备数据。

出于安全考虑，建议用户使用最新的 Wonderware DAServer Runtime 组件。有关详细信息和可用下载，请参阅 Invensys 全球技术支持 WDN 网站。

要求
要使 FastDDE 接口与服务器相连，必须允许运行时与桌面交互。有关详细信息，请参阅如何...允许桌面交互作用。

另请参阅：项目属性 - FastDDE/SuiteLink

iFIX 本机接口

概述
iFIX 本机接口允许直接连接到与本地 iFIX 应用程序，而无需使用 iFIX OPC 电动工具便可简化连接任务。在受支持的情况下，此接口还可以优化服务器与 iFIX 进程数据库 (PDB) 之间的连接。

另请参阅：项目属性 - iFIX PDB 设置

ThingWorx 本机接口

概述
ThingWorx 作为连接平台，允许用户基于其设备数据创建有用且可行性情报。KEPServerEX ThingWorx 本机接口使用户只需利用 ThingWorx Always On 技术进行很少的附加配置即可向 ThingWorx Platform 提供数据。随着 ThingWorx 新一代 Composer 的推出，ThingWorx 本机接口得到了更新，从而更好地与 Composer 用户界面集成。仅传统模式才需要用户手动运行服务来将属性添加到资产。

如 ThingWorx 文档所述，“ThingWorx 应用程序密钥”的配置对于提供安全的环境至关重要。使用的应用程序密钥应提供适当的权限，以便在服务器实例和 ThingWorx 平台之间进行正确的数据交换。

ThingWorx 本机接口支持“存储并转发”功能，可在工业服务器与 ThingWorx 平台断开连接时缓存属性更新。

另请参阅：
项目属性 - ThingWorx 本机接口

完成率示例
存储并转发系统标记

访问管理菜单

“管理菜单”工具可用于查看和/或修改用户管理设置并启动服务器应用程序。要访问“管理菜单”，可右键单击“系统托盘”中的“管理”图标。

“配置”：此选项可启动 OPC 服务器的配置。

“启动运行时服务”：此选项可启动服务器运行时进程并加载默认运行时项目。

“停止运行时服务”：此选项可断开所有客户端，并在停止服务器运行时进程前保存默认运行时项目。

“重新初始化”：此选项可断开所有客户端，并重置运行时服务器。可在不停止服务器运行时进程的情况下，自动保存并重新加载默认运行时项目。

“重置事件日志”：此选项可重置事件日志。重置日期、时间和源已添加至配置窗口中的事件日志。

“设置...”：此选项可启动“设置”组。有关详细信息，请参阅设置。

“OPC UA 配置”：此选项可启动 OPC UA Configuration Manager (如可用)。

“OPC .NET 配置”：此选项可启动 OPC .NET Configuration Manager。

Quick Client：此选项可启动 Quick Client。

License Utility：此选项可启动服务器的 License Utility。

“帮助”：此选项可启动服务器的帮助文档。

“支持信息”：此选项可打开一个对话框，其中包含与服务器和当前安装使用的驱动程序相关的基本汇总信息。有关详细信息，请参阅服务器汇总信息。

“退出”：此选项可关闭“管理”，并将其从“系统托盘”中移除。如需再次查看，请在 Windows 的“开始”菜单中将其选中。

设置

要访问“设置”组，右键单击“系统托盘”中的“管理”图标。选择**“设置”**。有关详细信息，请从下表中选择一个链接。

设置 - 管理
设置 - 配置
设置 - 运行时进程
设置 - 运行时选项
设置 - 事件日志
设置 - ProgID 重定向
设置 - 用户管理器
设置 - 配置 API 服务
设置 - 证书存储
设置 - 服务端口

安全策略 - 此插件可用于用户权限和访问控制。请咨询产品帮助系统。
Local Historian - 此插件可用于数据存储和访问。请咨询产品帮助系统。
IoT Gateway - 此插件可用于工业物联网集成。请咨询产品帮助系统。

设置 - 管理

“管理”组用于配置运行时管理的操作。

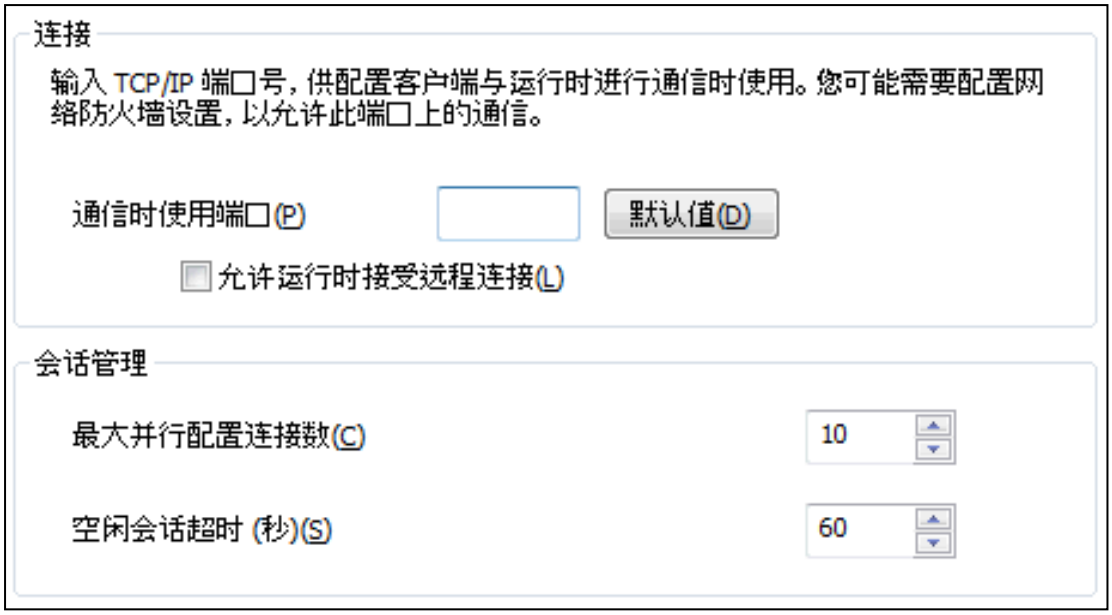
“自动启动管理”: 启用时，此属性可使“管理”自动启动。“管理”是一个系统托盘应用程序，可快速链接至各个服务器工具，包括“设置控制台”、“配置”、Licensing Utility、“用户管理器控制台”，以及用于停止和启动“运行时服务”的控件。

“产品语言选择”: 从下拉菜单中选择首选的用户界面语言。

提示: 语言设置默认为安装语言，而安装语言则默认为操作系统中的语言设置 (如果可能)。

设置 - 配置

“配置”组用于配置“配置”连接到“运行时”以及与其进行交互的方式。



连接

"通信时使用端口"：此属性为将用于在"配置"和"运行时"之间进行通信的 TCP/IP 端口。要获取默认设置，请单击**"默认"**。

"允许运行时接受远程连接"：启用时，运行时可接受远程连接。默认设置为禁用状态。

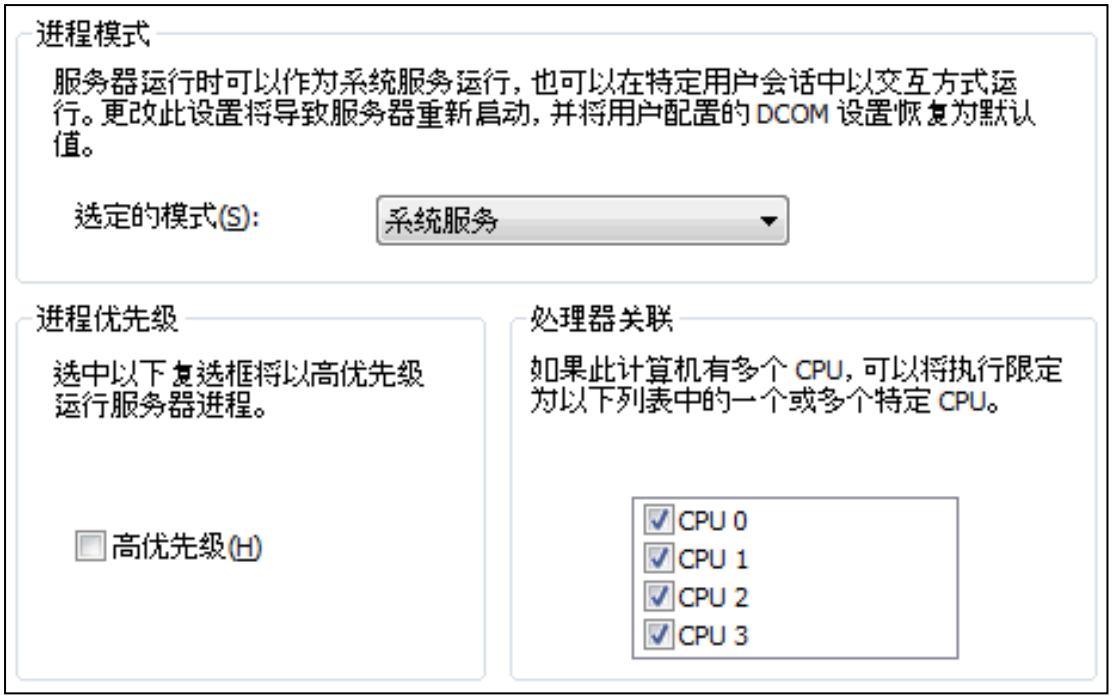
会话管理

"最大并行配置连接数"：指定一次可对运行时进行的配置连接数。范围为 1 到 64。默认值为 10。

"空闲会话超时"：设置控制台连接在关闭之前可保持非活动状态的时间长度。范围为 10 到 3600 秒。默认值为 60 秒。

设置 - 运行时进程

"运行时进程"组用于指定服务器运行时的进程模式，以及如何利用 PC 资源。



"选定的模式"：此属性用于指定服务器作为**"系统服务"**还是**"交互式"**运行。默认情况下，服务器作为系统服务安装并运行。更改此设置会导致所有客户端 (配置和进程) 断开连接，还会导致服务器停止和重新启动。还会将用户配置的 DCOM 设置恢复为默认值。

"高优先级"：此属性用于将服务器进程优先级设置为高。默认设置为正常。启用时，此设置允许服务器具有优先访问资源的权限。

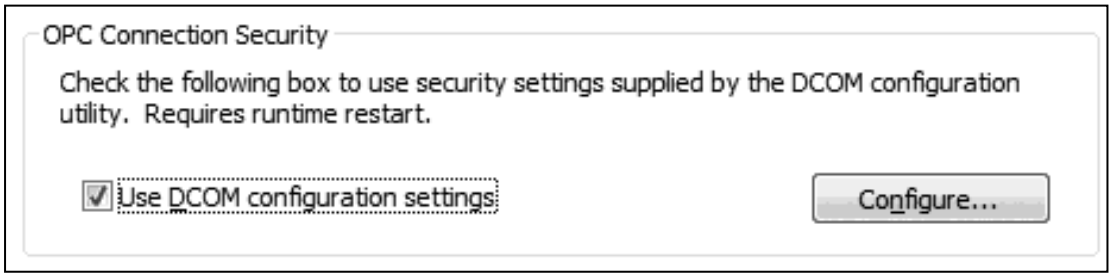
注意：Microsoft 不建议将应用程序设置为高优先级，因为它可能会对在同一系统中运行的其他应用程序产生不利影响。

"处理器关联"：此属性用于指定服务器在包含多个 CPU 的 PC 上运行时，可在哪些 CPU 上执行服务器。

设置 - 运行时选项

"运行时选项"(Runtime Options) 组用于更改在运行时中执行的项目中的设置。

OPC 连接安全性



"使用 DCOM 配置设置"：启用后可使用 DCOM 配置的身份验证和安全性能。

"配置..."单击以启动"DCOM 配置实用程序"，来指定安全级别并限制某些用户和/或应用程序的访问权限。

禁用此设置时，服务器将覆盖针对应用程序而设置的 DCOM 设置，且不会对接收自客户端应用程序的调用执行任何身份验证。在代表客户端应用程序执行任何操作时，它将模拟客户端的安全性。禁用此设置可提供最低的安全级别，但不建议如此。如果选择此设置，请确保客户端和服务器应用程序在安全环境下运行，从而使应用程序不受安全威胁。

项目备份

“在替换前备份运行时项目”：此属性使得运行时项目可在被覆盖之前进行备份。备份位置显示于事件日志中。默认情况下，此选项处于启用状态。

注意：如果在连接到运行时之时选择**“新建”**或**“打开”**，则运行时项目将被覆盖。此外，在离线处理项目的同时连接到运行时可能会导致运行时项目替换。

“保留最新的”：此属性可限制要保存到磁盘的备份文件数。范围为 1 到 1000。默认值为 10。

“立即清理...”：此属性将调用一个确认对话框，允许用户删除所有运行时项目备份。执行此操作不会影响当前正在运行的项目。

提示：出于故障恢复的目的，最好定期保存项目文件的副本。这些备份的默认目录为：

对于 64 位操作系统版本，备份项目文件保存在以下位置：
C:\ProgramData\Kepware\KEPServerEX\V6\Project Backups

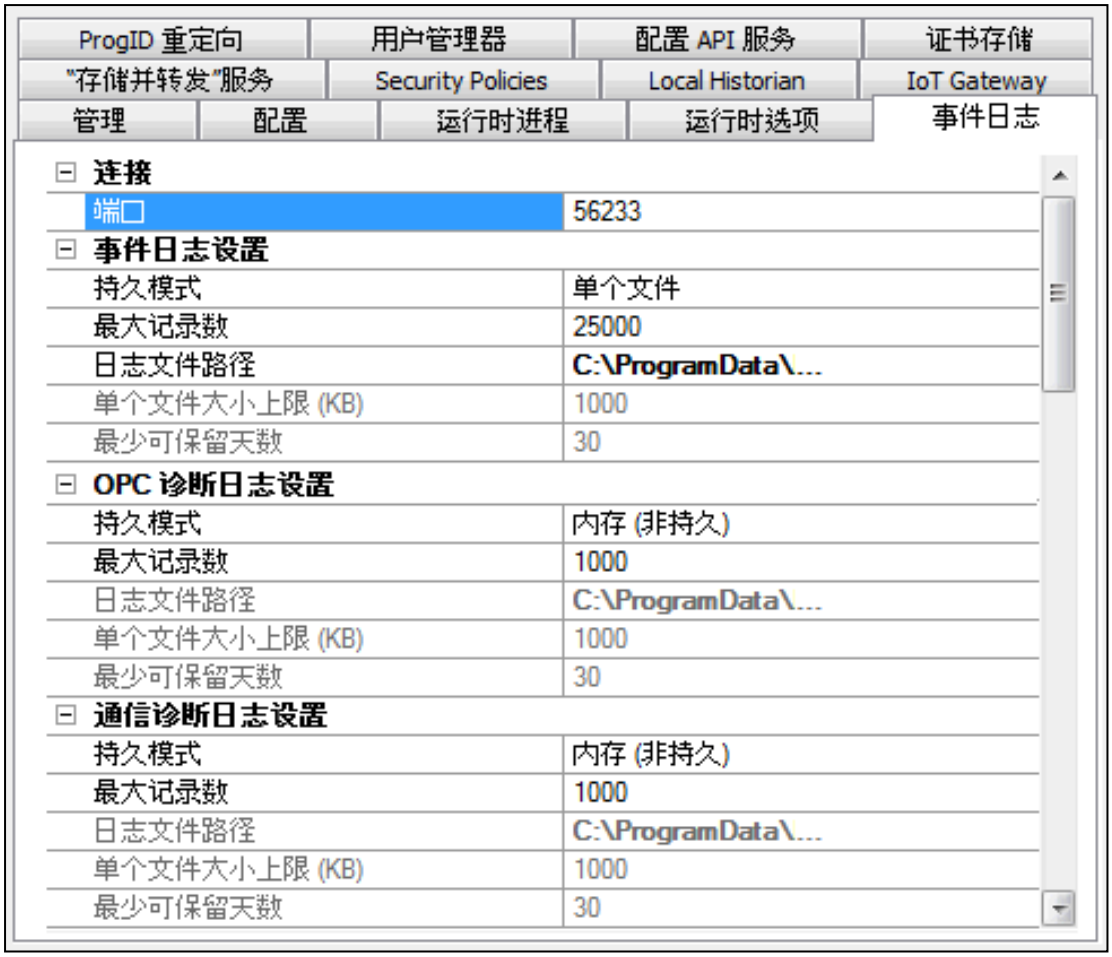
对于 32 位操作系统版本，备份项目文件保存在以下位置：
C:\ProgramData(x86)\Kepware\KEPServerEX\V6\Project Backups

提示：如果文件已保存到另一个位置，搜索 *.opf、*.sopf 或 *.json 可查找可用的项目文件。

设置 - 事件日志

“事件日志”组用于定义事件日志、OPC 诊断日志、通信诊断日志的通信和持久化设置。

为每个单独日志类型所做的设置独立于其他日志类型的设置。



连接

"端口"：指定用于在"日志"和"运行时"之间进行通信的 TCP/IP 端口。有效范围为 49152 到 65535。要恢复默认端口设置，请输入一个空值。

事件日志设置

"持久模式"：此图标可用于打开日志的持久模式。选项包括"内存"、"单个文件"和"长期数据存储"。"事件日志设置"的默认设置是"单个文件"。"OPC 诊断日志设置"和"通信诊断日志设置"的默认设置均为"内存 (非持久)" (Memory (no persistence))。选项说明如下：

- **"内存 (非持久)"**：选择此项时，此模式将记录内存中的所有事件，但不会生成磁盘日志。在开始删除最早的记录之前，保留指定数量的记录。每次启动服务器时，都会移除内容。
- **"单个文件"**：选择此项时，此模式会生成一个基于磁盘的日志文件。在开始删除最早的记录之前，保留指定数量的记录。启动服务器时，内容会从磁盘上的该文件中恢复。
- **"长期数据存储"**：选择此项时，此模式会将大量潜在记录保留在磁盘分布于许多文件的数据存储中。在从磁盘中删除记录之前，将记录保留指定的天数。启动服务器时，内容会从磁盘上的分布式文件存储中恢复。

"最大记录数"：指定日志系统在开始删除最早记录之前可保留的记录数。仅在将"持久模式"设置为"内存"或"单个文件"时，此选项才可用。有效范围为 100 到 100,000 项记录。默认设置为 25,000 项记录。

注意：如果将此属性设置为小于日志的当前大小的值，则日志将会被截断。

"日志文件路径"：指定磁盘日志的存储位置。仅当"持久模式"设置为"单个文件"或"长期数据存储"时可用。

注意：尝试使用映射路径持久保留诊断数据可能会失败，原因在于事件日志服务正在 SYSTEM 帐户的上下文中运行，并且无法访问本地主机上的映射驱动器。使用映射路径的用户自行决定执行此操作。建议改为使用统一命名约定 (UNC) 路径。

"单个文件大小上限"：指定在启动新的数据存储文件之前单个数据存储文件必须要达到的大小。仅当"持久模式"设置为"扩展数据存储"时可用。有效范围为 100 到 10000 KB。默认设置为 1000 KB。

"最少可保留天数"：指定将各个数据存储文件从磁盘删除时，存储于文件中的最新记录已至少存在这些天数。仅当"持久模式"设置为"扩展数据存储"时可用。有效范围为 1 到 90 天。默认设置为 30 天。

另请参阅：内置诊断

保存到文件时，监视 Windows 事件查看器，以查看与数据存储到磁盘中的持久性相关的错误。

从磁盘恢复持久数据存储

"事件日志"会在启动或出现下列情况时从磁盘恢复记录：

1. 将"持久模式"设置为"单个文件"或"长期数据存储"。

   注意：选择单个文件持久性后，服务器将在向客户端提供任何记录之前，从磁盘加载所有持久记录。

2. 日志文件路径设置为其中包含有效持久日志数据的目录。

长期数据存储持久性

"长期数据存储持久模式"能够从磁盘加载大量记录。要保持响应状态，日志服务客户端需在从磁盘加载记录期间请求记录。加载记录存储时，将为客户端提供日志中的所有记录，无论其是否经过筛选。加载所有记录后，服务器将应用筛选器，并按时间顺序对记录进行排序。客户端视图随即自动更新。

注意：加载大量记录存储可能导致日志服务器的响应能力不如平时。加载和处理完成后，其将重新获得全面的响应能力。在加载和处理完成期间，资源使用率高于平时。

磁盘完整行为

"长期数据存储持久模式"能够快速填满存储介质，尤其是在持续进行 OPC 诊断时。如果持续处理记录时发生磁盘错误，则错误将发布到 Windows 事件查看器。

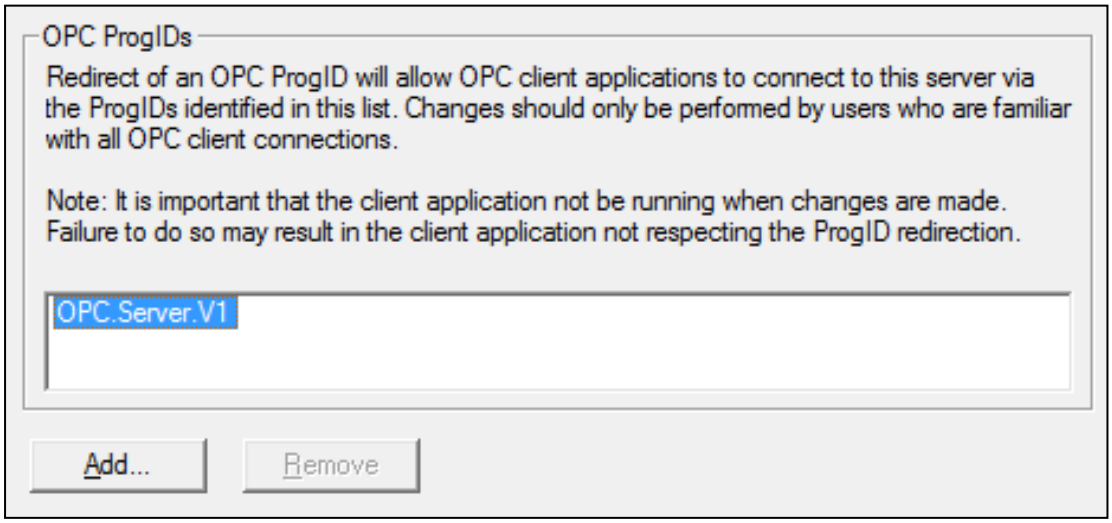
另请参阅：OPC 诊断查看器

如果不存在任何机制来保护事件日志系统的内容，则该事件日志系统将是无用的。如果操作员可以更改这些属性或重置日志，目的将会丢失。利用"用户管理器"来限制操作员可访问的功能。

设置 - ProgID 重定向

许多 OPC 客户端应用程序都通过 OPC 服务器的 ProgID 连接到 OPC 服务器。需要迁移或升级到新 OPC 服务器的用户通常更倾向于在无需更改其标记数据库 (其中可包含数千个链接到 OPC 服务器 ProgID 的标记) 的情况下进行此操作。此服务器提供了 ProgID 重定向，以帮助用户完成这些过渡。

"ProgID 重定向"功能允许用户输入传统服务器的 ProgID。服务器会创建必要的 Windows 注册表项，以允许客户端应用程序连接到使用传统服务器 ProgID 的服务器。



"添加"：此按钮用于将 ProgID 添加到重定向列表。单击后，系统将会调用"添加新的 ProgID"对话框。有关详细信息，请参阅下面的"添加新的 ProgID"。

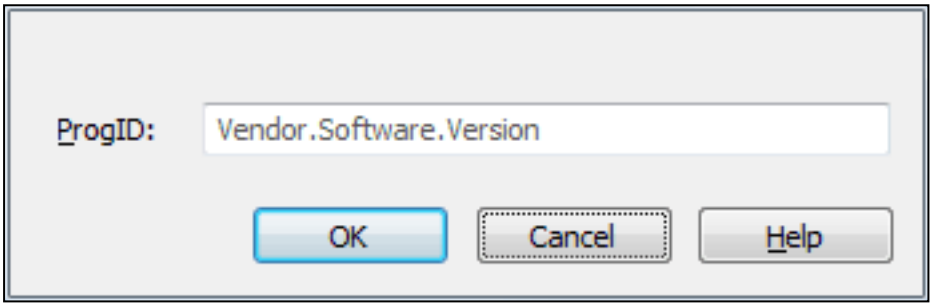
"移除"：此按钮用于从重定向列表中移除选定的 ProgID。

注意：使用 OpcEnum 服务来定位 OPC 服务器的 OPC 客户端应用程序不能浏览重定向的 ProgID。在大多数情况下，用户都可以手动将重定向的 ProgID 输入至客户端应用程序中。

添加新的 ProgID

有关详细信息，请参阅以下说明。

1. 在**"ProgID 重定向"**组中，单击**"添加"**。
2. 在 **ProgID** 中，输入传统服务器的 ProgID。



3. 完成后，单击**"确定"**。

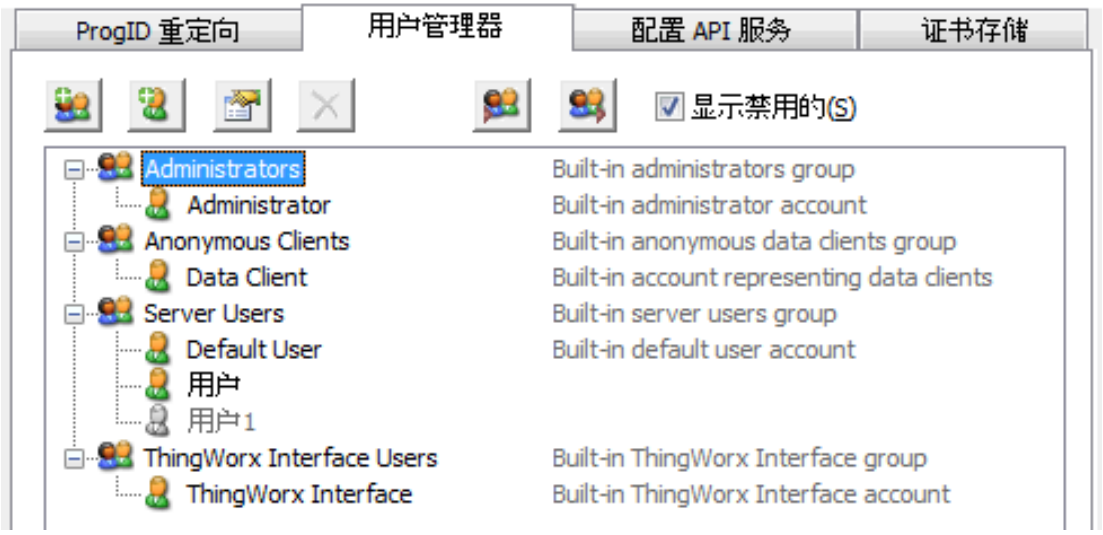
传统服务器 ProgID 添加到重定向列表时，客户端应用程序不应运行。未遵守此警告可能会导致客户端应用程序不遵循重新定向的 ProgID。

设置 - 用户管理器

"用户管理器"可控制客户端对项目对象 (即通道、设备、标记等) 及其对应功能的访问。用户管理器允许由用户组指定权限。例如，"用户管理器"可以根据"数据客户端"用户在"匿名客户端"用户组中的权限限制其对项目标记数据的访问。"用户管理器"还可以通过其导入/导出功能在服务器安装之间传输用户信息。

"用户管理器"具有三个内置组，每个组包含一个内置用户。默认组是"管理员"、"服务器用户"和"匿名客户端"。默认用户是"管理员"、"默认用户"和"数据客户端"。用户不能对说明字段进行重命名或更改。默认组和默认用户都不能禁用。

注意：虽然不能更改管理员的设置，但可以添加其他管理用户。



"新建组"：单击此按钮可添加新的用户组。*有关详细信息，请参阅用户组属性*。

"新建用户"：单击此按钮可将新用户添加至选定的用户组。该功能对匿名客户端是禁用的。*有关详细信息，请参阅用户属性*。

"编辑属性"单击此按钮可允许用户编辑选定用户或用户组的属性。

"禁用选定用户/组"：单击此按钮可禁用选定用户或用户组。此功能仅可用于自定义用户和用户组。禁用用户组会禁用组中所有的用户。
注意：禁用用户或用户组会调用**"显示禁用的"**选项。如果启用，此选项会使所有禁用的用户和用户组在用户组和用户列表中可见。

"恢复选定用户/组"：单击此按钮可恢复选定的用户或用户组。恢复用户组会使组内的用户恢复到禁用之前的状态。只有在禁用用户或用户组之后，此图标才可用。
注意：如果所有禁用的用户和用户组都已恢复，则不会显示**"显示禁用的"**选项。

"导入用户信息"：单击此按钮可从 XML 文件导入用户信息。为使导入成功，所选文件必须已从服务器的"管理"实用程序导出。内置管理员登录后，才会启用此功能。

"导出用户信息"：单击此按钮可将用户信息导出为 XML 文件。这对于需要将项目从一台计算机移到另一台计算机的用户来说很有用。管理员还可以选择为 XML 文件提供密码保护：如果使用了此选项，要想成功导入至新计算机，必须输入正确的密码。无法编辑和重新导入 XML 文件。此功能始终为启用状态。

- "导入/导出用户信息"功能已在服务器版本 5.12 中发布。在尝试导出之前，在使用先前的服务器版本时设置的任何用户密码都必须在 5.12 中更改；否则导出将失败。

- 在升级服务器或导入"用户信息"之后，建议您检查"用户管理器"权限的准确性。

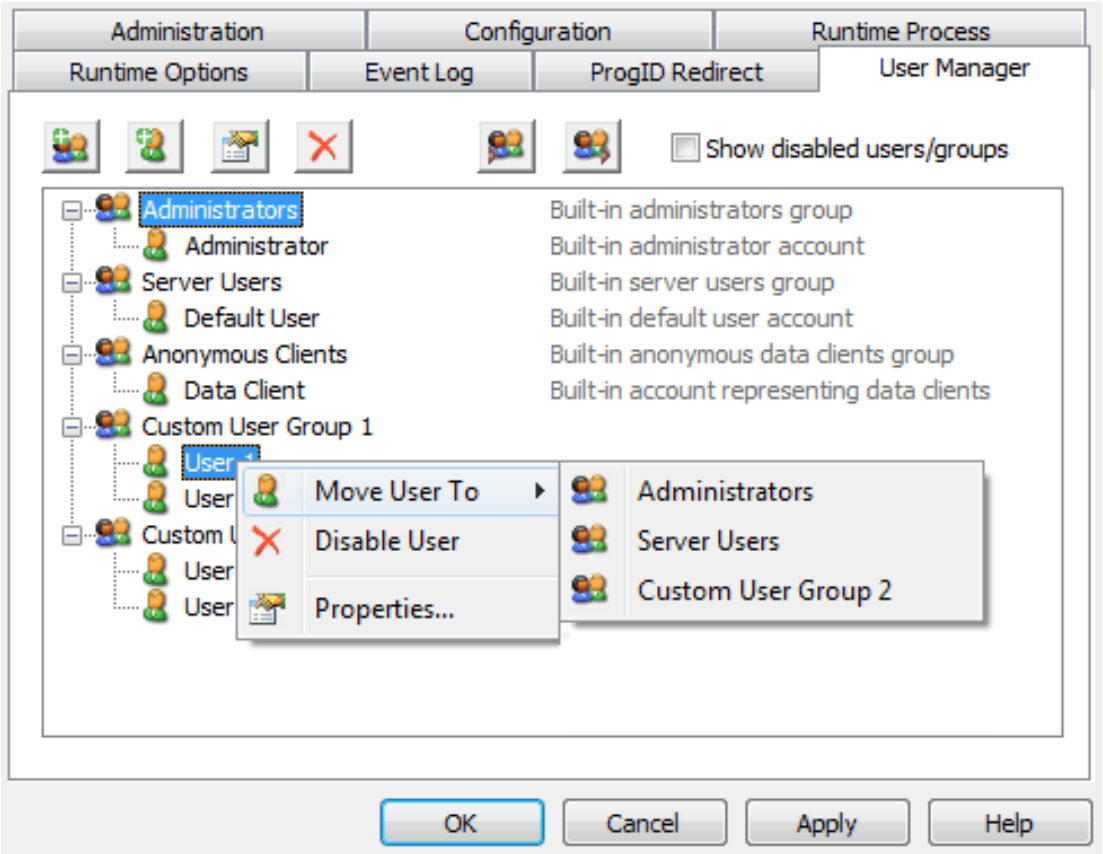
注意：尽管一经创建，自定义用户和用户组就不能删除，但"导入用户信息"选项会用导入的用户和用户组替换现有用户和用户组 ("管理员"内置用户除外)。

- 为了保存项目，建议用户在完成后，导出一份用户信息副本。若无正确的用户信息，项目无法加载。

另请参阅：ThingWorx Interface 用户，了解如何连接 ThingWorx Platform。

访问其他设置

可以通过用户组和用户的上下文菜单访问快捷方式和其他设置。

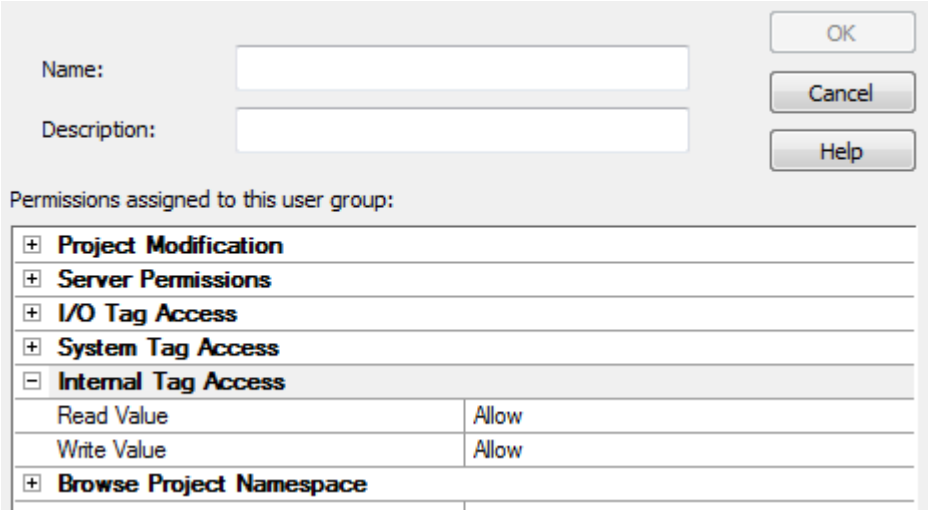


"将用户移至"此选项可将用户移动到不同的用户组。组的状况并不重要：禁用和启用的组都将出现在列表中。移动到已禁用组的活动用户也会变为禁用状态。已禁用的用户移动到已启用的组时仍会保持禁用状态，直到对其状态进行更改。

用户组属性

也可以通过右键单击用户组并选择**"属性"**访问用户组属性。

提示：要快速允许或拒绝类别中的所有选项，请右键单击该类别并选择**"全部允许"**或**"全部拒绝"**。显示粗体文本的设置表示它的值已更改。更改一经保存，文本即正常显示。



"名称"：单击此图标以开设新用户组的名称。允许的最大字符数量为 31。不允许出现重复的名称。

"说明"：此可选属性提供了用户组的简短说明。这对操作者创建新用户帐户可能特别有用。允许的最大字符数量为 128。

"分配给此用户组的权限"：此字段可分配选定用户组的权限。权限分为以下类别：项目修改、服务器权限、I/O 标记访问、系统标记访问、内部标记访问和浏览项目命名空间。类别的详细信息如下：

- **"项目修改"**：此类别可指定用于控制默认项目修改的权限。
- **"服务器权限"**：此类别可指定用于控制对服务器功能进行访问的权限。匿名客户端不支持这些权限。
- **"I/O 标记访问"**：此类别可指定用于控制对设备级 I/O 标记数据进行访问的权限。这些标记需要设备通信，并被描述为服务器中的静态标记。
- **"系统标记访问"**：此类别可指定用于控制对"系统"标记进行访问的权限。这些标记以下划线开头，并存在于服务器定义的位置。有关详细信息，请参阅系统标记。
- **"内部标记访问"**：此类别可指定用于控制对内部标记进行访问的权限。这些标记或由驱动程序管理 (控制驱动程序操作的某些方面)，或由用户指定 (在插件级别)。
- **"浏览项目名称空间"**：此类别可指定用于控制对支持浏览的客户端中的项目命名空间进行浏览访问的权限。目前只有少数客户端类型支持这一类别。

提示：要查看类别中特定对象的详细信息，请选择该对象。

用户属性

可以通过双击用户或右键单击用户然后选择**"属性..."**，来访问用户属性。

"旧密码"：此字段应输入该用户以前激活的密码。

"密码"：输入用户在登录系统时必须输入的新密码或更新密码。密码区分大小写，最多允许 512 个字符。

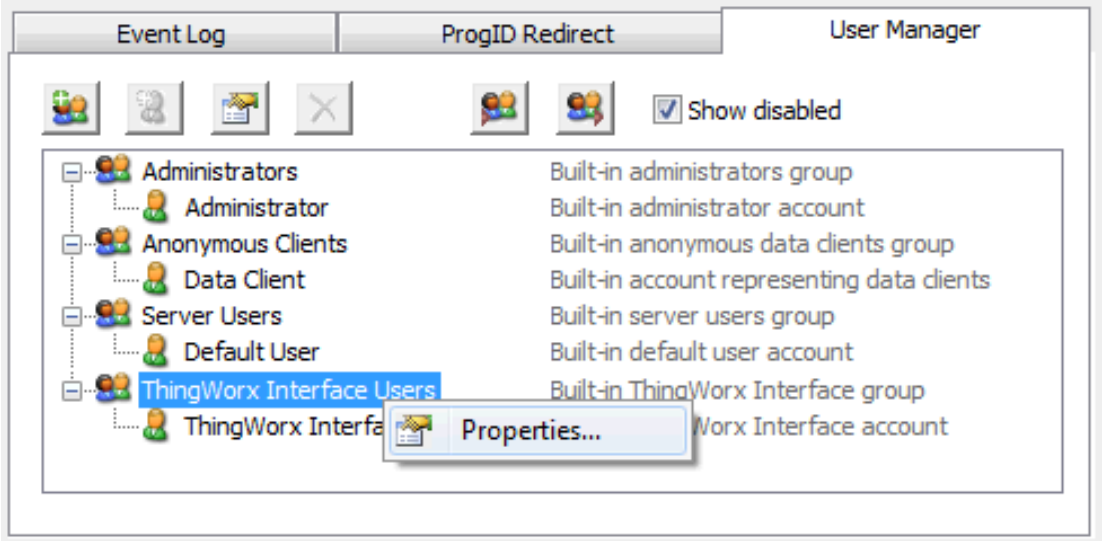
"确认密码"：重新输入相同的密码。必须在"新密码"和"确认密码"字段中输入完全相同的内容。

注意：建议密码至少为 14 个字符，并包括大写和小写字母、数字和特殊字符的组合。避免众所周知、容易猜到的或常见的密码。

设置 - 用户管理器 ThingWorx Interface 用户

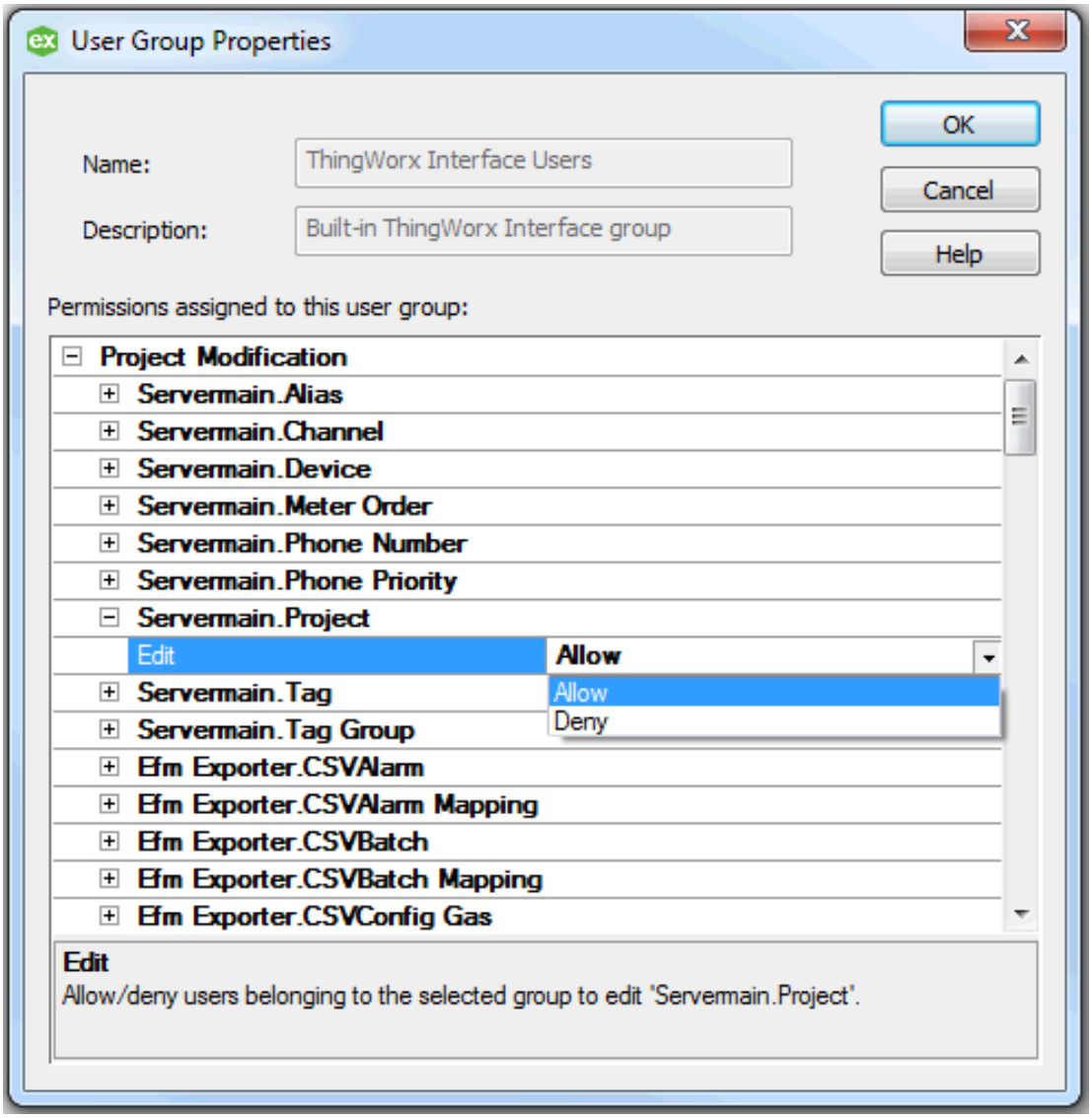
“用户管理器”控制客户端对于项目对象的访问和每个对象的相应功能。所有按钮和控件的作用如“常规用户管理器”部分中所述。

另请参阅：用户管理器



为了允许在服务器和 ThingWorx Platform 之间的数据传输具有足够访问权限，必须启用“项目修改”和“存储并转发”。要对此功能授予正确的访问权限：

1. 选择**“ThingWorx Interface 用户”**组。
2. 右键单击并选择**“属性...”**。
3. 展开**“项目修改”**组。
4. 找到并展开 **Servermain.Project** 权限。
5. 在下拉菜单中选择**“允许”**授权更改项目文件的权限。
6. 单击**“确定”**关闭。



设置 - 配置 API 服务配置

“配置 API 服务”已在安装时完成。如果需要对设置进行调整，请访问“配置 API 服务”设置，方法是：右键单击系统托盘中的“管理”图标，然后选择**“设置”|“配置 API 服务”**。

如果管理图标不在系统托盘中，请将其重新启动，方法是选择 **“开始” | “所有程序” | Kepware | KEPServerEX 6 | “KEPServerEX 6 管理” | “设置”“开始” | “所有程序” | PTC | OPC Aggregator | “OPC Aggregator 管理” | “设置”**。

“启用”：选择“是”可启用配置 API 服务器。如果禁用 (“否”)，服务运行，但无法绑定到 HTTP 和 HTTPS 端口，并且客户端无法访问服务器。

“启用 HTTP”：选择“否”可将数据限制为仅传输至安全/加密协议和端点。选择“是”可允许未加密的数据传输。

提示：

1. HTTP 仅适用于内部网络，原因在于用户身份验证以纯文本形式传输。
2. 为了防止通过不安全的 HTTP 进行外部访问，此端口应该由 Windows 防火墙进行阻止。

“HTTP 端口”：指定 REST 客户端的 TCP/IP 端口以通过未加密的 HTTP 进行通信。有效范围为 1 到 65535。HTTP 和 HTTPS 端口不得匹配。默认端口号为 57412。

“HTTPS 端口”：指定 REST 客户端的 TCP/IP 端口通过安全的 HTTP 进行通信。有效范围为 1 到 65535。HTTP 和 HTTPS 端口不得匹配。默认端口号为 57512。

“CORS 允许来源”：指定以逗号分隔的域规范的已批准“白名单”，通过该文件可以访问跨域资源共享 (CORS) 请求的配置 API 服务器。

“恢复默认值”：单击右侧的蓝色链接可恢复默认 HTTP 和 HTTPS 端口值。

“在浏览器中查看”：单击右侧的蓝色地址链接可在浏览器中打开配置 API 文档登录页面。

“在浏览器中查看 (SSL)”(View in Browser (SSL))：单击右侧的蓝色地址链接可通过安全 URL 在浏览器中打开配置 API 文档登录页面。

事务日志记录

“持久模式”：选择系统日志的记录保留方法。默认设置为“内存 (非持久)”。选项包括：

- **“内存 (非持久)”**：将所有事件记录在内存中，而不生成可保存至磁盘的日志。在开始删除最早的记录之前，保留指定数量的记录。仅当服务器运行时，才可以使用内容。
- **“单一文件”**：生成可保存至磁盘的已记录日志文件。在开始删除最早的记录之前，保留指定数量的记录。启动服务器时，内容会从该文件中恢复。
- **“扩展数据存储”**：将潜在的大量记录保存至磁盘中的多个分布式文件。在从磁盘中删除记录之前，将记录保留指定的天数。启动服务器时，内容会从磁盘上的分布式文件中恢复。

“最大记录数”：指定在删除最早记录前日志中可保留的事物处理数。在将“持续模式”设置为“内存”或“单一文件”时，此选项可用。有效范围为 100 到 30000 项记录。默认设置为 1000 项记录。

注意：如果为此参数设置的值小于日志的当前大小，则日志将会被截短。

“日志文件路径”：指示日志在磁盘上的存储位置。当“持久模式”设置为“单一文件”或“扩展数据存储”时可用。

尝试使用映射路径持续存储诊断数据可能会失败，原因在于事务处理日志服务正在 SYSTEM 帐户的上下文中运行，并且无法访问本地主机上的映射驱动器。使用映射驱动器路径时请小心。建议使用统一命名约定 (UNC) 路径。

"最大单一文件大小"：指示单一数据存储文件的大小限制 (以 KB 为单位)，当数据存储文件大小达到大小限制时，将开始一个新的数据存储文件。当"持久模式"设置为"扩展数据存储"时可用。有效范围为 100 到 10000 KB。默认设置为 1000 KB。

"最少可保留天数"：指定单个数据存储文件在从磁盘中删除前保留的天数。当"持久模式"设置为"扩展数据存储"时可用。有效范围为 1 到 90 天。默认设置为 30 天。

"详细"：选择"是"可记录在日志中记录数据的详细级别。除了非详细日志记录中包括的参数外，详细日志记录还包括 HTTP 请求和响应主体。*有关详细信息，请参阅详细日志记录*。选择"否"可大大减少数据记录，并会保持较小的日志文件。

"证书管理"

注意：X.509 证书用于建立客户端和 REST 服务器之间的 SSL 通信。REST 服务器安装后，即可生成默认自签名证书，但是，在安全网络以外访问服务器需要受信任的证书。

"查看证书"：单击右侧的蓝色链接可打开当前证书，以便进行查看。

"导出证书"：单击右侧蓝色链接可将当前证书保存为 .PEM 格式 (例如用于导入到第三方 REST 客户端)。

"重新颁发证书"：单击右侧蓝色链接可创建一个新证书，以替换当前证书。

"导入证书"：单击右侧蓝色链接可导入 .PEM 格式的证书。

注意：证书在安装过程中创建，并且无需进行其他配置。当重新颁发或导入证书时，只有配置 API 停止并通过 Windows 服务控制管理器重新启动，或者重新启动系统后，才会应用新证书。

设置 - 证书存储

对于使用传输层安全性 (TLS) 或更早的安全套接字层 (SSL) 协议进行安全通信的功能，证书存储可用于为该功能配置证书。只有当安装了能够利用它的功能 (如 ThingWorx 本机接口) 时，才会出现此选项卡。

证书存储

功能	ThingWorx Native Interface
实例证书	
查看	查看
导出	导出
重新颁发	重新颁发
导入	导入
管理可信存储	
证书	<空白>
查看	查看
导出	导出
删除	删除
扩展可信存储	
导入	导入

"实例证书"

"查看"：单击"查看"链接查看当前选定功能的实例证书。

"导出"：将当前所选功能的实例证书保存到用户选择的目录。建议的文件名反映证书特征，但用户可以自行更改此名称。输出是 PEM 编码，且包括单一证书。

"重新颁发"：重新颁发当前所选功能的实例证书。证书存储生成的证书是自签名的，在 10 年后过期。

"导入"：导入当前所选功能的实例证书。使用此选项可导入由 TLS/SSL 对等方信任的证书颁发机构签名的证书。

"管理可信存储"

"证书"：可信存储可包含零到多个证书。用户必须选择要查看、导出或删除的证书。

"查看"：查看当前所选功能的当前所选可信证书。

"导出"：导出当前所选功能的当前所选可信证书。与实例证书一样，输出文件是 PEM 编码，且包含单个证书。

"删除"：删除当前所选功能的当前所选可信证书。如果对等方提供的证书在其信任链中包含此证书，则该功能不再信任此对等方。

"扩展可信存储"

"导入"：将一个或多个证书颁发机构签名的证书或自签名证书导入可信存储。该功能信任提供此证书 (或由已导入证书签名的证书) 的 TLS/SSL 对等方。

实例证书导入行为

- 导入文件必须包含证书和未加密的私钥。
- 如果证书包含无效签名，则无法导入。
- 如果证书已过期，系统将提示用户。TLS/SSL 对等方会拒绝过期的证书。

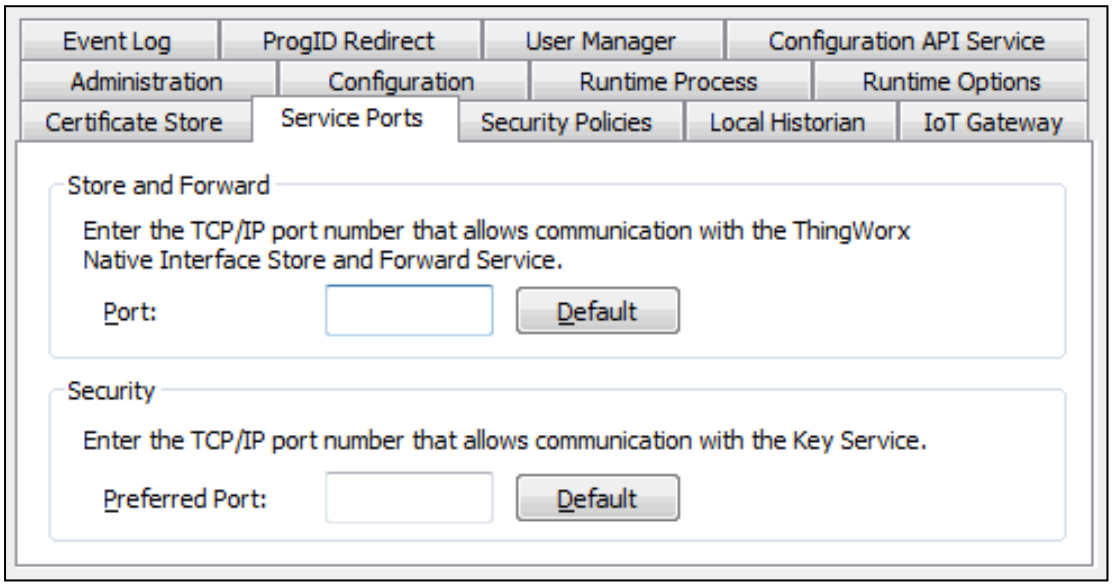
可信证书导入行为

- 导入文件应包含一个或多个证书。
- 密钥不是必需的，但可以存在于文件中。
- 如果一个或多个证书具有无效签名，则导入不成功。

- 如果一个或多个证书与可信存储中已存在的证书重复，则导入不会成功。
- 如果导入文件中的任何证书已过期，系统将提示用户。该功能可能会拒绝基于信任链中已过期证书的证书。

设置 - 服务端口

“管理”组用于配置运行时管理的操作。安装时自动配置服务端口管理设置。如果必须对设置进行更新，请访问“服务端口”系统设置，方法是：右键单击系统托盘中的“管理”图标，然后选择**“设置”|“服务端口”**。



存储并转发

“端口”：指定“存储并转发”客户端与“存储并转发”服务通信的 TCP/IP 端口。有效范围为 1024 到 65535。默认值由服务器配置。

“默认值”：单击以默认端口号填充此字段。

提示：

- 建议使用默认端口，除非与使用该端口的另一个服务器应用程序发生冲突。
- “存储并转发服务”不接受远程连接，因此，此端口分配不会对防火墙产生影响。
- 允许用户启用 SAF 所需的权限包括项目修改。通过用户管理器授予用户或组(可能是匿名客户端) 修改 servermain 项目的权限。ThingWorx 用户需要通过 ThingWorx Interface 用户组获得相同的访问权限，这可以参照用户管理器 ThingWorx Interface 用户中的步骤。

另请参阅：项目属性 ThingWorx

安全性

“首选端口”：指定密钥服务与服务器通信的 TCP/IP 端口。有效范围为 1024 到 65535。默认值由服务器配置。如果**“首选端口”**不可用或者因某种原因不适合使用，则服务会尝试获取备用端口。

“默认值”：单击以默认端口号填充此字段。

用户界面导航

“配置”提供了与服务器进行交互的常规方法。各插件和驱动程序添加了按钮、菜单和图标；标准界面元素如下所述。

菜单栏

“文件”	包括项目级命令；例如，“保存”、“打开”、“导入”和“导出”。
“编辑”	包括操作命令；例如，“复制”、“粘贴”和“新建通道”。
“视图”	包括显示命令；例如，用户界面的哪些元素是可见或隐藏的，以及要显示的树组织类型。
“工具”	包括配置命令；例如，常规选项、连接设置和“事件日志”筛选器。
“运行时”	包括服务器连接命令；例如，“连接...”、“断开连接”和“重新初始化”。
“帮助”	包括通过服务器、驱动程序或插件访问产品文档的命令。

按钮栏

标准按钮如下所述。插件和驱动程序根据活动项和视图的可用功能添加、移除、启用和禁用按钮。

“新建项目”：启动新项目文件的创建以替换活动项目。项目文件定义连接的设备及其设置和分组方式。

“打开项目”：可供用户浏览要加载的现有项目文件并替换活动项目。

“保存项目”：执行最近的任何更改，并将活动项目文件写入磁盘。

“另存为”：将活动项目及其更改写入新位置或文件名。

“新建通道”：为数据集合创建新组或媒体。

“新建设备”：为数据集合定义新的硬件组件或 PLC。

“新建标记组”：定义一个新的集合，以容纳可作为独立单元进行组织的数据点或标记。

“新建标记”：为集合定义一个新的数据点。

“主体标记创建”：定义在目标设备或环境中发现的标记。

“复制标记”：创建所选标记的副本。

“属性”：允许查看和编辑所选项的参数。

“撤销”：将值或项重置为最近更改前的配置。

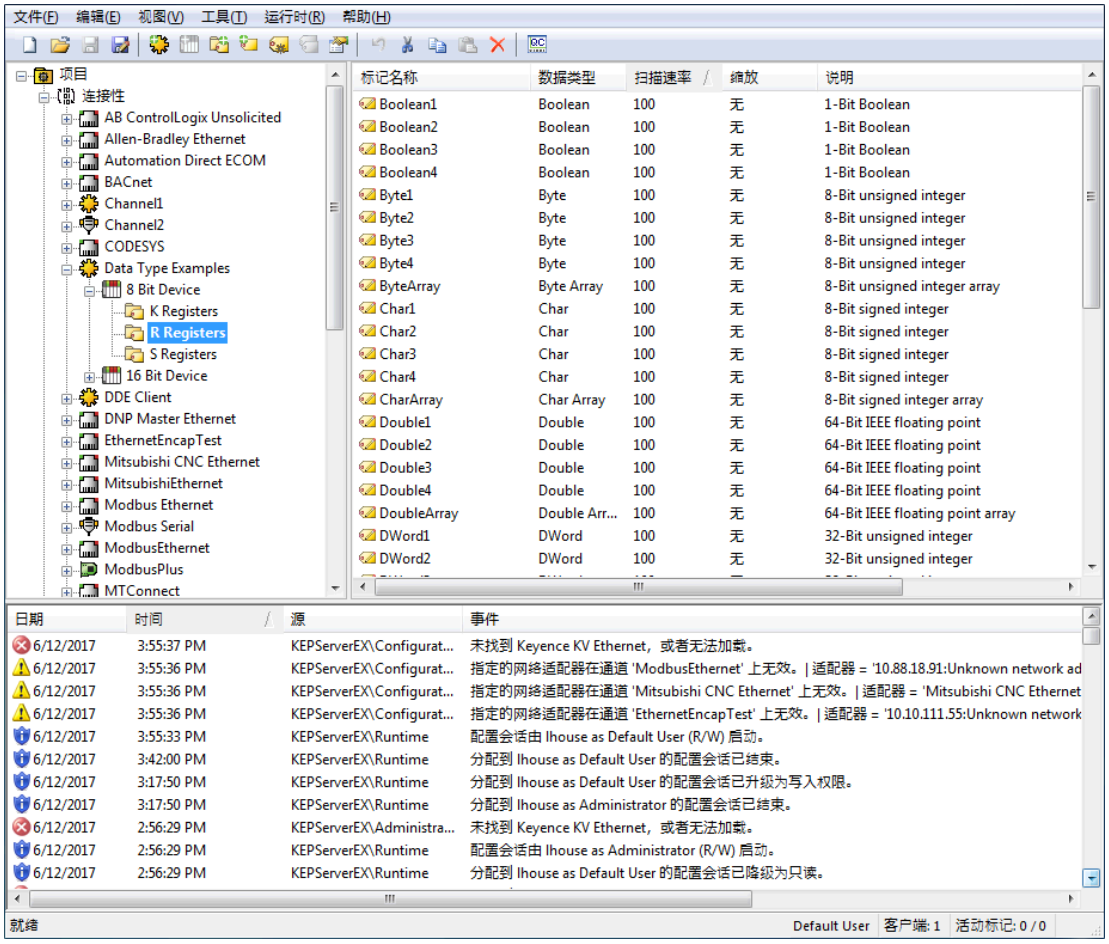
“剪切”：删除选定的项并将其存储在剪贴板上。

复制：创建选定项的副本并将其存储在剪贴板上。

“粘贴”：将剪贴板中的当前项插入所选区域。

“删除”：移除所选项和 / 或其定义。

"Quick Client"：运行集成的客户端接口。



项目树状视图

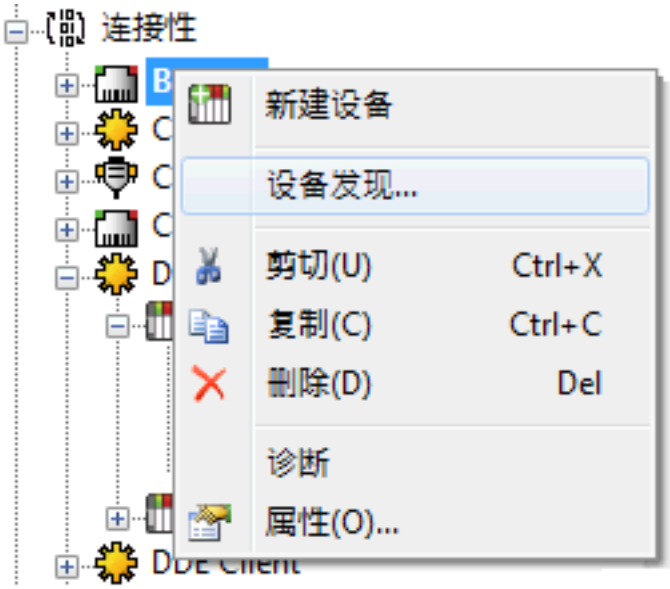
此视图在层次结构视图中显示当前项目内容、组织和设置。项目树状视图被设计为项目所有方面的统一位置。展开节点可以详细地深入了解设备、标记组或标记级别。特征和插件在树状视图中显示为节点，以便于在一个位置进行配置工作。树的主节点有：

“项目” - 可用于存储或更新活动项目的全局设置。
“连接性” - 可用于组织通道和设备，可以使用右键单击操作，并且可以访问属性以在“详细信息”窗格中显示。
“别名” - 可用于缩短系统资源、旧路径和复杂路由的映射，使其更加人性化，或实现 SCADA 兼容名称和快捷方式。
"Advanced Tags" - 可用于将操作或分析内置于标记处理中并进行存储。这是一个单独的产品插件。
"Alarms & Events" - 可用于定义和管理系统监控。这是一个单独的产品插件。
"DataLogge"r - 可用于在 ODBC 兼容数据库中组织和存储数据。这是一个单独的产品插件。
"EFM Exporter" - 可用于捕获和协调流数据以及趋势数据。这是一个单独的产品插件。
"IDF for Splunk" - 可用于将数据配置为送入数据管理和数据挖掘。这是一个单独的产品插件。
"IoT Gateway" - 可用于管理与企业系统、监控和分析的连接。这是一个单独的产品插件。
"Local Historian" - 可用于定义数据收集、日志记录、存储和保留。这是一个单独的产品插件。
"Scheduler" - 可用于协调数据收集、发布和带宽管理。这是一个单独的产品插件。
"SNMP Agent" - 可用于创建到信息技术和 SNMP 协议的通信桥接。这是一个单独的产品插件。

如果在很大的项目中或某些功能的使用频率较高，则可以通过筛选来自定义树。隐藏或显示**“视图”**菜单下的树节点。

“项目树”通过右击菜单提供了各种相应选项。例如，可以复制和粘贴设备及通道，以根据现有选项和设置来启动新配置。出现重复名称，通过添加编号 (如果粘贴次数较多，则编号递增) 来保持名称唯一。对于支持其他功能的驱动程序，这些选项也可用于右击菜单。

“设备发现”，可以在可访问网络中搜索兼容设备并自动添加这些设备。



详细信息视图

此视图会显示活动项目的多个配置选择选项之一。其信息与当前项目树状视图有关。

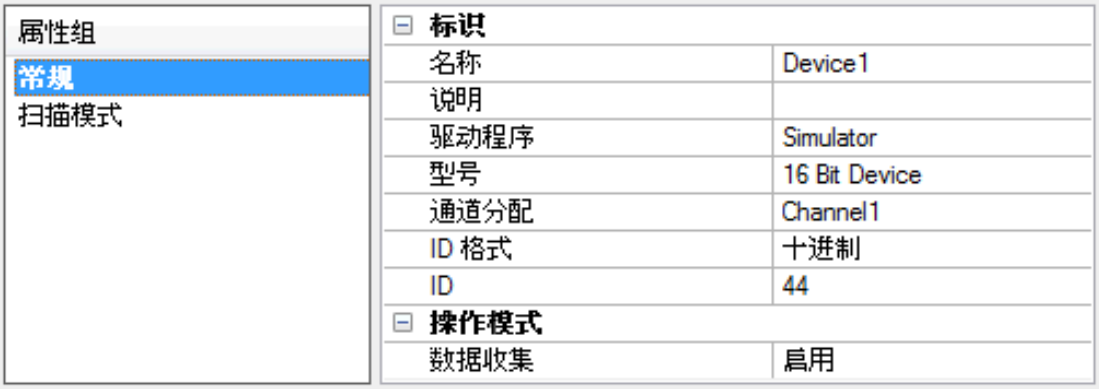
注意：选择项目树视图后，会始终显示"详细信息视图"列，直至选择了通道或设备。此时，列将恢复为显示设备或标记信息。

事件日志

此视图位于底部窗格中，显示四种类型的记录消息：来自服务器、驱动程序或插件的常规信息、安全警报、警告和错误。默认情况下，日志条目包括日期、时间、源和事件说明。*有关详细信息，请参阅事件日志选项。*

属性编辑器

可在属性编辑器中编辑某些属性。属性编辑器中的标准按钮操作如下：



默认值将所选属性组的设置恢复为其默认值 (包括已应用和待处理的更改)。

"确定"：退出属性编辑器并执行所有更改。

"取消"：退出属性编辑器但不执行待处理的更改。关闭属性编辑器具有相同的效果。

应用会在所有属性组中执行待处理的更改。

"帮助"：打开所选属性的帮助。

待处理的更改在应用之前显示为粗体。

项目属性

要通过配置访问“项目属性”组，请单击**“编辑”|“项目属性”**。有关详细信息，请从下表中选择一个链接。

项目属性 - 常规
项目属性 - OPC DA
项目属性 - DDE
项目属性 - FastDDE/SuiteLink
项目属性-iFIX PDB 设置
项目属性 - OPC UA
项目属性 - OPC AE
项目属性 - OPC HDA
项目属性 - OPC .NET
项目属性 - ThingWorx

项目属性 - 常规

常规属性用于将标题和备注附加到项目以供参考，并可用于管理项目的安全设置。尽管“标题”字段最多可支持 64 个字符的字符串，但是“说明”字段中没有实际限制。但是，将“说明”限制在字段可用区域内，可以缩短项目加载时间。

标识

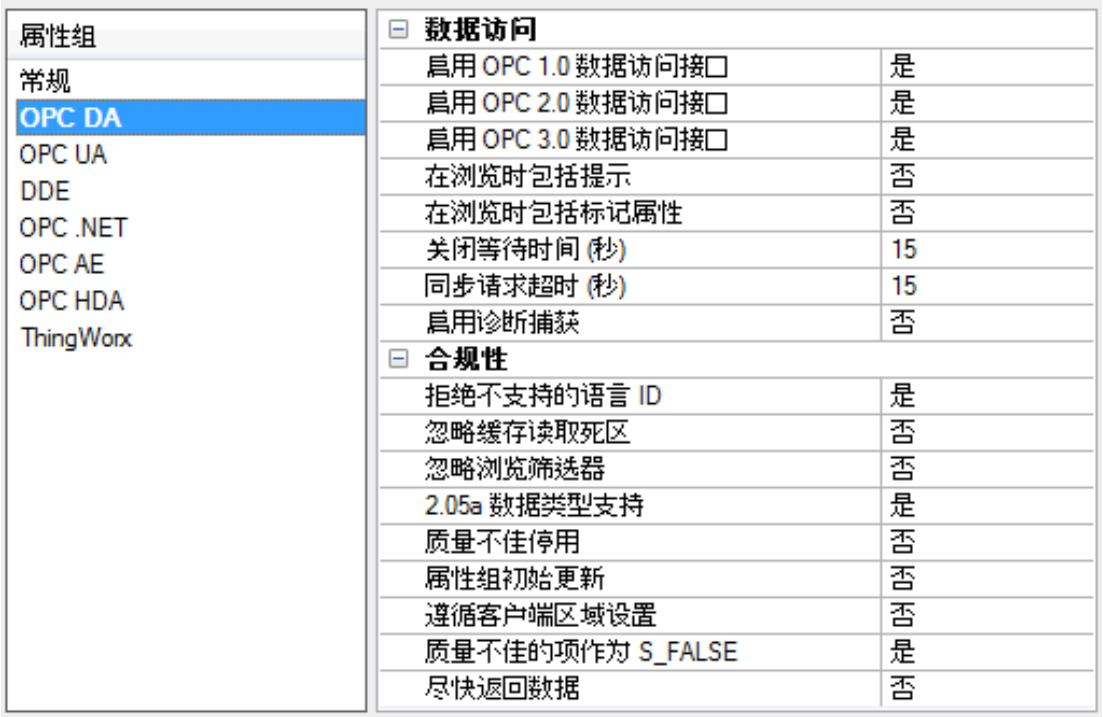
“说明”: 输入可选的短语，以帮助在报告和监视系统中标识此项目。

“标题”: 输入可选的字或短语，以在文件名称和报告中标识此项目。

“定义的标签”: 验证标记计数与此项目的数据集合预期 (和许可，如适用) 是否匹配。

“默认值”按钮用于将设置恢复到默认/预设值。

此服务器旨在实现与 OPC Foundation 规范的最佳兼容性。然而，在测试过程中发现，与规范完全兼容并使用所有 OPC 客户端应用程序，情况则有所不同。“OPC DA 合规”对话框可供用户自定义服务器的操作，以便更好地满足 OPC 客户端的需求。对于大多数 OPC 客户端应用程序而言，很少需要调整这些选项。



数据访问

通过参考“提示”可以快速了解如何处理特殊设备的数据。从 OPC 客户端输入动态标记时，这可能很有用。提示项不是有效的 OPC 标记。某些 OPC 客户端应用程序可能会尝试将提示标记添加到其标记数据库。发生此类情况时，客户端将接收到来自服务器的错误。尽管这可能会导致其他客户端停止自动添加标记或报告错误，但对大多数客户端来说，这并不是问题。可通过禁用提示来解决此问题。默认设置为禁用 (否)。

默认设置为禁用状态。

如果客户端应用程序未在超时之前返回，则服务器会关闭并退出。有效范围为 10 到 60 秒。默认设置为 15 秒。

如果同步操作未在超时之前完成，则服务器会强制操作完成，并导致客户端发生故障。这样可以防止客户端在使用同步操作时锁定。有效范围为 5 到 60 秒。默认设置为 15 秒。

注意：同步写入不会使用此属性设置，只有读取/请求才会使用此属性。

默认设置为禁用 (否)。

有关 OPC 数据访问 1.0、2.0 和 3.0 自定义规范的详细信息，请参阅 OPC Foundation 网站 www.opcfoundation.org。

合规性

如果 OPC 客户端应用程序尝试将 OPC 组添加到服务器中，但接收到一般故障，则可能是因为客户端向服务器提供的语言 ID 服务器本身不支持。如果发生这种情况，服务器将拒绝添加组。要解决此特定问题，请禁用兼容特性以强制服务器接受任何语言 ID。

对于某些 OPC 客户端，为死区传递正确的值可能会导致 OPC 客户端出现问题 (例如，即使其未频繁更新或完全不更新，也具有良好的数据)。这种情况很少见。在这种情况下，选项通常应处于默认禁用状态。

"质量不佳停用"：选择是服务器将返回失败，如果同步设备的一个或多个项读取中读取质量较差的结果。合规性要求服务器返回成功，表明即使一个或多个项目的数据的质量不佳且/或不确定，服务器也可以完成请求。

"属性组初始更新"：选择是服务器将返回所有未完成的初始项更新单个回调。禁用后，服务器会在初始更新可用时予以返回 (这可能会导致多次回调)。
启用此选项后，当将支持数据缓冲 (事件回放) 的驱动程序用于主动提供的设备协议时，可能会导致缓冲数据损失。如果出现缓冲数据损失问题，则应禁用合规性设置。

例如，如果使用英国公制，表示浮点数的字符串 1,200 将被转换为一千二百，但如果使用德国公制，则被转换为一又十分之二。如果在英文操作系统上运行德文软件，则用户需确定如何处理逗号。此设置可实现这样的灵活性。默认情况下，由于历史实现的原因，服务器会遵循操作系统的区域设置 ID。

Bad Quality Item as S_FALSE：选择是服务器将返回 S_FALSE 没有很好的质量的项目的项错误数组中。对于设置为完全合规性的现有项目，此设置的默认值为"是"，而对于未设置为完全合规性的现有项目，默认值为"否"。设置为"否"时，会发生之前的行为，即返回 E_FAIL (0x80004005)。

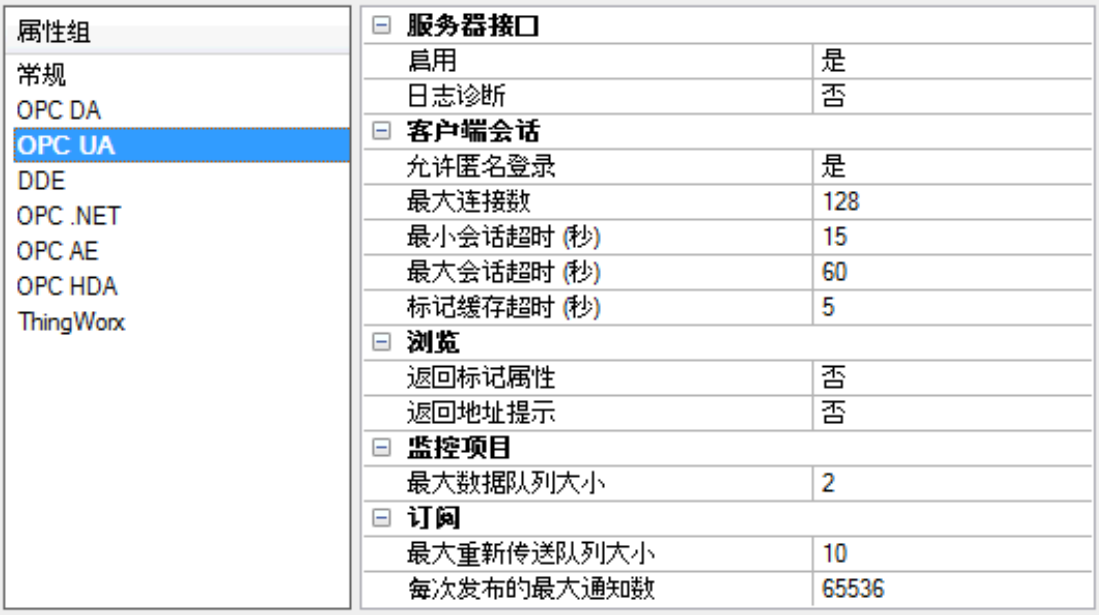
"尽快返回数据"：选择是显示要更新客户端的所有组。启用后，如果活动项目在值或质量方面发生改变，则会触发客户端更新。客户端指定的组更新速率用于设置添加到该组的项目的客户端请求扫描速率。默认设置为禁用状态。

"默认值"按钮用于将设置恢复到默认/预设值。

项目属性 - OPC UA

OPC 统一架构 (UA) 提供了一个平台无关的互操作性标准。它不是 OPC 数据访问 (DA) 技术的替代品：对于大多数工业应用，UA 补充或增强了现有的 DA 架构。OPC UA 组在服务器中显示当前的 OPC UA 设置。

注意：要更改设置，请单击特定属性的第二列。这将调用显示可用选项的下拉菜单。



服务器接口

- **"启用"**：启用后，UA 服务器接口将初始化，并接受客户端连接。禁用后，此页面上的其余属性将被禁用。
- **"日志诊断"**：启用后，会将 OPC UA 堆栈诊断记录到"事件日志"中。这只能用于故障排除目的。

客户端会话

- **"允许匿名登录"**：为了安全起见，此属性指定在建立连接时是否需要用户名和密码。默认设置为"否"，即不允许匿名访问且需要凭据。
  注意：如果禁用此设置，则用户无法作为用户管理器的默认用户 。用户可以以管理员身份登录，前提是 "用户管理器"中的默认用户登录 且该密码用于登录。
- **"最大连接数"**：指定所支持的最大连接数。有效范围为 1 到 128。默认设置为 128。
- **"最小会话超时"**：指定 UA 客户端建立会话的最小超时限制。可视应用程序需要更改值。默认值为 15 秒。
- **"最大会话超时"**：指定 UA 客户端建立会话的最大超时限制。可视应用程序需要更改值。默认值为 60 秒。
- **"标记缓存超时"**：指定标记缓存超时。有效范围为 0 到 60 秒。默认设置为 5 秒。
  注意：此超时控制在使用其完成 UA 客户端操作后，标记被缓存多长时间。在 UA 客户端以设定间隔读取/写入未注册的标记的情况下，用户可以通过增加超时来提高性能。例如，如果客户端每 5 秒读取一个未注册的标记，则标记缓存超时应设置为 6 秒。由于标记不必在每次客户端请求时重新创建，因此可以提高性能。

浏览

- **"返回标记属性"**：启用以允许 UA 客户端应用程序浏览地址空间中每个标记的标记属性。默认情况下禁用此设置。
- **"返回地址提示"**：启用以允许 UA 客户端应用程序浏览适用于每个项的地址格式化提示。虽然提示不是有效的 UA 标记，但某些 UA 客户端应用程序可能会尝试将其添加到标记数据库。发生此类情况时，客户端将接收到来自服务器的错误。这可能会导致客户端自动报告错误或停止添加标记。为了防止发生这种情况，请确保禁用此属性。默认情况下禁用此设置。

监控的项

- **"最大数据队列大小"**：指定一个项中将加入队列的数据通知的最大数量。有效范围为 1 到 100。默认设置为 2。
  注意：当监控项的更新速率高于订阅的发布速率时，使用数据队列。例如，如果监控项的更新速率为 1 秒，而订阅每 10 秒发布一次，则每 10 秒就会为该项发布 10 条数据通知。由于队列数据会占用内存，因此当内存出现问题时，应限制此值。

订阅

- **"最大重新传送队列大小"**：指定每次订阅时加入队列的发布的最大数量。有效范围为 1 到 100。值为零时将禁用重新传输。默认设置为 0。
  注意：订阅发布事件将按照客户端请求加入队列并进行重新传输。由于队列会占用内存，因此当内存出现问题时，应限制此值。
- **"每次发布的最大通知数"**：指定每次发布的最大通知数。有效范围为 1 到 65536。默认设置为 65536。
  注意：此值可能会通过限制从服务器发送到客户端的数据包大小来影响连接的性能。通常，大值应该用于高带宽连接，而小值应该用于低带宽连接。

"默认值"按钮用于将设置恢复到默认/预设值。

项目属性 - DDE

虽然该服务器的主要作用是一个 OPC 服务器，但有一些应用程序需要**动态数据交换 (DDE)** 才能共享数据。服务器将提供访问 DDE 应用程序的权限，这些应用程序支持以下 DDE 格式之一：**CF_Text**、**XL_Table** 和**高级 DDE**。CF_Text 和 XL_Table 是由 Microsoft 开发的标准 DDE 格式，与所有 DDE 感知应用程序一起使用。高级 DDE 是一种高性能格式，受到许多特定于工业市场的客户端应用程序的支持。

要使 DDE 接口与服务器相连，必须允许运行时与桌面交互。有关详细信息，请参阅如何...允许桌面交互。

要通过"配置"访问 DDE 服务器设置，请单击**"编辑"|"项目属性"**，然后定位 **DDE**属性。其属性可用于调整 DDE 操作以满足应用程序的需求。

常规

"启用 DDE 与服务器的连接"：此属性可确定是否启用服务器的 DDE 服务器部分。如果禁用 DDE 操作，则服务器不会响应对 DDE 数据的任何请求。如果要将服务器仅作为 OPC 服务器使用，用户可能要禁用 DDE 操作。这样做可以提高数据安全性并改进服务器的整体性能。默认情况下，DDE 处于禁用状态。

另请参阅：如何...在服务器上使用 DDE

"服务名称"：当服务器用作 DDE 客户端上的应用程序名称时，此属性可让用户更改其呈现方式。此名称最初设置为能够与先前版本的服务器兼容。但是，如果用户需要替换现有 DDE 服务器，可以更改服务器的服务名称，以与所替换的 DDE 服务器匹配。可输入 1 至 32 个字符的字符串作为服务名称。

格式

此属性允许用户配置为客户端应用程序提供的 DDE 格式。选择启用或禁用**"高级 DDE"**、**"XL 表格"**和 **CF_Text**。默认情况下启用全部三种格式。若用户在将 DDE 客户端应用程序连接服务器时遇到问题，此功能尤为有用：可禁用各个 DDE 格式，以将特定格式从测试中隔离。

注意：每个 DDE 感知应用程序必须至少支持 CF_Text。

定时

"客户端更新间隔"：此间隔设置可用于批处理 DDE 数据，以便将其传送到客户端应用程序。使用 DDE 格式时，只有服务器的大型数据块可通过单个 DDE 响应发送时，性能才能得以提高。为了提高服务器收集大型数据块的能力，可以设置更新计时器，以在将数据发送到客户端应用程序之前累积一组新的数据。更新计时器的有效范围是 20 至 60000 毫秒。默认设置为 100 毫秒。

"DDE 请求超时"：此属性可为 DDE 请求的完成配置超时。如果无法在指定超时内完成服务器上的 DDE 客户端请求 (读取或写入操作)，将会有一个错误返回 DDE 客户端。有效范围为 1 到 30 秒。默认设置为 15 秒。

注意：服务器运行时可能需要重新初始化，以使更改生效。

项目属性 - OPC .NET

要通过"配置"访问 OPC .NET 服务器设置，请单击**"编辑"|"项目属性"**，然后选择 **OPC .NET**选项卡。

"启用"：当启用时，OPC .NET 包装器将初始化，并接受客户端连接。

提示：

1. OPC .NET 包装器作为 "xi_server_runtime.exe" 系统服务运行。其仅在服务器处于启动状态且上述选项启用时启动。与 OPC DA 不同，客户端无法启动服务器。
2. 要安装和使用 OPC .NET，在安装服务器之前，计算机上必须存在 Microsoft .NET 3.5。

"默认值"按钮用于将设置恢复到默认/预设值。

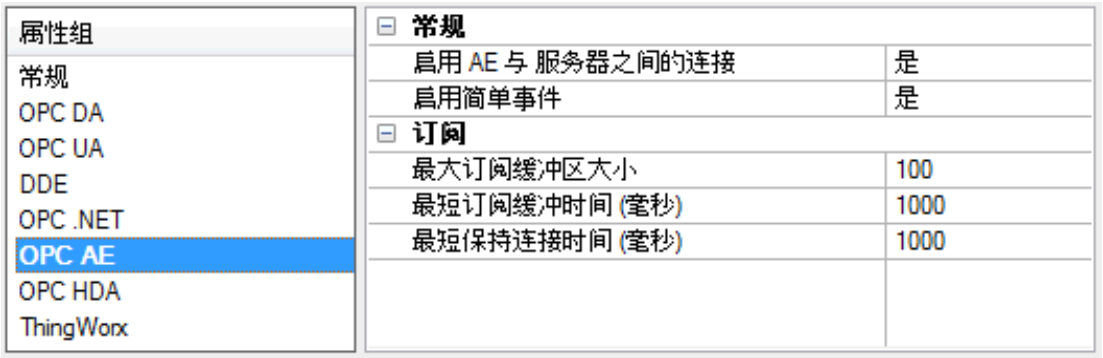
项目属性 - OPC AE

事件用于通过信号传递服务器中具体值，这与 OPC 数据访问中的数据更新类似。OPC AE 功能可供用户从服务器接收简单事件，包括系统启动和关闭消息、警告、错误等。这些事件会显示在"事件日志"中。

OPC AE 组用于指定多个项目级 AE 设置。对这些设置所做的更改会在 A&E 客户端与服务器断开连接后生效。

Alarms & Events 插件可供 Alarms & Events (A&E) 客户端从 OPC 服务器接收 A&E 数据。它用于将 OPC 服务器事件转换成 A&E 格式，并使用 OPC 服务器标记创建自定义警报。

有关详细信息，请联系 OPC 供应商。



常规

"启用 AE 与 服务器之间的连接"：此属性可开启或关闭 OPC AE 服务器。

"启用简单事件"：启用后，客户端即可使用简单事件。禁用后，会发送事件。默认设置为启用状态。

订阅

"最大订阅缓冲区大小"：用于指定在一个发送调用中发送给客户端的最大事件数。范围是 0 到 1000。默认设置为 100。0 代表无限制。

"最短订阅缓冲区时间"：用于指定向客户端发送两次调用之间的最短时间间隔。支持的范围为 100 到 60000 毫秒。默认设置为 1000 毫秒。

"最短保持连接时间"：用于指定服务器向客户端发送保持连接消息之间的最短时间。支持的范围为 100 到 60000 毫秒。默认设置为 1000 毫秒。

"默认值"按钮用于将设置恢复到默认/预设值。

项目属性 - FastDDE/SuiteLink

Wonderware 公司的 FastDDE 和 SuiteLink 服务器支持，简化了将服务器连接到 FactorySuite 应用程序所需的工作。Wonderware 连接工具包可用于同时提供 OPC 和 FastDDE/SuiteLink 连接，且无需使用中间桥接软件即可快速访问设备数据。

要使 FastDDE 接口与服务器相连，必须允许运行时与桌面交互。有关详细信息，请参阅如何...允许桌面交互。

注意：为了使 FastDDE/SuiteLink 能够正确操作 (同时使该选项卡在项目属性中可用)，必须在 PC 上安装 Wonderware FS2000 Common Components 或 InTouch Runtime Component 版本 8.0 或更高版本。

Property Groups		
General	**General**	
OPC DA	Enable FastDDE/SuiteLink connections to the server	Yes
OPC UA	Application name	server_runtime
DDE	**Timing**	
FastDDE/SuiteLink	Client update interval (ms)	100
OPC AE		
OPC HDA		
ThingWorx		

"启用 FastDDE/SuiteLink 与服务器的连接"：该属性可启用或禁用对客户端/服务器协议的支持。在计算机上安装 Wonderware 产品时，默认情况下启用此设置。如果禁用 FastDDE/SuiteLink 操作，则服务器不会响应 FastDDE 或 SuiteLink 数据的任何请求。为了更好的性能和安全，建议如果服务器仅用于 OPC 连接的情况下禁用此设置。

"应用程序名称"：此图标用于打开应用程序的名称。默认设置为 "server_runtime"。
注意：可对此名称进行自定义，以满足特定最终用户的需求。例如，安装期间选择"移除与重定向"的用户必须更改 "servermain" 的设置才能让某些 FactorySuite 应用程序在不经修改的情况下工作。

"客户端更新间隔 (毫秒)"(Client Update Interval (ms))：用于打开新数据发送到 FastDDE/SuiteLink 客户端应用程序的频率设置的图标。其范围为 20 至 32000 毫秒。默认设置为 100 毫秒。通过计时器可对 FastDDE/SuiteLink 数据进行批处理以便传输到客户端应用程序。使用 FastDDE 或 SuiteLink 之类的客户端/服务器协议时，只有大型服务器数据块可以在单个响应中发送才能提高性能。为了提高服务器收集大型数据块的能力，可以设置更新计时器，以在将数据发送到客户端应用程序之前累积一组新的数据。
注意：更新速度适用的是向客户端应用程序发送数据的频率，而不是从设备读取数据的频率。可以使用扫描速率来调整服务器从所连接设备获取数据的快慢程度。有关详细信息，请参阅标记属性 - 常规。

注意：服务器运行时可能需要重新初始化，以使更改生效。

"默认值"按钮用于将设置恢复到默认/预设值。

项目属性 - iFIX PDB 设置

****对于计数器值和 32 位计数器值，驱动程序可以容纳最大范围。要进行访问，请单击**"编辑"|"项目属性"**。

注意： 如果在计算机上安装了 iFIX，在工程属性中仅显示 iFIX PDB 设置。

在某些情况下，必须将"进程模式"设置为"系统服务"，才能使 iFIX PDB 接口适用于运行时。有关详细信息，请参阅 处理模式。

Property Groups		
General	**General**	
OPC DA	Enable connectivity to iFIX PDB	Yes
OPC UA	Enable latched data	No
DDE	Enable update per poll	No
iFIX PDB Settings	Use iFIX startup configuration file	Yes
OPC AE	Use unconfirmed updates	No
OPC HDA	**Timing**	
ThingWorx	PDB-to-server request timeout (s)	5
	Deactive tags on PDB read inactivity	Yes
	Inactivity timeout (s)	15

注意：建议用户保留每个字段的默认值。用户还应确保设置满足应用程序的需求。

常规

“启用与 iFIX PDB 的连接”：启用或禁用对客户端/服务器协议的支持。如果禁用 iFIX PDB 操作，则服务器不会响应 iFIX PDB 数据的任何请求。为了更好的性能和安全性，当服务器仅用于 OPC 连接时，请禁用此属性。

“启用锁存的数据”：通常，如果发生通信故障，iFIX 应用程序的数据链路会显示一系列问号 (如 "????")。然而，用户可能希望始终显示一个值。通过启用锁定的数据，在屏幕上保留成功读取的最后一个值。默认设置为启用状态。

注意：AR 和 DR 块不支持数据锁存。

“启用每次轮询时更新”：启用后，服务器会在驱动程序每次轮询设备时将当前值、质量和时间戳传递给 iFIX。禁用时，服务器只有在确定值或质量有所更改时才向 iFIX 传递更新。默认设置为禁用状态。

注意：此设置是动态的，这意味着在应用该选项后，服务器将立即开始以设备扫描速率向 iFIX 客户端传递更新。

“使用 iFIX 启动配置文件”：启用通过 iFIX 创建此文件以包含 iFIX 客户端访问的所有项目。在 iFIX 请求项目数据之前，它将自动开始扫描项目。默认设置为启用状态。

另请参阅：IFIX 应用程序的项目启动

“使用未确认的更新”控制服务器如何在通过 NIO 接口进行写入操作后更新 iFIX 的本地缓存。采用默认设置 (禁用) 时，服务器不会更新本地缓存，直到通过读取确认值为止。对于大多数应用程序，默认设置可从数据完整性角度提供最佳的用户体验。对于利用 iFIX 轻松数据库访问 (EDA) 的应用程序，用户可能希望启用未确认的更新，以便随着尝试写入的值立即更新 iFIX 的本地缓存。

注意：从数据完整性的角度来看，使用未确认的更新可能导致 iFIX 中显示写入成功和数据不准确的错误指示。使用未确认更新的另一个后果是，由于暂时未确认的更新 (尝试写入值) 以及之后确认的更新 (项目的实际读取值)，iFIX 中显示的数据可能会“闪烁”。

定时

“PDB 至服务器请求超时”(PDB-to server request timeout(s))：指定 iFIX PDB 在添加、删除、读取或写入请求响应之前需要等待的时间，超过此时间即会视为超时。一旦超时，该请求将以服务器的名义被丢弃。如果服务器正在处理其他请求或者在服务器已失去与 iFIX PDB 通信，则会出现超时。在通信丢失的情况下，iFIX PDB 将自动重新建立与服务器的通信，从而不会发生连续超时。有效范围为 5 到 60 秒。默认设置为 5 秒。

“PDB 读取不活动时取消激活标记”：指示服务器自动取消激活指定时间内未被 iFIX 读取的标记。这样可减少不必要的进程硬件轮询。启用后，服务器每 15 秒读取一次标记列表，并停用任何空闲标记。如果 iFIX 不具有指定时间段内执行标记的读取的请求，标记被视为空闲。由于服务器每 15 秒检查一次空闲标记，因此在上次读取标记后，服务器可能不会恰好在这个时间点将标记设为非活动状态；最长可能延迟 15 秒，具体取决于上次读取在检查周期内发生的时间点。如果 iFIX 请求来自先前已停用的标记中的数据，服务器将重新激活该标记，并恢复轮询硬件。默认设置为禁用状态。但是，启用此功能后，默认设置将适用于所有项目。用户可指定空闲时间，范围介于 15 到 607999 (15 秒至 1 周) 之间。

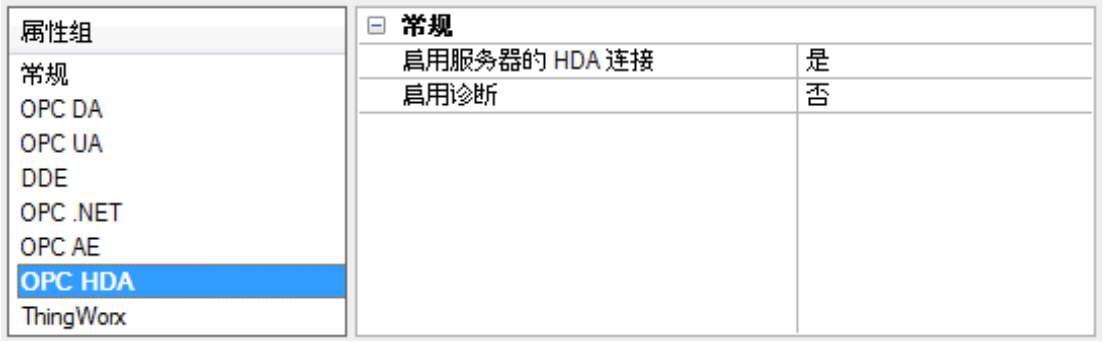
此功能是用于仅注册标记，但可能会导致非注册标记关闭扫描。为避免使用此功能时发生上述情况，请将非活动状态计时器设置为大于 iFIX 数据库中配置的最长扫描时间。

“不活动超时”(Inactivity timeout(s))：指定 iFIX pdb 在变为活动状态前需要等待的时间，超过此时间即会视为超时。在通信丢失的情况下，iFIX PDB 将自动重新建立与服务器的通信，从而不会发生连续超时。有效范围为 5 到 60 秒。默认设置为 5 秒。

“默认值”按钮用于将设置恢复到默认/预设值。

项目属性 - OPC HDA

要通过“配置”访问 OPC HDA 服务器设置，请单击**“编辑”|“项目属性”**，然后展开 **OPC HDA**组。



"启用 HDA 与服务器的连接"：当启用时，HDA 客户端可以连接到通过此服务器公开的 HDA 服务器。禁用时，客户端 HDA 连接将被禁用。无需重新启动"运行时"即可应用这些设置；但是，尽管服务器不会删除连接的客户端，但它也不再接受新的客户端连接。默认设置为启用状态。

"启用诊断"：当启用时，此选项允许在用于存储的"事件日志"服务中捕获和存储 OPC HDA 数据。默认设置为禁用状态。

注意：启用诊断对服务器运行时性能有负面影响。有关事件日志记录的详细信息，请参阅 OPC 诊断查看器。

"默认值"按钮用于将设置恢复到默认/预设值。

项目属性 - ThingWorx

在根据需要同时允许 OPC 和其他连接时，对 ThingWorx 本机接口的支持会简化与 ThingWorx Platform 的连接任务。

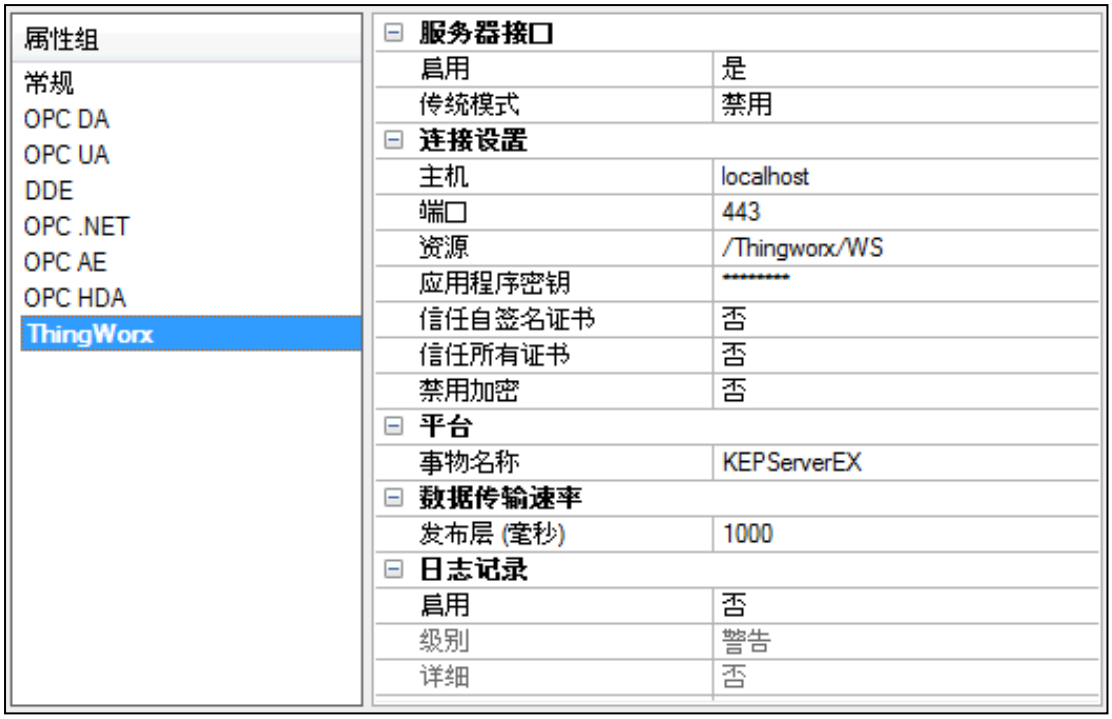
在配置 ThingWorx 本机接口之前，请在 ThingWorx Platform 中使用"工业网关"事物模板创建"事物"。建议使"事物"名称能够表明其数据源。创建"事物"后，将 OPC 服务器配置为使用"事物名称"连接 ThingWorx Platform。新的连接将自动绑定到此"事物"。

连接 ThingWorx Platform 后，使用左侧菜单中的"工业连接"选项从新创建的 OPC 服务器实例中选择标记。可以选择这些标记，并直接将其绑定到 ThingWorx Composer 中的新"事物"。

有关详细信息，请参阅 ThingWorx Composer 文档。

注意：

- 在 ThingWorx 平台中，必须将数据类型为数组的所有标记配置为"始终"推送类型。设置为值更改的推送阈值将无法向 ThingWorx 平台发布更新。
- 大多数的本机接口功能在客户端服务器配置中，而 ThingWorx 本机接口则更像一个客户端，它会创建到 ThingWorx 平台的出站连接。这将允许 ThingWorx 本机接口使用标准的端口和协议来连接到远程 ThingWorx 平台，而无需创建特殊的防火墙或路由规则。只要可以从运行 OPC 服务器的计算机的浏览器访问 ThingWorx Composer，OPC 服务器就能够通过本机接口将数据传递到该平台。
- 如 ThingWorx 文档所述，"ThingWorx 应用程序密钥"的配置对于提供安全的环境至关重要。应用程序密钥应提供适当的权限，以便在服务器实例和 ThingWorx Platform 之间进行正确的数据交换。



"服务器接口"

"启用"：在 ThingWorx 本机接口选择"是"来尝试与提供的信息进行连接。

"旧模式"：连接 Thingworx 新一代 Composer (v7.4 或更高版本) 时，应禁用旧式模式 (默认)。如需连接 ThingWorx 7.3x 或更低版本，请选择"启用"以强制使用 RemoteKEPServerEXThing 模板形态。要正确使用此模式，请将 RemoteKEPServerEXThing 扩展导入到 ThingWorx 实例上。然后，正确的 ThingShapes 和服务定义会使用此本机接口。

- 旧模式仅用于现有的 KEPServerEX V6.0 ThingWorx 本机接口项目和 ThingWorx Composer (v7.3x 或更低版本)。

"连接设置"

"主机"：指定 ThingWorx 服务器的 IP 地址或 DNS 名称。

"端口"指定 ThingWorx 服务器所使用的 TCP 端口号。

"资源"：ThingWorx 服务器上指定的 URL 端点。

"应用程序密钥"：为连接到 ThingWorx 服务器而输入或粘贴到身份验证字符串中的密钥。

"信任自签名证书"：为确保最高安全级别，请选择"否"。选择"是"可在开发过程中接受自签名证书。
- **警告**：在生产环境中，请勿将此设置为"是"，否则会危及安全。

"信任所有证书"：为确保最高安全级别，请选择"否"。选择"是"则 TLS 库将不会验证服务器证书。
- **注意**：在生产环境中，请勿将此设置为"是"，否则会危及安全。

"禁用加密"：指示是否允许连接至非 SSL 加密的 ThingWorx Platform。
- **注意**：在生产环境中，请勿将此设置为"是"，否则会危及安全。

"平台"

"事物名称"：输入 ThingWorx 服务器上表示此数据源的实体 (远程事物) 的名称。使用 OPC 服务器模板创建远程事物。

OPC 服务器模板：输入字符串用作此服务器实例的标识符。此属性仅在"传统模式"下可用。

"数据传输速率"

"扫描速率": 指定扫描项时的默认频率，单位为毫秒。若为零，则所有标记的扫描率均设为服务器中的标记指定扫描率，除非 ThingWorx 平台的 AddItems 服务传入了具体的扫描率。此属性仅在"传统模式"下可用。

Send every scan： 选择是更新 ThingWorx 在每次扫描，而不是仅在数据更改时。ThingWorx 中的属性必须设置为 **按类型始终推送**，这是默认情况下，此设置，使其生效。此属性仅在"传统模式"下可用。

Publish floor： 指定对平台发送更新的最小速度。零尽可能频繁发送更新。

日志记录

"启用": 选择"是"可激活 ThingWorx 本机接口的高级日志记录。此日志记录发送到服务器事件日志。此日志记录可能会导致事件日志迅速填满，建议在不进行故障排除时将其保持为禁用状态。

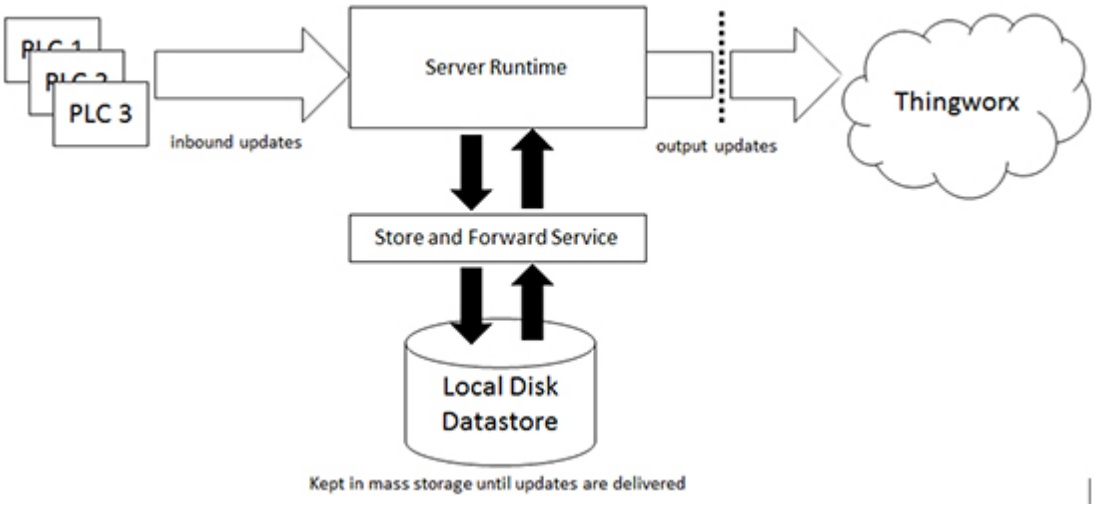
"级别"：选择发送到事件日志的日志记录的严重程度。"追溯"包括本地 ThingWorx Interface 中的所有消息。

"详细信息"：选择**"是"**以生成尽可能详细的错误消息。

另请参阅：事件日志、事件日志选项

"存储并转发"

当工业服务器与 ThingWorx 平台的连接断开时，ThingWorx 本机接口可通过"存储并转发"数据存储来持久保留属性更新。启用后，"存储并转发"功能将所有传入的属性更新持久保留到磁盘，直到 ThingWorx 本机接口收到平台已接收更新的确认。如果与平台的连接断开，所有更新都存储并保留在磁盘上，直到存储更新的磁盘空间达到 500 MB，或者存储的更新大小超出指定的上限，两者中取较小者。一旦数据存储或磁盘已满，传入的更新将被丢弃，直到有足够的空间可存储传入数据。



另请参阅：完成率示例

存储并转发属性

"启用" ：选择**"是"** 将数据保存到本地磁盘目录中，以避免在连接中断或大量数据传输时丢失数据。如果启用此设置，将允许对数据进行排队，并在连接建立且数据接收确认后转发数据。

"存储位置": 输入或浏览数据缓存目录的完全限定路径。

注意：当"存储并转发"数据存储无法被初始化时，ThingWorx 本机接口会将更新存入内存中排队等候。服务器会自动重试，直到数据存储可以被初始化为止。*有关特定失败信息，请参阅事件日志。*

"数据存储上限"：选择清除之前允许的最大数据量 (MB 或 GB)。可用的数据存储大小范围为 128 MB 到 16 GB。

"转发模式"：选择一种方法来确定在连接恢复时将哪些更新发送到 ThingWorx。如需对生产数据进行有效监控，要求即使在与平台的连接断开时也不会出现任何数据丢失，可在重新连接时"存储并转发"，或者安排对生产未得到有效监控 (例如，在生产停机时间) 期间所存储的更新进行转发。选项包括"活动"和"暂停"：

- **活动模式** - 当将 "转发模式"设置为"活动"时，会按照接收到的顺序依次发送存储的属性更新，直到 ThingWorx 平台收到所有更新为止。更新随即被实时发送到平台。当在 ThingWorx 平台断开连接期间收集了很多更新时，由于属性更新转发遵循先进先出的原则，属性更新可能会延迟。

- **暂停模式** - 当将 "转发模式"设置为"暂停"时，当从断开连接恢复后，仅会将最新的更新发送到平台。这可确保 ThingWorx 应用程序能够对生产进行有效监控并获得最新的数据。当未对生产进行有效监控，可将模式设置为"活动"，以开始转发在服务器与平台断开连接时所存储的较旧的更新。在将属性更新存储到磁盘中之前，工业服务器可在内存中缓存多达 25,000 个属性更新。一旦达到了 25,000 个的更新数上限，便会将属性更新推送到磁盘并保存在其中，直到将"转发模式"设置为"活动"为止。这样，工业服务器便可在恢复与 ThingWorx 平台的连接时，优先处理最新的 25,000 个更新，对更新进行保留，以免这些更新丢失，并稍后对其进行转发。当达到数据存储大小上限或磁盘中存储的内容超过 500MB 的上限时 (以先发生者为准)，就会丢弃新的更新。通常只有当与 ThingWorx 平台的连接丢失时，内存中的缓冲区才会被存满；然而，当属性更新的收集速度超过可转发到平台的速度时，也可能出现这种情况。

存储并转发注意事项

- 目前可通过工业服务器的配置工具或配置 API 来进行"存储并转发"配置。
- 默认情况下，"存储并转发"处于禁用状态，必须在工业服务器的项目属性中或通过配置 API 来启用它。
- 没有必要从 ThingWorx 平台配置"存储并转发"。但是，要将转发的更新存储到 ThingWorx 平台中，必须配置"值流"，并为历史记录所需的所有属性启用"日志记录"。
- 当修改了 (在"存储位置"设置中定义的) 数据存储路径配置时，现有的数据存储仍将保留在磁盘上。如果数据存储路径配置得以恢复，与当前项目相关联的更新将被转发到平台。
- 更改"存储并转发"属性无需重新初始化平台连接。在应用更改时，Thingworx 接口会继续收集更新。

在配置和运行时均会对"存储并转发"路径进行验证，该路径必须符合以下要求：

- 必须介于 3 至 256 个字符之间
- 不得包含系统禁止的任何字符或符号
- 必须为绝对路径 (以盘符开头)
- 不得引用网络资源 (映射驱动器 * 或 UNC 共享)
- 不得引用可移动媒体，例如 U 盘 *
  ** 指的是仅在运行时验证的项*

可通过以下方式访问"存储并转发的状态和监控"

- 工业服务器的"存储并转发"标记
- 工业服务器的事件日志

存储并转发操作注意事项

- 启用"存储并转发"后，其可靠性要求会使性能略微降低，因为所有的更新在被发送到 ThingWorx 平台之前都要经过磁盘缓冲区，并且 ThingWorx 本机接口要在收到平台已接收最新更新集的确认之后，才会发送下一组更新。
- 服务器重启后，存储的更新仍然存在。
- 确保在软件升级前转发所有存储的更新，因为在进行主要/次要服务器升级后，将无法保留更新。

操作 - 传统模式

旧模式仅用于现有的 KEPServerEX V6.0 ThingWorx 本机接口项目和 ThingWorx Composer (v7.3x 或更低版本)。

若接口配置了有效信息且为启用状态，KEPServerEX 会建立与 ThingWorx Platform 的连接。在 ThingWorx Platform 上，必须创建与在项目属性中使用的“事物名称”相同的新“事物”。RemoteKEPServerEXThing 事物的扩展名必须导入到 ThingWorx Platform 中并用于创建新的集成的事物。在平台上创建事物后，可通过平台调用以下服务。

请参阅 ThingWorx 帮助指南，了解有关导入扩展名的说明。

注意：RemoteKEPServerEXThing 扩展名可在在线的 ThingWorx Marketplace 或下列文件夹中找到：

- 64 位 Windows：C:\Program Files (x86)\Kepware\KEPServerEX 5\Utilities\KEPServerEX Extension for the ThingWorx IoT Platform
- 32 位 Windows：C:\Program Files\Kepware\KEPServerEX 5\Utilities\KEPServerEX Extension for the ThingWorx IoT Platform

“浏览组”：返回通道设备和标记组的列表。可在其中输入筛选器和路径。筛选器与包括字符列表的 OPC 筛选器相同。路径是通道和设备的列表，如 "Channel1.Device1"。

“浏览项目”：返回特定路径下的标记列表。可在其中输入筛选器和路径。筛选器与包括字符列表的 OPC 筛选器相同。路径是通道和设备的列表，如 "Channel1.Device1"。

“添加项”：允许订阅标签，并将其作为属性添加到事物中。需要一个信息表来调用此服务。信息表必须包含以下信息。只读：布尔型、扫描速率 MS (可选)：整数，说明 (可选)：字符串、基本类型：ThingWorx 数据类型、源类型：KEPServer 数据类型、持续：布尔型、登录：布尔型、源：标记地址 (通道.设备.标记)、名称：标记的本地名称。
注意：根据 ThingWorx 限制，在名称字段中不得出现空格、特殊字符和先行数字。在名称中或名称末尾可使用连字符、下划线和数字。

“移除项”：从标记中删除订阅。需要一个信息表来调用此服务。信息表必须包含以下信息。名称：标记的本地名称。也可以启用 ForceRemove 布尔型解除属性中的标记而不删除属性。

“获取配置”：返回信息表，包含以毫秒为单位的扫描速率、服务器说明和以毫秒为单位的发布层。

“设置配置”：设置以毫秒为单位的扫描速率、服务器说明和以毫秒为单位的发布层。如将值留为空白，则保留其当前设置。

注意：

1. 当使用日期数据类型，来自 KEPServerEX 的值被解释为 UTC。允许适当的时区偏移。
2. 将项添加到服务器是一个同步过程，且完成速度很快。平台的“自动绑定”属性可能需要一些时间，且发生在项已经添加完成的情况下。自动绑定进程完成并产生绑定进程结果时将触发事件。
3. 调用“移除项”将仅从属性中移除绑定。调用“移除项”后，可通过编程方式或通过 Composer 将新的标记重新绑定到该属性，或者从 Composer 上删除该属性。这些属性将以一个空白的“远程属性名称”显示在 Composer 上，直到重新绑定或删除属性。
4. 同时添加多个项时，如果两个或多个项目被配置为使用相同的 ThingWorx 名称，则整个“添加项”调用将失败。请确保所有属性都具有唯一的名称。
5. 示例中的命令可通过 cURL 或其他 POST/PUT/GET 实用程序来实现。以上仅为一部分示例；有关与所有可用服务进行交互的信息，请参阅 ThingWorx 文档。

“默认值”按钮用于将设置恢复到默认/预设值。

ThingWorx 示例 - 传统模式

<- -> 之间的文本必须被替换为合适的信息。

应通过所有 API 调用发送以下标题：

```
Headers:
Accept=application/json-compressed
Content-Type=application/json
appKey=<-AppKey->
POST or PUT commands:
AddItem
```

URL:
https://<-URL or IP->/Thingworx/Things/<-ThingName->/Services/AddItems
Body:
{"items":{"description":"","name":"Infotable","dataShape":
{"fieldDefinitions":{"ReadOnly":{"name":"ReadOnly","aspects":
{},"description":"ReadOnly","baseType":"BOOLEAN","ordinal":0},"ScanRateMS":
{"name":"ScanRateMS","aspects":
{},"description":"ScanRateMS","baseType":"INTEGER","ordinal":0},"Description":
{"name":"Description","aspects":
{},"description":"Description","baseType":"STRING","ordinal":0},"BaseType":
{"name":"BaseType","aspects":
{},"description":"BaseType","baseType":"STRING","ordinal":0},"SourceType":
{"name":"SourceType","aspects":
{},"description":"SourceType","baseType":"STRING","ordinal":0},"Persistent":
{"name":"Persistent","aspects":
{},"description":"Persistent","baseType":"BOOLEAN","ordinal":0},"Logged":
{"name":"Logged","aspects":
{},"description":"Logged","baseType":"BOOLEAN","ordinal":0},"Source":
{"name":"Source","aspects":
{},"description":"Source","baseType":"STRING","ordinal":0},"Name":
{"name":"Name","aspects":
{},"description":"Name","baseType":"STRING","ordinal":0}},"name":"KEPAddItems","descr-
iption":""},"rows":
[{"ReadOnly":<-true or false->,"ScanRateMS":<-Rate in Milliseconds-
>,"Description":"<-Optional Description->","BaseType":"<-ThingWorx DataType-
>","SourceType":"<-KEPServerEX DataType->","Persistent":<-true or false->,"Logged":<-
true or false->,"Source":"<-path to tag on KEPServerEX->","Name":"<-name in
ThingWorx->"}]}}
RemoveItem
URL:
https://<-URL or IP->/Thingworx/Things/<-ThingName->/Services/RemoveItems
Body:
{"items":{"description":"","name":"Infotable","dataShape":
{"fieldDefinitions":{"Name":{"name":"Name","aspects":
{},"description":"Name","baseType":"STRING","ordinal":0}},"name":"KEPItemNames","desc-
ription":""},"rows":
[{"Name":"<-name in ThingWorx->"}]},"forceRemove":<-true or false->}
BrowseGroup
URL:
https://<-URL or IP->/Thingworx/Things/<-ThingName->/Services/BrowseGroups
Body:
{"path":"<-Path->","filter":"<-Optional Filter->"}
BrowseItems
URL:
https://<-URL or IP->/Thingworx/Things/<-ThingName->/Services/BrowseItems
Body:
{"filter":"<-Optional Filter->","path":"<-Path->"}
GetConfiguration
URL:
https://<-URL or IP->/Thingworx/Things/<-ThingName->/Services/GetConfiguration
Body:
{}
SetConfiguration
URL:
https://<-URL or IP->/Thingworx/Things/<-ThingName->/Services/SetConfiguration
Body:
{"ScanRateMS":<-Rate in Milliseconds->,"ServerDescription":"<-Server Description-
>","PublishFloorMS":<-Rate in Milliseconds->}
PUT Command
设置值：
URL:
https://<-URL or IP->/Thingworx/Things/<-ThingName->/Properties/*
Body:
{"<-ThingWorx Name->":<-Value->}

```
GET commands
Get value:
https://<-URL or IP->/Thingworx/Things/<-ThingName->/Properties/<-ThingWorx Name->
Get all property values:
https://<-URL or IP->/Thingworx/Things/<-ThingName->/Properties/
```

存储并转发 - 完成率示例

需要考虑所存储更新的**数据存储上限**和数据类型，以确定最大更新计数和完成率。下表描述了多个数据类型场景的更新计数限制和完成率 (假设数据存储上限为 128 MB、每秒更新 1 次)。

数据类型	最大更新计数	完成率 (字节/秒)
字/短整型	5817792	22
双字型/长整型/浮点型	5333076	24
双精度	4571321	28
字符串 (长度 = 10)	3764743	34

使用以下方程和上表中的信息，可将与项目的标记数据类型相对应的完成率加在一起求和，以确定给定项目的完成率：

总完成率 =
扫描速率 (秒) *
属性计数 (布尔型) * 完成率 (布尔型) +
属性计数 (字) * 完成率 (字) +
属性计数 (字) * 完成率 (短整型) +
属性计数 (双字型) * 完成率 (双字型) +
属性计数 (字) * 完成率 (长整型) +
属性计数 (字) * 完成率 (浮点型) +
属性计数 (双精度) * 完成率 (双精度) +
属性计数 (字符串) * 完成率 (字符串)

下表描述了示例项目数据丢失前的完成率和离线时间，该项目包含多个扫描速率 (假设数据存储上限为 128 MB) 的 500 个字属性、500 个双字型属性、10 个字符串属性和 100 个双精度属性。

每个属性的扫描速率 (毫秒)	完成率 (字节/秒)	离线时间 (分钟)
10000	2614	816
1000	26140	81
250	104560	20

存储并转发 - 系统标记

系统标记用于提供数据存储状态信息及允许服务器客户端管理更新。仅当启用“存储并转发”后，服务器客户端才可以使用这些系统标记。这些标记位于客户端浏览树中的 _System 文件夹的同级 _ThingWorx group 文件夹下。

标记	类	数据类型	说明
_StoreAndForwardEnabled	读/写	布尔型	此标记可启用或禁用“存储并转发”。当此标记设置为 False 时，将禁用“存储并转发”。当“存储并转发”被禁用时，所有数据存储相关的系统标记均会报告相当于 0 的默认值。 **注意：**

标记	类	数据类型	说明
			• 配置并非总能指示“存储并转发”的“启用/禁用”状态。可使用 _StoreAndForwardEnabledStatus 系统标记来获取正在使用的配置。例如，当发生阻止“存储并转发”的错误时，_StoreAndForwardEnabledStatus 会返回 0。
_StoreAndForwardEnabledStatus	读/写	布尔型	此标记用于指示接口是否正在使用“存储并转发”。
_DatastoreDiskFull	只读	布尔型	此标记用于指示数据存储用的磁盘中存储的内容是否超过存储更新所需的 500 MB 的阈值。
_DatastoreFull	只读	布尔型	此标记用于指示数据存储是否已达到已配置的可用于存储更新的**数据存储上限**。
_StoredUpdateCount	只读	双字型	此标记用于指示数据存储中的更新数量。 **注意：** • 非零值并不能表示 ThingWorx 连接已丢失，因为只要启用了“存储并转发”，更新便一直会经由数据存储。 • 在稳态运行过程中，该数字预计会有波动；然而，已存储的更新计数不应随着时间而增加。此行为表示所收集的数据量超过了可传递到 ThingWorx 平台的数据量。
_DeleteStoredData	读/写	布尔型	此标记可用于删除数据存储的内容。将任何值写入此标记中均会删除“存储并转发”数据存储中所存储的全部更新。
_DatastoreCurrentSizeMB	只读	双精度	此标记用于报告当前在磁盘上的全部更新所占用的空间量 (MiB)
_DatastoreRemainingSpaceMB	只读	双精度	此标记用于报告数据存储中剩余的可用于存储更新的空间量 (MiB)。其基于是“数据存储上限”属性，而非可用磁盘空间。*有关剩余磁盘空间的信息，请参阅 _DatastoreUsableDiskSpace 标记。*
_DatastoreUseableDiskSpaceMB	只读	双精度	此标记用于报告数据存储所在的磁盘中可用于存储更新的空间量 (MiB)。“存储并转发”使用了 500MiB 的安全缓冲区，以免存满整个磁盘。此系统标记在计算时会将此安全缓冲区考虑在内。此标记不会根据用户的指定来反映数据存储中的剩余空间量。*有关详细信息，请参阅 _DatastoreSizeRemaining。*
_DatastoreAttachError	只读	布尔型	此标记用于指示已发生的阻止使用“存储并转发”的错误。当标记值为 True 时，表示发生了错误。有关此错误的信息，请参阅服务器事件日志。*请参阅“可能的原因/解决方案”，以解决阻止使用“存储并转发”数据存储的错误。*
_DroppedUpdates	只读	长整型	此标记用于报告自 ThingWorx Interface 启动以来丢弃的所有更新数量。当此值达到 2,147,483,647 时，其会重置为 0。当重新初始化 ThingWorx 连接时，此值将会重置为 0。
_ForwardMode	读/写		此标记用于报告 ThingWorx 本机接口的当前转发

标记	类	数据类型	说明
			模式配置。此标记支持通过写入来更改已配置的模式。有效的值包括 0 (表示“活动”) 和 1 (表示“暂停”)。所有其他的写入值均会被忽略。 **注意：** • 配置并非总能指示正在使用的转发模式。可使用 __ForwardModeStatus 系统标记来获取正在使用的模式。例如，当发生阻止“存储并转发”的错误时，__ ForwardModeStutus 会返回空白。
_ForwardModeStatus	只读	字符串	此标记用于报告本机接口当前正在使用的转发模式。可能的值包括“活动”和“暂停”。当前未使用“存储并转发”时，系统标记会返回空白字符串。

另请参阅：ThingWorx Interface 用户，了解如何控制对 ThingWorx Platform 的访问和相关的数据传输。

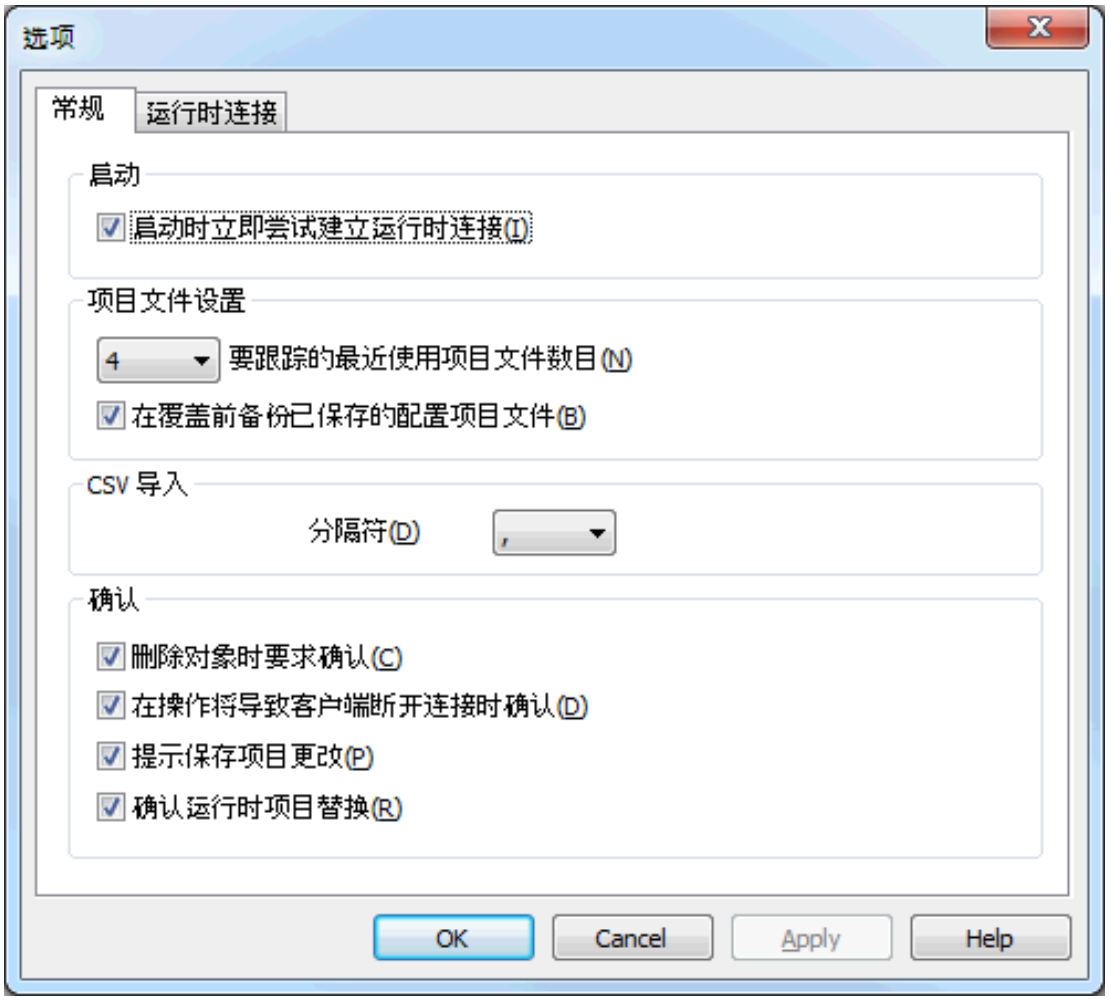
服务器选项

要通过配置访问“服务器选项”组，请单击**“工具”|“选项”**。单独对这些设置进行配置。有关详细信息，请从下表中选择一个链接。

选项 -“常规”
选项 -“运行时连接”

选项 - 常规

此对话框可用于指定常规服务器选项 (例如，何时与“运行时”建立连接、何时备份保存的配置项目文件以及在什么条件下调用警告弹出窗口)。



“启动时立即尝试建立运行时连接”：通过此图标选择是否在启动时将配置工具连接至“运行时”。禁用时，用户必须手动连接。默认设置为启用状态。

Number of recently used project files to track：选择的项目 (**最近使用过**) **MRU** 列表中显示的项目文件数。有效范围为 1 到 16。默认设置为 8。

“覆盖前备份保存的配置项目文件”：启用后，在上次保存的配置项目文件被新项目文件覆盖前，系统会自动建立备份副本。备份文件的名称和位置显示在事件日志中。

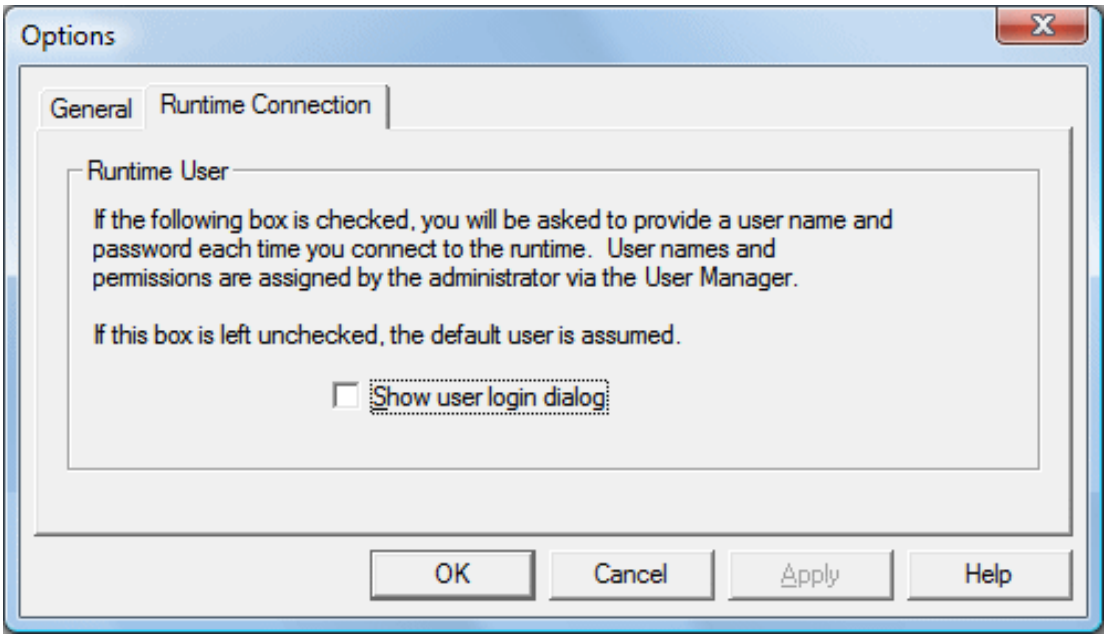
“CSV 导入”：**“分隔符”**设置可指定逗号分隔变量 (CSV)，服务器可使用该变量将标记数据以 CSV 文件的格式导入和导出。选项包括逗号和分号。默认设置为逗号。

"确认"：选择在哪些情况下对配置进行强制，以向操作员显示警告弹出窗口。

- **"删除对象时要求确认"**：启用后，所有配置删除操作均会调用警告弹出窗口，您需要对窗口进行确认，然后才能完成删除操作。
- 此弹出窗口需要确认，然后才能启动断开序列。
- **"提示保存项目更改"**：启用后，当项目有未保存的更改时，如果服务器正在关闭，则配置会调用弹出窗口。
- **"确认运行时项目替换"**：启用后，此选项会发出警告：在配置连接到"运行时"的情况下，可离线打开项目并进行编辑。

选项 - 运行时连接

此对话框用于指定如何管理与运行时的连接。



"显示用户登录对话框"：启用后，需要具有有效的用户名和密码，才可将配置连接运行时，以进行项目编辑。默认设置为禁用状态。

注意：用户名和权限由管理员分配。*有关详细信息，请参阅设置 - 用户管理器*。

基本组件

有关特定服务器组件的详细信息，请从下表中选择一个链接。

什么是信道？
什么是设备？
什么是标记？
什么是标记组？
什么是别名映射？
什么是事件日志？

什么是信道？

信道表示 PC 与一个或多个外部设备之间的通信媒介。信道可用于表示串行端口、PC 中安装的卡或以太网插槽。

用户必须在为项目添加设备前定义与设备通信时使用的信道。信道和设备驱动器需紧密相关。创建信道后，只能将选定驱动程序支持的设备添加到该信道。

添加信道

使用信道向导添加信道，该信道向导可指导用户完成信道定义过程。首先，系统将提示用户将分配信道的逻辑名称。此名称在项目中定义的所有信道和设备中必须具有唯一性。有关保留字符的信息，请参阅如何正确命名信道、设备、标记和标记组。

系统将提示用户将使用的设备驱动程序。将出现一个列表框，其中显示当前所有安装在系统中的设备驱动程序。所有串行驱动程序都可用于同一个项目中的多个信道。

注意：有关硬件卡驱动程序，请参阅驱动程序的帮助文档，以确定其是否能够用于单个项目中的多个信道。有关如何确定受支持信道数量的信息，请参阅服务器汇总信息。

系统将提示用户将使用的具体通信参数。多个信道不能公用相同的通信参数，例如，两个串行驱动程序不能使用 COM1。有关特定设备的正确通信参数，请参阅制造商帮助文档和驱动程序帮助文档。

注意：串行驱动程序的流量控制设置主要在通过转换器将 RS422/485 网络设备连接到 RS232 串行端口时使用。大多数 RS232 到 RS422/485 转换器需要无流量控制 (“无”)，或在 PC 传送时 RTS 线路开启，在监听时，RTS 关闭。

信道向导完成后将产生新信道的摘要。

移除信道

选择需要移除的信道，然后按 **Delete** 键，以移除项目中的信道。或者，在“编辑”菜单或“工具栏”中选择**“编辑”|“删除”**。

显示信道属性

选择信道，然后在“编辑”菜单或“工具栏”中单击**“编辑”|“属性”**，以显示特定信道的信道属性。

另请参阅：信道属性 - 常规

通道属性 - 常规

此服务器支持同时使用多个通信驱动程序。服务器项目中使用的各个协议或驱动程序称为通道。服务器项目可以由具有相同通信驱动程序或具有唯一通信驱动程序的多个通道组成。通道充当 OPC 链路的基础构建块。此组用于指定常规通道属性，如标识属性和操作模式。

标识

"名称"：此通道的用户定义标识。在每个服务器项目中，每个通道名称都必须是唯一的。尽管名称最多可包含 256 个字符，但在浏览 OPC 服务器的标记空间时，一些客户端应用程序的显示窗口可能不够大。通道名称是 OPC 浏览器信息的一部分。

有关保留字符的信息，请参阅服务器帮助中的"如何正确命名通道、设备、标记和标记组"。

"说明"：有关此通道的用户定义信息。

这些属性 (包括 Description) 当中有很多具有关联的系统标记。

"驱动程序"：为该通道选择的协议/驱动程序。该属性指定在通道创建期间选择的设备驱动程序。它在通道属性中为禁用设置。

注意：服务器全天在线运行时，可以随时更改这些属性。其中包括更改通道名称以防止客户端向服务器注册数据。如果客户端在通道名称更改之前已从服务器中获取了项，那么这些项不会受到任何影响。如果客户端应用程序在通道名称更改之后发布项，并尝试通过原来的通道名称重新获取项，则该项将不被接受。考虑到这一点，一旦开发完成大型客户端应用程序，就不应对属性进行任何更改。利用"用户管理器"可防止操作员更改属性并限制对服务器功能的访问权限。

诊断

"诊断数据捕获"：启用此选项后，通道的诊断信息即可用于 OPC 应用程序。由于服务器的诊断功能所需的开销处理量最少，因此建议在需要时使用这些功能，而在不需要时禁用这些功能。默认设置为禁用状态。

注意：如果驱动程序不支持诊断，则该属性不可用。

有关详细信息，请参阅服务器帮助中的"通信诊断"。

通道属性 - 高级

此组用于指定高级通道属性。并非所有驱动程序都支持所有属性，因此不会针对不支持的设备显示"高级"组。

"非规范浮点数处理"：非规范值定义为无穷大、非数字 (NaN) 或不正规编号。默认值为"替换为零"。具有原生浮点数处理功能的驱动程序可能会默认设置为"未修改"。通过非规范浮点数处理，用户可以指定驱动程序处理非规范 IEEE-754 浮点数据的方式。选项说明如下：

- **"替换为零"**：此选项允许驱动程序在将非规范 IEEE-754 浮点值传输到客户端之前，将其替换为零。
- **"未修改"**：此选项允许驱动程序向客户端传输 IEEE-754 不正规、规范、非数字和无穷大值，而不进行任何转换或更改。

注意：如果驱动程序不支持浮点值或仅支持所显示的选项，则此属性不可用。根据通道的浮点规范化设置，将仅对实时驱动程序标记 (如值和数组) 进行浮点规范化。例如，此设置不会影响 EFM 数据。

有关浮点值的详细信息，请参阅服务器帮助中的"如何使用非规范化浮点值"。

"设备间延迟"：指定在接收到同一通道上的当前设备发出的数据后，通信通道向下一设备发送新请求前等待的时间。设置为零 (0) 将禁用延迟。

注意：此属性并不适用于所有驱动程序、型号和相关设置。

通道属性 - 以太网通信

以太网通信可用于与设备进行通信。

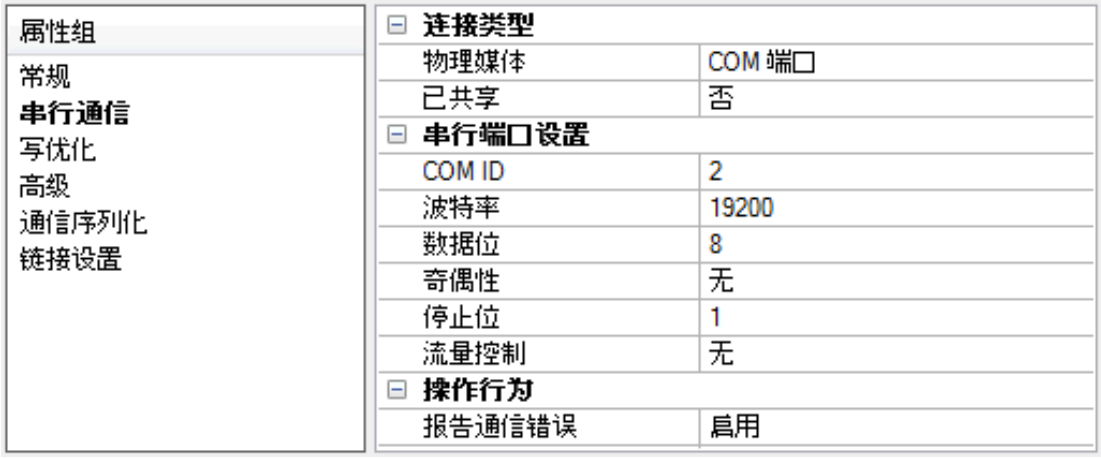
以太网设置

"网络适配器"：指定要绑定的网络适配器。如果选择"默认"(Default)，则操作系统将选择默认适配器。

通道属性 - 串行通信

串行通信属性可用于串行驱动程序，且随驱动程序、连接类型以及所选选项的不同而变化。以下是可能具有的属性的超集。
单击跳转至下列其中一个部分："连接类型"、"串行端口设置"或"以太网设置"以及"操作行为"。

注意：服务器全天在线运行时，可以随时更改这些属性。由于对这些属性进行更改后可能会暂时中断通信，因此可通过"用户管理器"来限制对服务器功能的访问权限。



连接类型

"物理媒体"：选择用于数据通信的硬件设备的类型。选项包括"COM 端口"、"无"、"调制解调器"和"以太网封装"。默认选项为 COM 端口。

- **"无"**：选择"无"表示没有物理连接，此时将显示"无通信的操作"部分。
- **"COM 端口"**：选择"Com 端口"可显示和配置"串行端口设置"部分。
- **"调制解调器"**：当用电话线进行通信时，选择"调制解调器"，并在"调制解调器设置"部分中对该选项进行配置。
- **"以太网封装"**：选择是否将"以太网封装"用于通信，此时将显示"以太网设置"部分。
- **"共享"**：验证是否已将连接正确标识为与其他通道共享当前配置。为只读属性。

"串行端口设置"

“COM ID”：指定在与分配给通道的设备进行通信时要使用的通信 ID。有效范围为 1 至 9991 至 16。默认值为 1。

“波特率”：指定用于配置选定通信端口的波特率。

“数据位”：指定每个数据字的数据位数。选项包括 5、6、7 或 8。

“奇偶性”：指定数据的奇偶类型。选项包括“奇”、“偶”或“无”。

“停止位”：指定每个数据字的停止位数。选项包括 1 或 2。

“流量控制”：选择 RTS 和 DTR 控制线的使用方式。在与一些串行设备进行通信时需要对流量进行控制。选项包括：

- **“无”**：此选项不会切换或添加控制线。
- **“DTR”**：当通信端口打开并保持开启状态时，此选项将添加 DTR 线路。
- **“RTS”**：此选项指定，如果字节适用于传输，则 RTS 线路为高电平。在发送所有缓冲字节后，RTS 线路变为低电平。这通常用于 RS232/RS485 转换器硬件。
- **“RTS, DTR”**：此选项是 DTR 和 RTS 的组合选项。
- **“始终 RTS”**：当通信端口打开并保持开启状态时，此选项将添加 RTS 线路。
- **“RTS 手动”**：此选项将基于为“RTS 线路控制”输入的定时属性添加 RTS 线路。该选项仅在驱动程序支持手动 RTS 线路控制 (或属性共享且至少有一个通道属于提供此类支持的驱动程序) 时可用。“RTS 手动”添加**“RTS 线路控制”**属性时具有如下选项：
    - **“上升”**：该属性用于指定在数据传输前 RTS 线路上升为高电平所需的时间量。有效范围为 0 至 9999 毫秒。默认值为 10 毫秒。
    - **“下降”**：该属性用于指定在数据传输后 RTS 线路保持高电平的时间量。有效范围为 0 至 9999 毫秒。默认值为 10 毫秒。
    - **“轮询延迟”**：该属性用于指定通信轮询的延迟时间量。有效范围为 0 到 9999。默认值为 10 毫秒。

提示：在使用双线 RS-485 时，通信线路上可能会出现“回波”。由于此类通信不支持回波抑制，因此建议禁用回波或使用 RS-485 转换器。

“操作行为”

- **“报告通信错误”**：启用或禁用报告低级通信错误。启用时，如果出现低级错误，则会将其发布到“事件日志”。禁用时，即使正常请求失败，也不会发布这些相同的错误。
- **“关闭空闲连接”**：当通道上的客户端不再引用任何标记时，选择关闭连接。默认设置为“启用”。
- **“关闭前空闲时间”**：指定在移除所有标记后服务器在关闭 COM 端口前所等待的时间。默认值为 15 秒。

以太网设置

注意：不是所有的串行驱动程序都支持以太网封装。若此组未出现，则无法支持相关功能。

如果要同与以太网终端服务器相连的串行设备进行通信，则可通过“以太网封装”来实现。终端服务器本质上是将以太网上的 TCP/IP 消息转换为串行数据的虚拟串行端口。消息转换完毕后，用户可将支持串行通信的标准设备连接到终端服务器。必须对终端服务器的串行端口进行正确配置，以满足所连串行设备的要求。*有关详细信息，请参阅服务器帮助中的“如何使用以太网封装”。*

- **“网络适配器”**：用于指示此通道中以太网设备绑定的网络适配器。选择要绑定的网络适配器，或者允许操作系统选择默认项。
  某些特定的驱动程序可能会显示其他“以太网封装”属性。有关详细信息，请参阅“通道属性 - 以太网封装”。

“调制解调器设置”

- **“调制解调器”**：指定用于通信的已安装调制解调器。
- **“连接超时”**：指定读取或写入失败前建立连接所等待的时间。默认值为 60 秒。

- **"调制解调器属性"**：配置调制解调器硬件。单击该选项后，将打开供应商特定的调制解调器属性。
- **"自动拨号"**：启用自动拨打电话簿中的条目。默认设置为"禁用"。*有关详细信息，请参阅服务器帮助中的"调制解调器自动拨号"。*
- **"报告通信错误"**：启用或禁用报告低级通信错误。启用时，如果出现低级错误，则会将其发布到"事件日志"。禁用时，即使正常请求失败，也不会发布这些相同的错误。
- **"关闭空闲连接"**：当通道上的客户端不再引用任何标记时，选择关闭调制解调器连接。默认设置为"启用"。
- **"关闭前空闲时间"**：指定在移除所有标记后服务器在关闭调制解调器连接前所等待的时间。默认值为 15 秒。

无通信的操作

- **"读取处理"**：选择要在请求显式设备读取时执行的操作。选项包括"忽略"和"失败"。"忽略"不执行任何操作；"失败"会为客户端提供一条指示失败的更新信息。默认设置为"忽略"。

通道属性 - 以太网封装

"以太网封装"可用于无线网络连接 (例如 802.11b 和 CDPD 数据包网络)，并且还经过开发，可以支持多种串行设备。通过终端服务器设备，用户可在工厂中放置 RS-232 和 RS-485 设备，同时仍然允许单个本地化 PC 访问远程挂载设备。"以太网封装"还可以根据需要将各个网络 IP 地址分配到设备。通过使用多个终端服务器，用户可以从单个 PC 访问数百个串行设备。用户可以将一个通道定义为使用本地 PC 串行端口，而将另一个通道定义为使用"以太网封装"。

注意：这些属性仅适用于串行驱动程序。显示的属性取决于所选通信驱动程序和支持的功能。

"网络适配器"：此属性用于指定网络适配器。

"设备地址"：此属性用于指定与此设备连接的终端服务器的四字段 IP 地址。IP 指定为 *YYY.YYY.YYY.YYY*。*YYY* 可指定 IP 地址：每个 *YYY* 字节应在 0 至 255 的范围内。每个通道均有其自己的 IP 地址。

"端口"：此特性用于配置在连接到远程终端服务器时使用的以太网端口。有效范围是 1 至 65535，其中某些数字予以保留。默认值为 2101。

"协议"：此属性用于指定 TCP/IP 或 UDP 通信，具体取决于正在使用的终端服务器的性质。默认值为 TCP/IP。有关可用协议的详细信息，请参阅终端服务器的帮助文档。

重要事项："以太网封装"模式对于实际的串行通信驱动程序是完全透明的。用户必须配置其余的设备属性，如同他们直接通过本地 PC 串行端口连接到设备一样。

"连接超时"：此属性用于指定为要调整的远程设备建立套接字连接所需的时间。在许多情况下，设备的连接时间比向该同一设备发送正常通信请求所需的时间更长。有效范围为 1 到 999 秒。默认值为 3 秒。

注意：服务器全天在线运行时，可以随时更改这些属性。使用"用户管理器"可限制对服务器功能的访问权限，并防止操作员更改属性。

通道属性 - 通信序列化

服务器的多线程架构使通道能够与设备并行通信。尽管这十分高效，但在存在物理网络限制 (如以太网无线电) 的情况下，通信可能会进行序列化。通信序列化将限制在虚拟网络中每次仅使用一个通道进行通信。

术语"虚拟网络"是指使用同一管线进行通信的通道和相关设备的集合。例如，以太网无线电管线是主无线电。使用同一主无线电的所有通道均与同一虚拟网络相关联。默认情况下，通道在向另一通道传递通信前，可处理一个事务。一个事务中可包括一个或多个标记。如果控制通道包含的设备未响应请求，则在事务超时之前，通道无法释放控制权。这会导致虚拟网络中其他通道的数据更新延迟。

通道级别设置

"虚拟网络"：此属性可指定通道的通信序列化模式。选项包括"无"和"网络 1 - 网络 500"。默认值为"无"。选项说明如下：

- **"无"**此选项将禁用通道的通信序列化。
- **"网络 1 - 网络 500"**：此选项可指定分配通道的虚拟网络。

"每周期的事务数"：此属性可指定通道中可能发生的单一分块/非分块读/写事务的数量。当通道可以进行通信时，将尝试该事务数。有效范围为 1 到 99。默认值为 1。

全局设置

- **"网络模式"**：此属性用于控制委派通道通信的方式。在**"负载平衡"**模式下，每个通道可以逐一轮流进行通信。在**"优先级"**模式下，通道可以根据以下规则 (优先级由高到低) 进行通信：
  - 具有待处理写入操作的通道具有最高优先级。
  - 具有待处理显式读取操作 (通过内部插件或外部客户端接口) 的通道的优先级基于读取的优先级。
  - 扫描读取和其他定期事件 (特定于驱动程序)。

  默认设置为"负载平衡"，这并影响*所有*虚拟网络和通道。

依赖于主动响应的设备不应置于虚拟网络中。在必须进行通信序列化的情况下，建议启用"自动降级"。

由于驱动程序的数据读取和写入方式的差异 (如单一、分块或非分块事务)，可能需要调整应用程序的"每周期的事务数"属性。执行此操作时，请考虑以下因素：

- 必须从每个通道读取多少标记？
- 数据写入各个通道的频率如何？
- 通道使用串行驱动程序还是以太网驱动程序？
- 驱动程序是读取单独请求中的标记还是读取块中的多个标记？
- 设备的定时属性 (如请求超时和 *x* 次连续超时后失败) 是否针对虚拟网络通信媒介进行了优化？

通道属性 - 网络接口

有了"以太网封装"，几乎所有当前可用的驱动程序均可支持某种形式的以太网通信。无论是基于本地以太网的驱动程序还是为"以太网封装"配置的串行驱动程序，均使用了某种形式的网络接口。在大多数情况下，该接口会采用网络接口卡 (NIC) 的形式。对于安装有网络的 PC 而言，这通常意味着安装单个 NIC 便可实现与 IT 或车间网络 (或与两者) 的连接。

此配置非常适用于典型的网络配置和网络加载。然而，如果需要定期从以太网设备接收数据，则可能会出现问题。如果车间网络与 IT 网络混合，则大批量文件传输可能会完全扰乱车间数据的间隔。处理此问题最常见的方法就是在 PC 中再安装一个 NIC。一个 NIC 可用于访问 IT 网络，另一个 NIC 可用于访问车间数据。虽然这听起来似乎很合理，但在尝试分离网络时可能会出现问题。当使用多个 NIC 时，用户必须确定绑定顺序。绑定顺序决定了访问以太网的不同部分时所使用的 NIC。在许多情况下，可以使用操作系统工具来管理绑定设置。

当没有明确界定每个 NIC 卡所使用的协议类型和服务类型时，可通过操作系统创建绑定顺序。如果没有明确的方式用以选择特定绑定顺序，则用户可能会发现以太网设备连接会被传送至错误的网络。在这种情况下，可通过如下所示的网络接口选择用于以太网驱动程序的特定 NIC 卡。网络接口选择可用于根据 NIC 名

称或当前为其分配的 IP 地址来选择特定的 NIC 卡。此列表列出了可用的 NIC，其中包括唯一 NIC 卡或分配有多个 IP 地址的 NIC 卡。此选择将显示处于活动状态的任何 WAN 连接 (例如拨号连接)。
注意：此属性仅适用于以太网驱动程序。

通过选择一个特定的 NIC 接口，用户可以强制驱动程序通过指定的 NIC 发送所有以太网通信。选择 NIC 后，正常的操作系统绑定顺序会被完全忽略。这可确保用户可以控制网络的运行方式并消除一切疑虑。

显示在"网络适配器"下拉菜单中的选项取决于网络配置设置、PC 上安装的唯一 NIC 数量以及分配给 NIC 的唯一 IP 数量。要强制操作系统创建绑定顺序选择，可将网络适配器选为"默认值"。这使得驱动程序可通过操作系统的正常绑定顺序对 NIC 进行设置。
重要事项：当不确定该使用哪个 NIC 时，选择默认条件。此外，当基于以太网的设备已被占用且此功能已通过产品升级推出时，可选择默认条件。
注意：服务器全天在线运行时，可以随时更改这些属性。使用"用户管理器"可限制对服务器功能的访问权限，并防止操作员更改属性。请注意，更改该属性可能会暂时中断通信。

通道属性 - 写入优化

与任何服务器一样，将数据写入设备可能是应用程序应具备的最重要的功能。服务器旨在确保从客户端应用程序写入的数据能够准时发送到设备。为了达到此目标，服务器提供了可用来满足特定需求以提高应用程序响应能力的优化属性。

写入优化

"优化方法"：控制如何将写入数据传递至底层通信驱动程序。选项包括：

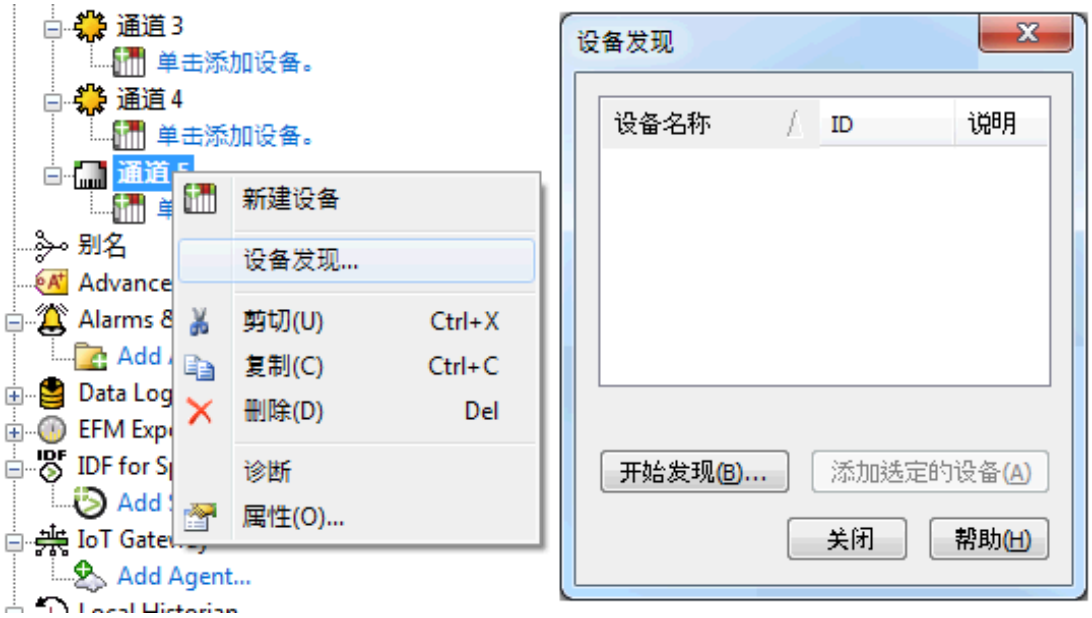
- **"写入所有标记的所有值"**：此选项可强制服务器尝试将每个值均写入控制器。在此模式下，服务器将持续收集写入请求并将它们添加到服务器的内部写入队列。服务器将对写入队列进行处理并尝试通过将数据尽快写入设备来将其清空。此模式可确保从客户端应用程序写入的所有数据均可发送至目标设备。如果写入操作顺序或写入项的内容必须且仅能显示于目标设备上，则应选择此模式。
- **"写入非布尔标记的最新值"**：由于将数据实际发送至设备需要一段时间，因此对同一个值的多次连续写入会存留于写入队列中。如果服务器要更新已位于写入队列中的某个写入值，则需要大大减少写入操作才能获得相同的最终输出值。这样一来，便不会再有额外的写入数据存留于服务器队列中。几乎就在用户停止移动滑动开关时，设备中的值达到其正确值。根据此模式的规定，任何非布尔值都会在服务器的内部写入队列中更新，并在下一个可能的时机发送至设备。这可以大大提高应用性能。
  注意：该选项不会尝试优化布尔值的写入。它允许用户在不影响布尔运算的情况下优化 HMI 数据的操作，例如瞬时型按钮等。
- **"写入所有标记的最新值"**：该选项采用的是第二优化模式背后的理论并将其应用至所有标记。如果应用程序只需向设备发送最新值，则该选项尤为适用。此模式会通过在当前写入队列中的标记发送前对其进行更新来优化所有的写入操作。此为默认模式。

"占空比"：用于控制写操作与读操作的比率。该比率始终基于每一到十次写入操作对应一次读取操作。占空比的默认设置为 10，这意味着每次读取操作对应十次写入操作。即使在应用程序执行大量的连续写入操作时，也必须确保足够的读取数据处理时间。如果将占空比设置为 1，则每次读取操作对应一次写入操作。如果未执行任何写入操作，则会连续处理读取操作。相对于更加均衡的读写数据流而言，该特点使得应用程序的优化可通过连续的写入操作来实现。

注意：建议在将应用程序投入生产环境前使其与写入优化增强功能相兼容。

??????

"????"??????????????????????,??????????????????????????? 65535?



1. ??????????????
2. ??????????**“????...”**?
3. ??**“????...”**?????????
4. ??????????????,??:????????????
5. ??**“??”**?
6. ?????????????????/??:**???ID???**?
7. ??????????,?????????**“???????...”**?
8. ??**“??”**?

什么是设备？

设备代表与服务器通信的 PLC 或其他硬件。信道使用的设备驱动程序限制了设备选择。

添加设备

在初始设置及后续阶段，可使用“新建设备向导”添加设备。要执行此操作，请单击**“编辑”(Edit) | “新建设备”(New Device)**。会提示用户输入“设备名称”(Device Name)，该名称由用户自定义且应符合该设备的情况。此为 OPC 链接中使用的浏览器分支名称，用于访问设备的已分配标记。有关保留字符的信息，请参阅如何正确命名通道、设备、标记和标记组。

还会提示用户输入“网络 ID”(Network ID)，该 ID 为数字或字符串，用于对设备网络上的设备进行唯一标识。联网的多点通信设备必须具有唯一标识符，以便正确路由服务器的数据请求。非多点通信设备无需 ID；此设置对其不可用。

移除设备

要从项目中移除设备，请选择所需设备，然后按**“删除”(Delete)**。或者，单击**“编辑”(Edit) | “删除”(Delete)**。

显示设备属性

要显示设备属性，请先选择设备，然后单击**“编辑”(Edit) | “属性”(Properties)**。

有关详细信息，请参阅设备属性。

设备属性 - 常规

一个设备代表通信通道上的单一目标。如果驱动程序支持多个控制器，则用户必须为每个控制器输入一个设备 ID。

属性组		
常规	⊟ 标识	
扫描模式	名称	设备 1
	说明	
	驱动程序	Simulator
	型号	16 Bit Device
	通道分配	通道 2
	ID 格式	十进制
	ID	1
	⊟ 操作模式	
	数据收集	禁用

标识

名称：此属性用于指定设备的名称。此为用户定义的逻辑名称，最长可达 256 个字符，并且可以用于多个通道。

注意：尽管描述性名称通常是很好的选择，但浏览 OPC 服务器的标记空间时，一些 OPC 客户端应用程序的显示窗口可能不够大。设备名称和信道名称也成为浏览树信息的一部分。OPC 客户端中，信道名称和设备名称的组合将显示为“信道名称.设备名称”。

有关详细信息，请参阅服务器帮助中的“如何为通道、设备、标记和标记组正确命名”。

“说明”(Description)：有关此设备的用户定义信息。

在这些属性 (包括 Description) 当中有很多具有关联的系统标记。

“通道分配”：该设备当前所属通道的用户定义名称。

驱动程序：为该设备选择的协议驱动程序。该属性指定在通道创建期间选择的驱动程序。它在通道属性中是禁用的。

型号：此属性指定与此 ID 关联的特定设备类型。下拉菜单中的内容取决于正在使用的通信驱动程序类型。驱动程序不支持的型号将被禁用。如果通信驱动程序支持多个设备型号，则只有当设备未与任何客户端应用程序连接时，才能改变型号的选择。

注意：如果通信驱动程序支持多种型号，则用户应将型号选择与物理设备进行匹配。如果下拉列表菜单中未显示该设备，则选择与目标设备最相近的型号。一些驱动程序支持名为“开放式”的型号选择，该选择使用户无需了解目标设备的具体信息即可进行通信。有关详细信息，请参阅驱动程序帮助文档。

ID：此属性指定设备的工作站/节点/标识/地址。输入的 ID 类型取决于正在使用的通信驱动程序。对于许多驱动程序而言，ID 是一个数值。支持数字 ID 的驱动程序使用户能够输入格式可更改的数值，以适应应用程需要或所选通信驱动程序特点。ID 格式可以是十进制、八进制和十六进制。如果驱动程序基于以太网，或者支持非常规工作站或节点名称，则可使用设备的 TCP/IP 地址作为设备 ID。TCP/IP 地址包含四个由句点分隔的值，每个值的范围在 0 至 255 之间。某些设备 ID 基于字符串。根据不同驱动程序，也可以在 ID 字段中配置其他属性。

操作模式

“数据收集”：此属性控制设备的活动状态。尽管默认情况下会启用设备通信，但可使用此属性禁用物理设备。设备处于禁用状态时，不会尝试进行通信。从客户端的角度来看，数据将标记为无效，且不接受写入操作。通过此属性或设备系统标记可随时更改此属性。

“模拟”：此选项可将设备置于模拟模式。在此模式下，驱动程序不会尝试与物理设备进行通信，但服务器将继续返回有效的 OPC 数据。模拟停止与设备的物理通信，但允许 OPC 数据作为有效数据返回到 OPC 客户端。在“模拟模式”下，服务器将所有设备数据处理为反射型：无论向模拟设备写入什么内容，都会读取回来，而且会单独处理每个 OPC 项。项的内存映射取决于组更新速率。如果服务器移除了项 (如服务器重新初始化时)，则不保存数据。默认值为“否”。

注意：

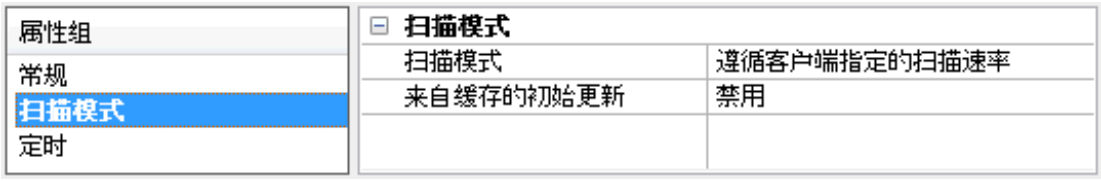
1. “系统”标记 (_Simulated) 为只读且无法写入，从而达到运行时保护的目的。“系统”标记允许从客户端监控此属性。

2. 在“模拟”模式下，项的内存映射取决于客户端更新速率 (OPC 客户端的“组更新速率”或本机和 DDE 接口的扫描速率)。这意味着，参考相同项、而采用不同更新速率的两个客户端会返回不同的数据。

“模拟模式”仅用于测试和模拟目的。该模式永远不能用于生产环境。

设备属性 - 扫描模式

“扫描模式”为需要设备通信的标记指定订阅客户端请求的扫描速率。同步和异步设备的读取和写入会尽快处理；不受“扫描模式”属性的影响。



“扫描模式”：为发送到订阅客户端的更新指定在设备中扫描标记的方式。选项说明如下：

- **“遵循客户端指定的扫描速率”**：此模式可使用客户端请求的扫描速率。
- **“不超过扫描速率请求数据”**：此模式可指定要使用的最大扫描速率。有效范围为 10 至 99999990 毫秒。默认值为 1000 毫秒。
  注意：当服务器有活动的客户端和设备项且扫描速率值有所提高时，更改会立即生效。当扫描速率值减小时，只有所有客户端应用程序都断开连接，更改才会生效。
- **“以扫描速率请求所有数据”**：此模式将以订阅客户端的指定速率强制扫描标记。有效范围为 10 至 99999990 毫秒。默认值为 1000 毫秒。
- **“不扫描，仅按需求轮询”**：此模式不会定期轮询属于设备的标签，也不会在一个项变为活动状态后为获得项的初始值而执行读取操作。客户端负责轮询以便更新，方法为写入 _DemandPoll 标记或为各项发出显式设备读取。*有关详细信息，请参阅服务器帮助中的“设备需求轮询”。*
- **“遵循标签指定的扫描速率”**：此模式将以静态配置标记属性中指定的速率强制扫描静态标记。以客户端指定的扫描速率扫描动态标记。

“来自缓存的初始更新”：启用后，此选项允许服务器为存储 (缓存) 数据的新激活标签参考提供第一批更新。只有新项参考共用相同的地址、扫描速率、数据类型、客户端访问和缩放属性时，才能提供缓存更新。设备读取仅用于第一个客户端参考的初始更新。默认设置为禁用；只要客户端激活标记参考，服务器就会尝试从设备读取初始值。

设备属性 - 自动降级

自动降级属性可以在设备未响应的情况下使设备暂时处于关闭扫描状态。通过将特定时间段内无响应的设备脱机，驱动程序可以继续优化与同一信道上其他设备的通信。该时间段结束后，驱动程序将重新尝试与无响应设备进行通信。如果设备响应，则该设备会进入开启扫描状态；否则，设备将再次开始其关闭扫描时间段。

“故障时降级”：启用后，将自动对设备取消扫描，直到该设备再次响应。
提示：使用 _AutoDemoted 系统标记来监视设备的降级状态，确定何时对设备取消扫描。

“降级超时”：指定在对设备取消扫描之前，请求超时和重试的连续周期数。有效范围是 1 到 30 次连续失败。默认值为 3。

“降级期间”：指示当达到超时值时，对设备取消扫描多长时间。在此期间，读取请求不会被发送到设备，与读取请求关联的所有数据都被设置为不良质量。当此期间到期时，驱动程序将对设备进行扫描，并允许进行通信尝试。有效范围为 100 至 3600000 毫秒。默认值为 10000 毫秒。

“降级时放弃请求”：选择是否在取消扫描期间尝试写入请求。如果禁用，则无论是否处于降级期间都始终发送写入请求。如果启用，则放弃写入；服务器自动将接收自客户端的写入请求视为失败，且不会在事件日志中记录消息。

设备属性 - 通信参数

“以太网封装”模式旨在为通过以太网与终端服务器相连的串行设备提供通信。终端服务器实质上是虚拟串行端口。终端服务器会将以太网上的 TCP/IP 消息转换为串行数据。消息转换为串行形式后，用户可将支持串行通信的标准设备连接到终端服务器。

有关详细信息，请参阅服务器帮助中的“如何使用以太网封装”。

注意：因为以太网封装模式对于实际串行通信驱动程序是完全透明的，用户配置其余的设备属性时，应与直接连接本地 PC 串行端口上的设备一样。

“IP 地址”(IP Address)：此属性用于输入与设备连接的终端服务器的四字段 IP 地址。IP 指定为 YYY.YYY.YYY.YYY。YYY 指定 IP 地址：每个 YYY 字节应在 0 至 255 的范围内。每个串行设备都可以有其自己的 IP 地址；但是，如果多个设备与单个终端服务器进行多点通信时，则这些设备可能使用相同的 IP 地址。

“端口”(Port)：在连接到远程终端服务器时，此属性用于配置以太网端口。

“协议”(Protocol)：此属性用于选择 TCP/IP 或 UDP 通信。该选择取决于正在使用的终端服务器的性质。默认协议选项为 TCP/IP。有关可用协议的详细信息，请参阅终端服务器的帮助文档。

注意：

1. 服务器全天在线运行时，可以随时更改这些属性。使用“用户管理器”可限制对服务器功能的访问权限，并防止操作员更改属性。
2. 有效的 IP 地址范围大于 (>) 0.0.0.0 且小于 (<) 255.255.255.255。

设备属性 - 以太网封装

“以太网封装”旨在为通过以太网与终端服务器相连的串行设备提供通信。终端服务器实质上是虚拟串行端口。终端服务器会将以太网上的 TCP/IP 消息转换为串行数据。消息转换为串行形式后，用户可将支持串行通信的标准设备连接到终端服务器。

有关详细信息，请参阅服务器帮助中的“如何使用以太网封装”。

“以太网封装”对于驱动程序来说是透明的；配置其余属性时，应与直接连接本地串行端口上的设备一样。

“IP 地址”(IP Address)：此属性用于输入与设备连接的终端服务器的四字段 IP 地址。IP 指定为 YYY.YYY.YYY.YYY。YYY 指定 IP 地址：每个 YYY 字节应在 0 至 255 的范围内。每个串行设备都可以有其自己的 IP 地址；但是，如果多个设备与单个终端服务器进行多点通信时，则这些设备可能使用相同的 IP 地址。

“端口”(Port)：在连接远程终端服务器时，可使用此属性配置以太网端口。

“协议”(Protocol)：此属性用于选择 TCP/IP 或 UDP 通信。该选择取决于正在使用的终端服务器的性质。默认协议选项为 TCP/IP。有关可用协议的详细信息，请参阅终端服务器的帮助文档。

注意

1. 服务器全天在线运行时，可以随时更改这些属性。使用“用户管理器”可限制对服务器功能的访问权限，并防止操作员更改属性。
2. 有效的 IP 地址范围大于 (>) 0.0.0.0 且小于 (<) 255.255.255.255。

设备属性 - 标记生成

自动标记数据库生成功能使设置应用程序成为一项即插即用操作。选择可以配置为自动构建标记列表的通信驱动程序 (标记与特定于设备的数据相对应)。可以从客户端浏览这些自动生成的标记 (这取决于支持驱动程序的性质)。

并非所有设备和驱动程序都支持全自动标记数据库生成，也并非所有都支持相同的数据类型。有关具体信息，请参阅各驱动程序的数据类型说明或支持的数据类型列表。

如果目标设备支持其自身的本地标记数据库，则驱动程序会读取设备的标记信息，并使用该数据来在服务器中生成标记。如果该设备本身不支持已命名的标记，则驱动程序会根据特定于驱动程序的信息来创建标记列表。这两个条件的示例如下：

1. 如果数据采集系统支持其自身的本地标记数据库，则通信驱动程序将使用在设备中发现的标记名称来构建服务器的标记。
2. 如果以太网 I/O 系统支持其自身可用 I/O 模块类型的检测，则通信驱动程序会基于插入以太网 I/O 机架的 I/O 模块类型在服务器中自动生成标记。

注意：自动标记数据库生成的操作模式可进行完全配置。有关详细信息，请参阅下方的属性说明。

"属性更改时"：如果设备支持在特定属性更改时自动生成标记，系统会显示**"属性更改时"**选项。默认情况下，该选项设置为**"是"**，但可以将其设置为**"否"**来控制何时生成标记。在此情况下，必须手动执行"创建标记"操作以执行标记生成。

"设备启动时"：此属性指定何时自动生成 OPC 标记。选项说明如下：

- **"启动时不生成"**：此选项可防止驱动程序向服务器的标记空间添加任何 OPC 标记。这是默认设置。
- **"始终在启动时生成"**：此选项可使驱动程序评估设备，以便获得标记信息。每次启动服务器时，它还会向服务器的标记空间添加标记。
- **"首次启动时生成"**：此选项可使驱动程序在首次运行项目时评估目标设备，以便获得标记信息。它还可以根据需要向服务器标记空间添加任何 OPC 标记。

注意：如果选择自动生成 OPC 标记的选项，添加到服务器标记空间的任何标记都必须随项目保存。用户可以在**"工具"|"选项"**菜单中将项目配置为自动保存。

"对于重复标记"：启用自动标记数据库生成后，服务器需要了解如何处理先前已添加的标记，或在初始创建通信驱动程序后添加或修改的标记。此设置可控制服务器处理自动生成的以及当前存在于项目中的 OPC 标记的方式。它还可以防止自动生成的标记在服务器中累积。

例如，如果用户更改机架中的 I/O 模块，并且服务器配置为**"始终在启动时生成"**，则每当通信驱动程序检测到新的 I/O 模块时，新标记就会添加到服务器。如果未移除旧标记，则许多未使用的标记可能会在服务器的标记空间中累积。选项包括：

- **"创建时删除"**：此选项可在添加任何新标记之前，将先前添加到标记空间的任何标记删除。这是默认设置。
- **"根据需要覆盖"**：此选项可以指示服务器仅移除通信驱动程序要用新标记替换掉的标记。所有未被覆盖的标记仍将保留在服务器的标记空间中。
- **"不覆盖"**：此选项可以防止服务器移除任何之前生成的标记或服务器中已存在的标记。通信驱动程序只能添加全新的标记。
- **"不覆盖，记录错误"**：此选项与前一选项有相同效果，并且在发生标记覆盖时，也会将错误消息发布到服务器的事件日志。

注意：删除 OPC 标记会影响通信驱动程序已自动生成的标记以及使用匹配已生成标记的名称添加的任何标记。如果标记所使用的名称可能与驱动程序自动生成的标记相匹配，则用户应避免将此类标记添加到服务器。

“父组”：此属性通过指定将要用于自动生成标记的组，来防止自动生成的标记与已手动输入的标记发生混淆。组名称最多可包含 256 个字符。此父组具有一个根分支，可将所有自动生成的标记添加到其中。

“允许自动生成的子组”：此属性用于控制服务器是否为自动生成的标记自动创建子组。这是默认设置。如果禁用，则服务器会在没有任何分组的简单列表中生成设备标记。在服务器项目中，生成的标记使用地址值命名。例如，生成过程中不会保留标记名称。

注意：如果在服务器生成标记的过程中，分配给标记的名称与现有标记的名称相同，则系统会自动递增到下一个最高数字，以免标记名称发生重复。例如，如果生成过程中创建了名为 "AI22" 的标记且该名称已存在，则会将标记创建为 "AI23"。

“创建”：开始创建自动生成的 OPC 标记。如果已修改设备的配置，则**“创建标记”**可强制驱动程序重新评估设备以发现可能的标记更改。由于该选项可以通过系统标记进行访问，这使得客户端应用程序能够启动标记数据库创建。

注意：当“配置”对项目进行离线编辑时，会禁用**“创建标记”**。

设备属性 - 时间同步

此组用于指定设备的时区和时间同步属性。它主要适用于带时间戳数据或来自远程位置 (设备时间可能有偏差，造成带时间戳数据的问题) 电池供电设备的信息。要防止发生此问题，用户可指定服务器同步设备时间。

注意：并非所有驱动程序和型号都支持所有选项。

“时区”(Time Zone)：此属性指定设备的时区。要忽略时区，请从列表的前四个选项中选择一个 (它们没有偏移)。默认值为本机系统的时区。
注意：驱动程序会在同步设备时间和将该设备的 EFM 时间戳转换为 UTC 时间时使用此属性。

“考虑夏令时”(Respect Daylight Saving Time)：选择“是”(Yes) 将在同步设备时间时遵循夏令时。选择“否”可忽略夏令时。只有遵循夏令时的时区才会受到影响。默认设置为“否”(禁用)。
注意：启用时，设备的时间会加 1 小时，调整为使用夏令时 (春季)，夏令时结束后 (秋季)，会减掉 1 小时。

“时间同步方法”：此属性可指定同步方法。选项包括“禁用”、“绝对值”和“间隔”。默认设置为“禁用”。选项说明如下：

- **“禁用”**：不进行同步。
- **“绝对值”(Absolute)**：同步到一日当中的绝对时间，该时间是通过“时间”(Time) 属性 (仅在选中“绝对值” (Absolute) 时显示) 来指定的。
- **“间隔”(Interval)**：在启动时，以及每次过去一定的分钟数后进行同步，该分钟数是通过“同步间隔” (Sync Interval) 属性 (仅在选中“间隔”(Interval) 时显示) 来指定的。默认值为 60 分钟。
- **“轮询”**：在轮询完成时进行同步 (仅适用于 EFM 设备)。

“时间同步阈值”：此属性指定：在将设备时间与系统时间同步之前，两者时间差异的最大允许值 (秒)。如果阈值设置为 0，则每次都进行时间同步。默认值为 0 秒。最大允许的阈值为 600 秒。

设备属性 - 定时

设备的“定时”属性允许调整驱动程序对错误条件的响应，以满足应用程序的需要。在很多情况下，需要更改环境的此类属性，以便获得最佳性能。由电气原因产生的噪音、调制解调器延迟以及较差的物理连接等因素都会影响通信驱动程序遇到的错误数或超时次数。“定时”属性特定于每个配置的设备。

通信超时

“连接超时”(Connect Timeout)：此属性 (主要由基于驱动程序的以太网使用) 控制建立远程设备套接字连接所需的时间长度。设备的连接时间通常比针对同一设备的正常通信请求所花费时间更长。有效范围为 1 到 30 秒。默认值通常为 3 秒钟，但可能会因驱动程序的具体性质而异。如果驱动程序不支持此设置，则此设置将被禁用。

注意：鉴于 UDP 连接的性质，当通过 UDP 进行通信时，连接超时设置不适用。

“请求超时”(Request Timeout)：此属性可指定一个所有驱动程序使用的间隔来决定驱动程序等待目标设备完成响应的时间。有效范围是 50 至 9,999,999 毫秒 (167.6667 分钟)。默认值通常是 1000 毫秒，但可能会因驱动程序而异。大多数串行驱动程序的默认超时是基于 9600 波特或更高的波特率来确定的。当以较低的波特率使用驱动程序时，请增加超时，以补偿获取数据所需增加的时间。

“超时前的尝试次数”：此属性用于指定在认定请求失败以及设备出错之前，驱动程序发出通信请求的次数。有效范围为 1 到 10。默认值通常是 3，但可能会因驱动程序的具体性质而异。为应用程序配置的尝试次数很大程度上取决于通信环境。此属性适用于连接尝试和请求尝试。

定时

“请求间延迟”(Inter-Request Delay)：此属性指定驱动程序在将下一个请求发送到目标设备之前等待的时间。它会覆盖设备关联标记的一般轮询频率，以及一次性读取和写入次数。在处理周转时间慢的设备时，以及担心网络负载问题时，这种延迟很有用。为设备配置延迟会影响与通道上所有其他设备的通信。建议用户尽可能将所有需要请求间延迟的设备隔离至单独的通道。其他通信属性 (如通信序列化) 可以延长此延迟。有效范围是 0 至 300,000 毫秒；但是，某些驱动程序可能因某项特别设计的功能而限制最大值。默认值为 0，它表示对目标设备的请求之间没有延迟。

注意：不是所有的驱动程序都支持“请求间延迟”。如果不可用，则此设置不会出现。

设备属性 - 冗余

Media-Level Redundancy 插件提供冗余。

有关详细信息，请参阅网站、向销售代表咨询或查阅用户手册。

什么是标记？

标记表示与服务器通信的 PLC 或其他硬件设备的地址。服务器允许同时使用“动态”标记和用户定义的“静态”标记。“动态”标记可在 OPC 客户端中直接输入，并指定设备数据。用户定义的“静态”标记创建于服务器并支持标记缩放。可以从支持标记浏览的 OPC 客户端浏览这些标记。

显示标记属性

双击服务器配置的“标记选择”窗格，以为特定标记调用标记属性。

Tag Name	Address	Data Type	Scan Rate	Scaling	Description
Tag1	40001	Word	100	None	
Tag2	40002	Word	100	None	
Tag3	40003	Word	100	None	
Tag4	40004	Float	100	None	
Tag5	40005	Word	100	None	
Tag6	40006	Word	100	Square Root	
Tag7	40007	Word	100	None	
Tag8	40008	Word	100	None	
Tag9	40009	Word	100	None	
Tag10	40010	Word	50	None	
Tag11	40011	Word	100	None	
Tag12	40012	Word	100	None	
Tag13	40013	Word	100	None	
Tag14	40014	Word	100	Linear	
Tag15	40015	Word	100	None	
Tag16	40016	Word	100	None	
Tag17	40017	Word	100	None	
Tag18	40018	LBCD	100	None	
Tag19	40019	Word	100	None	
Tag20	40020	Word	100	None	
Tag21	40021	Word	100	None	
Tag22	40022	Word	100	None	

标记属性 - 常规

标记表示 PLC 或其他硬件设备与服务器通信的地址。服务器允许同时使用“动态”标记和用户定义的“静态”标记。“动态”标记可在 OPC 客户端中直接输入，并指定设备数据。用户定义的“静态”标记创建于服务器并支持标记缩放。可以从支持标记浏览的 OPC 客户端浏览这些标记。

有关详细信息，请参阅动态标记 和静态用户定义标记。

“名称”(Name)：输入字符串来表示标记中的可用数据。标记名称的长度不能超过 256 个字符。尽管使用长的描述性名称通常是一个好方法，但在浏览 OPC 服务器的标记空间时，一些 OPC 客户端应用程序的显示窗口

可能不够大。标记名称是 OPC 浏览数据标记名称的一部分，在指定设备分支或标记组分支内必须唯一。*有关保留字符的信息，请参阅如何...正确命名信道、设备、标记和标记组。*
提示：如果应用程序最适合使用名称相同的标记块，请使用标记组分隔标记。*有关详细信息，请参阅标记组属性。*

"说明"(Description)：输入字符串来表示标记中的可用数据。标记名称的长度不能超过 256 个字符。尽管使用长的描述性名称通常是一个好方法，但在浏览 OPC 服务器的标记空间时，一些 OPC 客户端应用程序的显示窗口可能不够大。标记名称是 OPC 浏览数据标记名称的一部分，在指定设备分支或标记组分支内必须唯一。*有关保留字符的信息，请参阅如何...正确命名信道、设备、标记和标记组。*
提示：如果应用程序最适合使用名称相同的标记块，请使用标记组分隔标记。*有关详细信息，请参阅标记组属性。*

"地址"(Address)：输入目标标记的驱动程序地址。地址格式以驱动程序协议为基础。地址最多可为 128 个字符。
提示：有关地址的输入方式的提示，请单击浏览 (...) 按钮。 如果驱动程序接受输入的地址，则不会显示任何消息。弹出菜单会通知有何错误。某些错误与数据类型选择有关，而不是地址字符串。

"说明"(Description)：对标记应用备注。可在说明中输入的字符串长度最多为 255 位。使用支持"数据访问 2.0"标记属性的 OPC 客户端时，可从标记的项目"说明"(Description) 属性访问说明属性。

"数据类型"(Data Type)：指定在物理设备中找到此标记时，该标记的数据格式。在大多数情况下，这也是数据返回客户端时的格式。数据类型设置是通信驱动程序如何读取并将数据写入设备的重要组成部分。对于多数驱动程序，数据特定部分的数据类型完全固定，而且驱动程序知道在读取设备数据时需使用何种格式。但是，在某些情况下，对设备数据的解释很大程度上由用户决定。以一个使用 16 位数据寄存器的设备为例。通常会指明数据为"短整型"或"字"。许多基于寄存器的设备还支持跨越两个寄存器的值。在这些情况下，双寄存器值可能是长整型、双字型或浮点型。如果正在使用的驱动程序支持此级别的灵活性，用户则须告知驱动程序如何读取此标记的数据。通过选择相应的数据类型来告知驱动程序读取一个、两个、四个、八个或十六个寄存器或者可能的布尔值。驱动程序控制所选取的数据格式。

- **默认** - 使用驱动器默认的数据类型
- **布尔型** - true 或 false 的二进制值
- **字符** - 有符号的 8 位整数数据
- **字节** - 无符号的 8 位整数数据
- **短整型** - 有符号的 16 位整数数据
- **字** - 无符号的 16 位整数数据
- **长整型** - 有符号的 32 位整数数据
- **双字型** - 无符号的 32 位整数数据
- **双长整型** - 有符号的 64 位整数数据
- **四字型** - 无符号的 64 位整数数据
- **浮点型** - 32 位实数值 IEEE-754 标准定义
- **双精度** - 64 位实数值 IEEE-754 标准定义
- **字符串** - 空终止 Unicode 字符串
- **BCD** - 两个字节封装的 BCD 值的范围是 0-9999
- **LBCD** - 压缩为四个字节的 BCD 值的范围是 0-99999999
- **日期** - 请参阅 Microsoft® 知识库。

"客户端访问"：指定标记是否为**"只读"**或 **"读/写"**。通过选择"只读"，可以防止客户端应用程序更改此标记中包含的数据。通过选择**"读/写"**，允许客户端应用程序根据需要更改此标记的值。选择**"客户端访问"**还会影响标记在 OPC 客户端浏览空间中的显示方式。许多 OPC 客户端应用程序允许基于属性筛选标记。更改此标记的访问方法可能会更改此标记在 OPC 客户端浏览空间中的显示方式和时间。

"扫描速率"：指定与非 OPC 客户端配合使用时此标记的更新间隔。OPC 客户端可以通过使用更新速率，即所有 OPC 组的一部分，来控制扫描数据的速率。非 OPC 客户端通常不具有此特殊功能。服务器用于指定非 OPC 客户端中每个标记基础的标记更新速率。使用扫描速率，用户可以定制服务器的带宽要求以适应应用程序的需要。例如，如果需要读取变更非常慢的数据，则不必非常频繁地读取其值。使用扫描速率，则可强制此标记以较低的速率进行读取，从而减少对通信信道的需求。有效范围为 10 到 99999990 毫秒 (ms)，其中增量为 10 毫秒。默认值为 100 毫秒。

- 服务器全天在线运行时，可以随时更改这些属性。对标记属性的更改立即生效；但是，已经连接到此标记的 OPC 客户端在发布并尝试重新获取标记之前不会受到影响。使用“用户管理器”可限制对服务器功能的访问权限，并防止操作员更改属性。

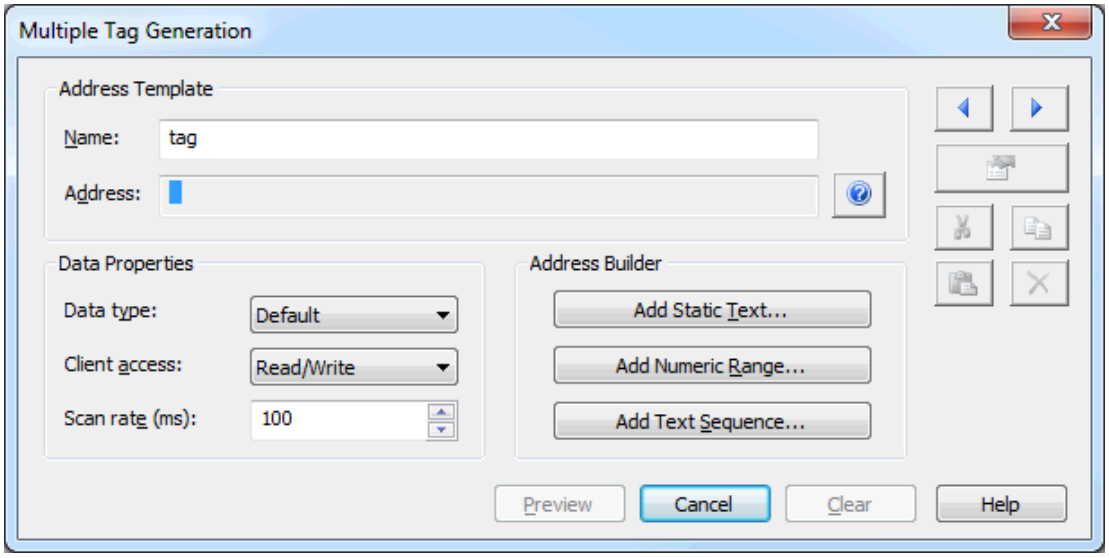
多标记生成

“多标记生成”工具使用用户定义的驱动程序命名法来动态创建标记。它允许使用各种地址格式 (例如，使用十进制、十六进制和八进制数字系统的范围)。为避免数据重叠，“标记生成器工具”还可以按用户定义的数据类型递增。

有关特定对话框的信息，请从下表中选择一个链接：

添加数字范围
添加静态文本
添加文本序列
多标记生成预览
标记名称属性

多标记生成



地址模板

“名称”: 输入用户定义的标记名称。

“地址”: 验证通过“地址构建器”部分中定义的选项生成的标记地址。

数据属性

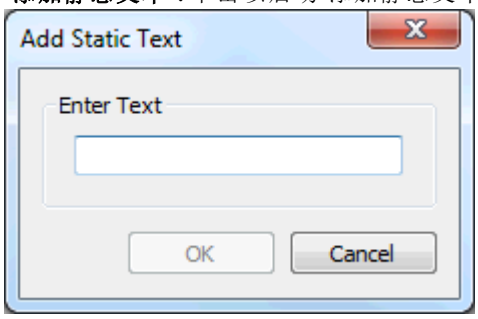
“数据类型”: 选择适用于所有生成标记的数据类型。根据驱动程序所支持的本机接口，数据类型可覆盖最后一个元素的“添加数字范围”属性的默认增量。默认设置为“默认”。

“客户端访问”: 从“只读”或“读/写”选择标记的权限设置。默认设置为“只读”。

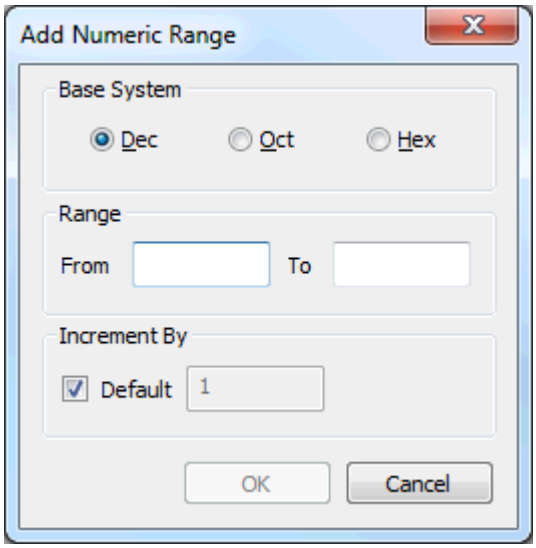
“扫描速率”: 指定扫描标记的频率。有效范围为 10 至 99999990 毫秒。默认设置为 100 毫秒。

地址构建器

“添加静态文本”：单击以启动“添加静态文本”对话框，可在其中输入单行文本。

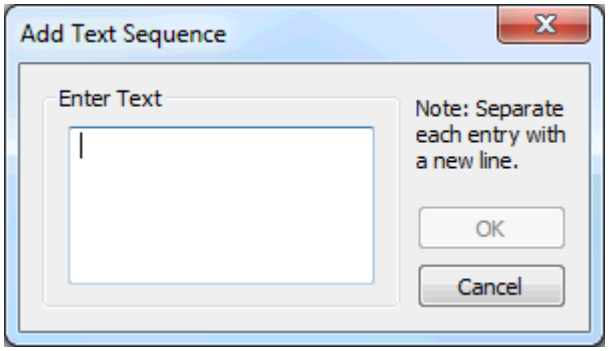


“添加数字范围”：单击以启动“添加数字范围”对话框。



- **“基本系统”**选择“基本系统”格式：十进制、八进制或十六进制。默认设置为“十进制”。
- **“范围”**在“自”和“至”字段中输入数字范围的起始值和终止值。
- **“增量”**如果不使用“默认值”(增量为 1)，用户可指定自定义增量值。范围根据所选的“基本系统”增加。

“添加文本序列”：单击以启动“添加文本序列”对话框，可在其中创建多个字符串。每个字符串均独立于列表中指定的其他字符串插入。

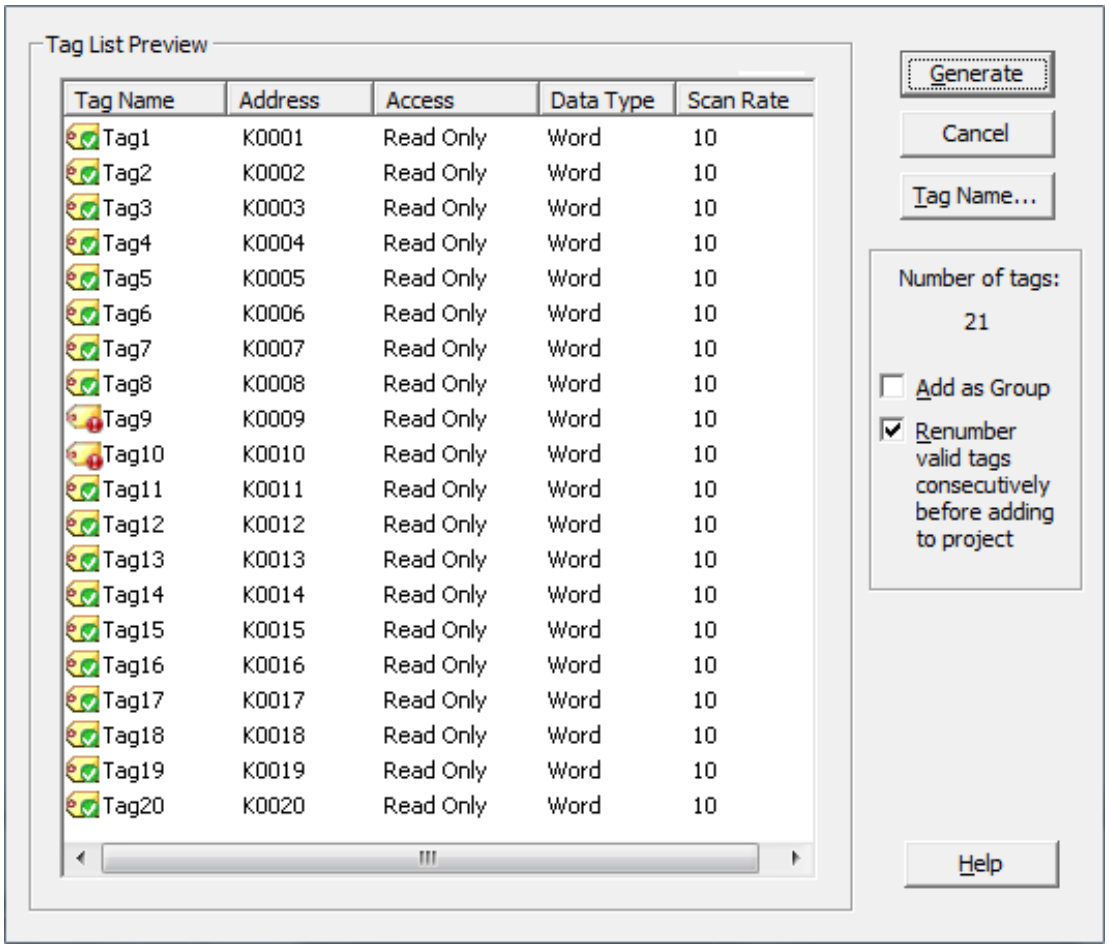


提示

1. 要启用右侧的“编辑”图标，请突出显示标记地址语法元素的一部分。
2. “提示”图标可打开“地址说明”内的帮助文件。

“预览”：单击以生成已生成标记的测试视图。

多标记生成预览



"生成"：单击以将所有有效的标记发送到服务器以供插入。

"取消"：单击以拒绝对标记所做的任何更改并返回到先前的对话框。

"标记名称"：单击以调用"标记名称属性"对话框。

"添加为组"：启用以将标记添加到单个组织组中。默认设置为禁用状态。

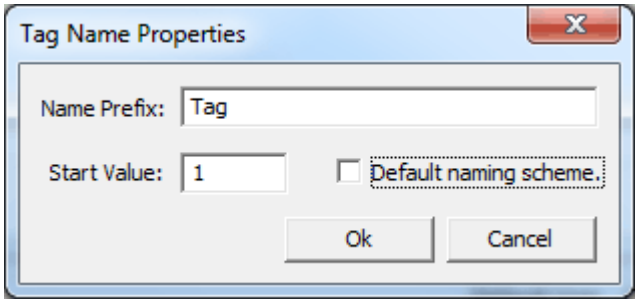
"在添加到项目之前连续重新编号有效标记"：启用以在添加到项目之前连续重新编号标记。默认设置为启用状态。

注意：显示绿色复选标记的标记为有效标记。显示红色叹号 (!) 的标记为无效标记。

标记名称属性

"标记生成器工具"包括自定义命名方案选项，该选项可供用户同时为所有标记指定名称前缀和数字后缀。每个标记的数字后缀自动递增，允许用户为标记创建自定义名称以提高可读性。分配的标记名称可能会在生成后更改。如果用户未通过"标记名称属性"对话框定义自定义名称，则会针对每个生成标记执行默认命名方案。

注意：在"生成"对话框中更改命名方案的用户在返回到"标记重复"对话框之前，可以选择保存下一次生成标记列表时的命名方案。



"名称前缀"：输入自定义名称前缀 (标记名称前面追加的字母)。

"起始值"：指定每个标记递增的第一个数字值。

"默认命名方案"：启用后，使用默认命名方案。默认设置为禁用状态。

另请参阅：生成多个标记

标记属性 - 缩放

此服务器支持标记缩放，允许将设备的原始数据缩放到应用程序的合适范围。

"类型"：选择缩放原始数据的方法。选择**"线性"**、**"平方根"**或**"无"**以禁用。缩放类型公式如下所示。

类型	缩放值公式
线性	(((ScaledHigh - ScaledLow)/(RawHigh - RawLow))*(RawValue - RawLow)) + ScaledLow
平方根	(Square root ((RawValue - RawLow)/(RawHigh - RawLow))*(ScaledHigh - ScaledLow)) + ScaledLow

"原始低"(Raw Low)：指定设备数据范围的下限。有效范围取决于原始标记数据类型。例如，如果原始值为短整型，则原始值的有效范围为 -32768 到 32767。

"原始高"：指定设备数据范围的上限。"原始高"值必须大于"原始低"值。有效范围取决于原始标记数据类型。

"缩放数据类型"(Scaled Data Type)：为要进行缩放的标记选择数据类型。可将数据类型设置为任何有效的 OPC 数据类型，包括原始数据类型，例如，短整型，可将其设置为具有长整型数据类型的工程值。默认的缩放数据类型为双精度。

"缩放低"：指定生成的有效缩放数据值的范围下限。有效范围取决于标记数据类型。

"缩放高"：指定生成的有效缩放数据值的范围上限。有效范围取决于标记数据类型。

"钳位低"(Clamp Low)：选择**"是"(Yes)** 以防止结果数据超出指定范围的下限。选择**"否"**以允许数据超出已建立的范围。

“钳位高”：选择**“是”**以防止结果数据超出指定范围的上限。选择**“否”**以允许数据超出已建立的范围。

“求反值”(Negate Value)：选择**“是”(Yes)** 以强制对结果值进行求反，然后传递给客户端。选择**“否”**可将未修改的值传递给客户端。

服务器支持 2.0 数据访问规范中所提供的 OPC 标记属性。如果所使用的 OPC 客户支持这些属性，那么它可以使用缩放设置自动配置对象 (例如，用户输入对象或显示) 的范围。使用用户管理器可限制对服务器功能的访问权限，以防止任何未经授权的操作员对这些属性进行更改。

动态标记

动态标记寻址是另一种定义标记的方法，仅允许用户在客户端应用程序中定义标记。在这种情况下，用户只需在直接访问设备驱动器地址的客户端中创建一个标记项，而无需在对服务器中创建的另一个标记项进行寻址的客户端中创建标记项。在客户端连接时，服务器会为该位置创建一个虚拟标记并自动开始数据扫描。

要指定可选的数据类型，请在 '@' 符号后面附加以下字符串：

- BCD
- 布尔型
- 字节
- 字符
- 双精度
- 双字型
- 浮点型
- LBCD
- LLong
- 长整型
- 四字型
- 短整型
- 字符串
- 字

如果省略了数据类型，驱动程序将根据所参考的设备和地址选择默认数据类型。所有位置的默认数据类型都记录在每个单独的驱动程序帮助文档中。如果指定的数据类型对于设备位置无效，则服务器会拒绝标记，并且事件日志中会发布错误。

使用动态寻址的 OPC 客户端示例

在 Simulator 设备上扫描 16 位位置 "R0001"。以下动态标记示例假定创建的项目是示例的一部分。

1. 启动 OPC 客户端应用程序并连接到服务器。
2. 使用 Simulator 驱动程序创建一个信道并命名为 "Channel1"。然后，创建一个设备并命名为 "Device1"。
3. 在客户端应用程序中，将一个项的名称定义为 "Channel1.Device1.R0001@Short"。
4. 客户端项目将自动开始接收数据。Simulator 设备中地址 R0001 的默认数据类型是字。为将此覆盖，已经附加了 @Short 以选择短整型数据类型。

注意：当在 OPC 客户端应用程序中利用动态标记时，通常不需要使用 @[数据类型] 修饰符。注册特定数据项的链路时，OPC 客户端可以指定所需的数据类型作为请求的一部分。如果通信驱动程序支持，可使用 OPC 客户端指定的数据类型。当需要确保通信驱动程序根据需要准确解释一个数据片段时，@[数据类型] 修饰符会很有用。

非 OPC 客户端示例

非 OPC 客户端可以通过附加 @[更新速率] 覆盖每个标记的更新速率。

例如，附加：
<DDE 服务名称>|_ddedata!Device1.R0001@500 只覆盖更新速率。
<DDE 服务名称>|_ddedata!Device1.R0001@500,Short 覆盖更新速率和数据类型。

提示：

1. 服务器可以为项目中的每个设备创建一个特殊布尔型标记，客户端可用其来确定该设备是否运行正常。要使用此标记，请将链接中的项指定为“错误”。如果设备通信正常，标记的值为零；否则，标记的值为一。
2. 如果将设备地址用作链接的项，以便地址与服务器中用户定义标记的名称相匹配，链接会参考用户定义标记所指向的地址。
3. 静态标记必须用于缩放服务器中的数据。

另请参见：
静态标记 (用户定义)
设计项目：添加用户定义标记

静态标记 (用户定义)

使用服务器获取从设备传输至客户端应用程序的数据可以采用最常用的方法，该方法具有两个要求。用户必须首先在服务器中定义一组标记，方法为将已分配标记名称用作客户端与服务器之间的每个链接的项。使用此方法的主要优点是，所有用户定义标记均可用于浏览大部分 OPC 客户端。在确定是否创建静态标记之前，请确保客户端可以浏览或从服务器导入标记。

提示：用户定义标记支持缩放。

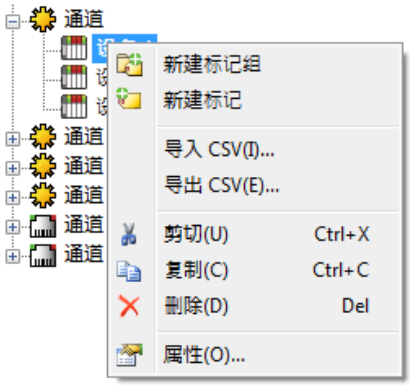
什么是标记组？

此服务器允许为项目添加标记组。标记组用于将 OPC 数据布局调整为符合应用程序需求的逻辑分组。标记组允许在同一设备中添加多组相同标记：这对于单个设备处理大量相似机器段来说非常方便。

标记组属性

从 OPC 客户端的角度来看，标记组允许用户将 OPC 数据分隔成较小的标记列表，以使在浏览服务器时查找特定标记更为轻松。下图使用提供的 OPC Quick Client 创建 Cell1 和 Cell2 标记组并简化 OPC 客户端浏览。

要向项目添加新的标记组，请右键单击现有设备或标记组分支，然后从上下文菜单中选择**“新建标记组”(New Tag Group)**。或者，单击现有设备或标记组分支，然后单击工具栏上的“新建标记组”(New Tag Group) 图标。



标记组可在设备级别以下的任意级进行添加，并可以一起嵌套多个标记组以满足应用程序的需求。如上方 OPC Quick Client 对话框所示，完全限定的 OPC 项目路径为 "Channel1.Device1.Machine1.Cell1.Tag1"。对于此 OPC 项目，"Machine1" 和 "Cell1" 段为嵌套的标记组。

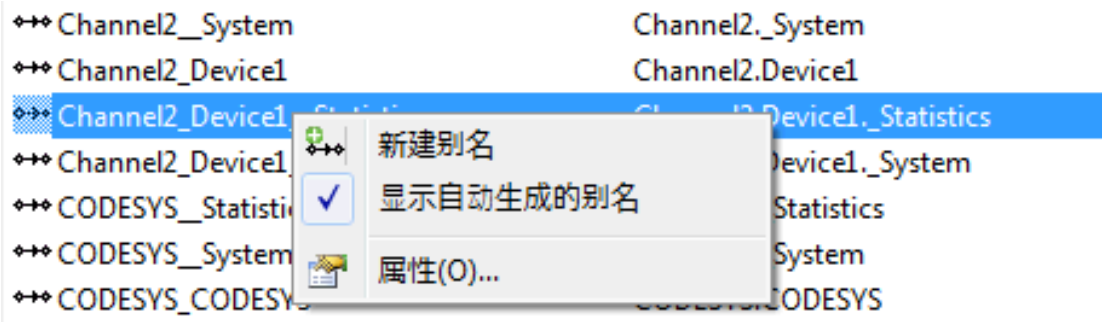
注意：服务器全天在线运行时，可以随时更改这些属性。对标记所做的任何更改会立即生效。如果名称已更改，则已将该标记组用作 OPC 项目请求一部分的 OPC 客户端在发布并尝试重新获取该项目之前不会受到影响。可通过 OPC 客户端立即浏览新添加到项目的标记组。使用"用户管理器"可限制对服务器功能的访问权限，以防止操作员更改属性。

什么是别名映射？

"别名映射"提供与传统服务器应用程序向后兼容的机制和为复杂标记参考分配简单别名的方法。这对于限制标记地址路径大小的客户端应用程序尤其有用。虽然服务器最新版本会自动创建别名映射，但是用户可以添加自己的别名映射条目以补充由服务器创建的条目。用户还可对服务器创建的别名进行筛选，使其仅对用户自己可见。

可通过右键单击树状视图窗格中的目标别名导入及导出别名映射元素。

可通过右键单击详细信息窗格中的目标别名添加、编辑和删除别名映射元素。



注意：启用后，**"显示自动生成的别名"**会显示由服务器自动创建的别名映射。

另请参阅：如何创建和使用别名

别名属性

别名映射可用于将别名分配给可在客户端应用程序中使用的复杂标记参考。通过输入别名并单击所需设备名称或组名称，即可构建别名。

"名称"(Name)：指定长度不超过 256 个字符的别名。该名称在别名映射中必须唯一。有关保留字符的信息，请参阅如何正确命名信道、设备、标记和标记组。

"说明"(Description)：输入此别名的说明以阐明数据源和报告 (可选)。

"映射到"(Mapped to)：指定或浏览至别名所在位置。由于别名映射不允许从别名表浏览标记项，因此需要创建短别名以替换指向标记的地址。这样，可以更轻松地在不支持标记浏览的客户端应用程序中对项进行寻址。

"扫描速率覆盖"(Scan Rate Override)：指定要应用于使用此别名映射条目访问的所有非 OPC 标记的更新速率。有效范围为 0 到 99999990 毫秒。默认值为 0 毫秒。

提示：此设置等同于众多仅 DDE 服务器中的主题更新速率。
注意：设置为 0 毫秒时，服务器将遵循在单个标记级别设置的扫描速率。

什么是事件日志？

“事件日志”显示错误、警告、信息或安全事件的日期、时间和源。有关详细信息，请从下表中选择一个链接。

事件日志选项
事件日志设置

事件日志

用户可指定在事件日志中显示的事件的类型。当前有四种类型可以记录的事件：错误事件、警告事件、信息事件和安全事件。事件的说明如下所示：

信息：提供状态和数据，但无需用户与之交互或进行修复的消息，例如成功连接或数据收集。

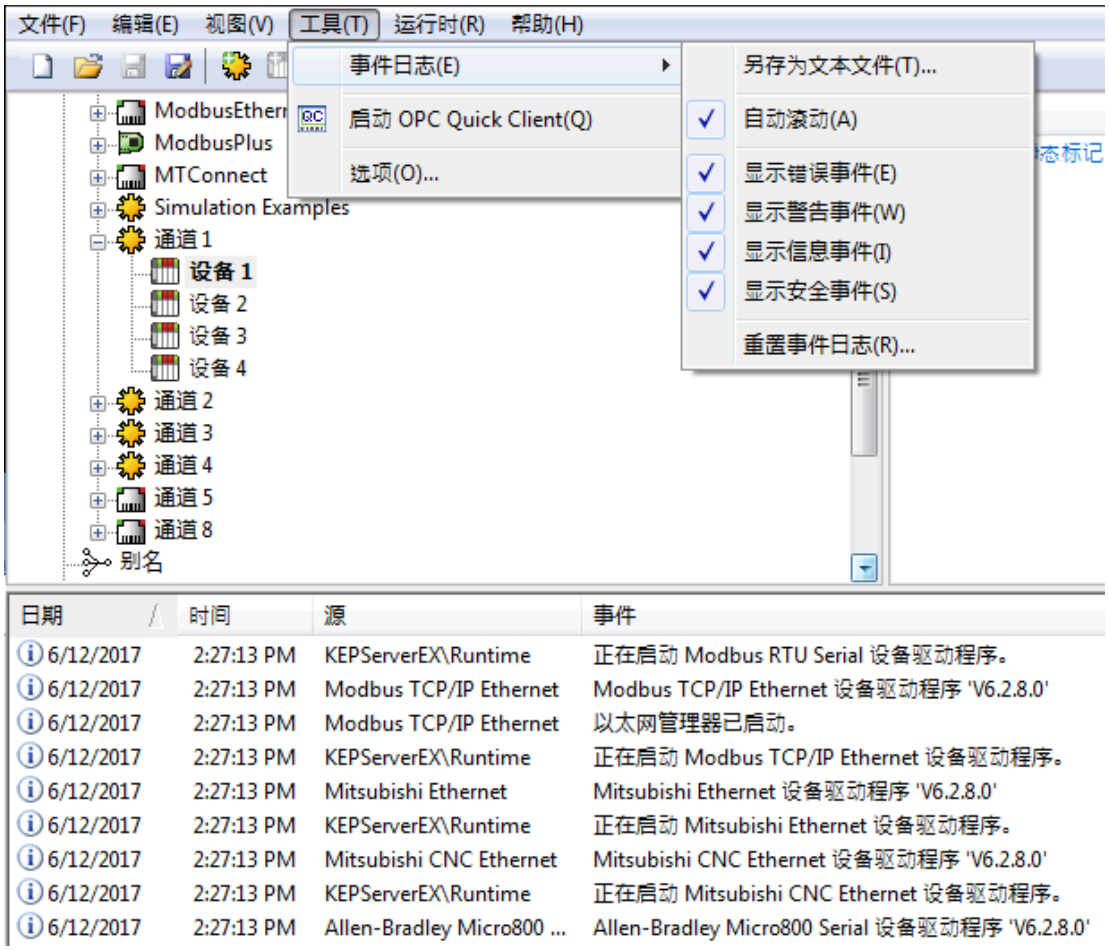
安全：从安全角度提示您注意不是最佳做法的情况的消息，如以默认用户而不是使用有效凭据登录的用户身份来运行软件。

警告：指示问题不需要用户与之交互，但可能会导致意外结果的消息，如设备未响应。

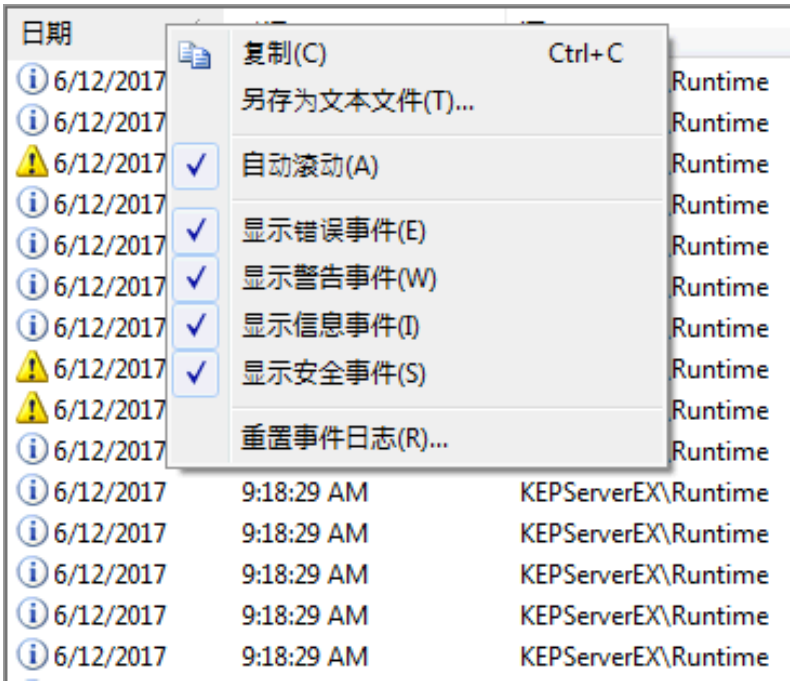
错误：提醒用户通常应进行研究和修复才能得到最佳结果的失败或问题之类的消息。

注意：要访问配置客户端中的事件类型，请单击**“工具”|“事件日志”**。或者，右键单击“事件日志”显示中任意位置。

工具菜单



右键单击



注意：如果不存在任何机制来保护事件日志系统的内容，则该事件日志系统将是无用的。如果操作员可以更改这些属性或重置日志，则会失去它存在的目的。利用“用户管理器”来限制操作员可以访问的功能并防止这些操作发生。

另请参阅：*设置-事件日志*

标记管理

服务器的用户定义标记管理功能可创建标记数据库结构，以适应每个应用程序的特定性质。用户可以在设备对设备的基础上定义多个标记组来分隔标记数据，还可通过拖放编辑轻松添加大量标记。CSV 导入和导出还允许在任何应用程序中进行标记编辑。与其他服务器功能一样，可以随时向应用程序添加新标记。

自动标记数据库生成

OPC 服务器可针对所选通信驱动程序自动生成标记的功能使 OPC 技术距即插即用操作更近一步。可以直接从设备读取标记信息，并且还可以利用存储的标记数据生成标记。在任一情况下，用户均不再需要将 OPC 标记手动输入到服务器。

系统标记

系统标记向客户端应用程序提供常规错误反馈，允许在设备主动收集数据时进行操作控制，并允许从 OPC 客户端应用程序更改信道或设备的标准属性。信道或设备级别的可用系统标记数量取决于正在使用的驱动程序的性质。

注意：系统标记可以根据其目的，按照状况和控制或属性操作进行分组。

属性标记

属性标记是任何“数据访问”客户端均可访问的附加标记，方法是：向完全限定的标记地址附加属性名称。当使用支持项目浏览的 OPC 客户端时，用户可以通过打开 OPC DA 设置下的**“当客户端浏览服务器时，包括标记属性”(Include tag properties when a client browses the server)** 浏览标记属性。有关详细信息，请参阅项目属性 - OPC DA。

统计信息标记

统计信息标记用于向客户端应用程序提供有关服务器信道通信操作的反馈。启用诊断时，有七个内置的“统计信息”标记可供使用。有关详细信息，请参阅 OPC 诊断查看器。

调制解调器标记

调制解调器标记用于配置调制解调器属性和监控调制解调器状况。仅在**“信道属性”(Channel Properties)** 中的**“连接类型”(Connection Type)** 设为**“调制解调器”(Modem)** 时可用。有关详细信息，请参阅信道属性 - 串行通信。

通信序列化标记

驱动程序通信通常跨多个信道同时发生，因此会产生较高的数据吞吐量。然而，在某些应用程序中，要求每次仅允许一个信道进行通信。通信序列化则提供此类支持。通信序列化标记用于配置和监控信道的序列化状况。功能及其标记仅供特定驱动程序使用。有关详细信息，请参阅驱动程序的帮助文档。

CSV 导入和导出

此服务器可导入和导出逗号分隔变量 (CSV) 文件中的标记数据以在应用程序中快速创建标记。CSV 函数仅在选择设备或标记组时可用。

注意：有关可将哪些字符指定为变量的信息，请参阅选项 - 常规。

要跳转到特定部分，请从下表中选择一个链接。

导出服务器标记列表
将服务器标记列表导入到服务器
使用其他字符作为分隔符

创建模板

创建并导入 CSV 文件的最简单方法是创建一个模板。有关详细信息，请参阅以下说明。

1. 首先，请单击**“文件”|“导出 CSV”**。定义项目的通道和设备。
2. 定义每个设备的标记。
3. 将各设备或标记组导出为 CSV 文件。

4. 在支持 CSV 文件的电子表格应用中使用此模板并根据需要修改文件。

注意：生成的 CSV 文件可保存到磁盘并重新导入到同一 (或新的) 设备或标记组下的服务器中。

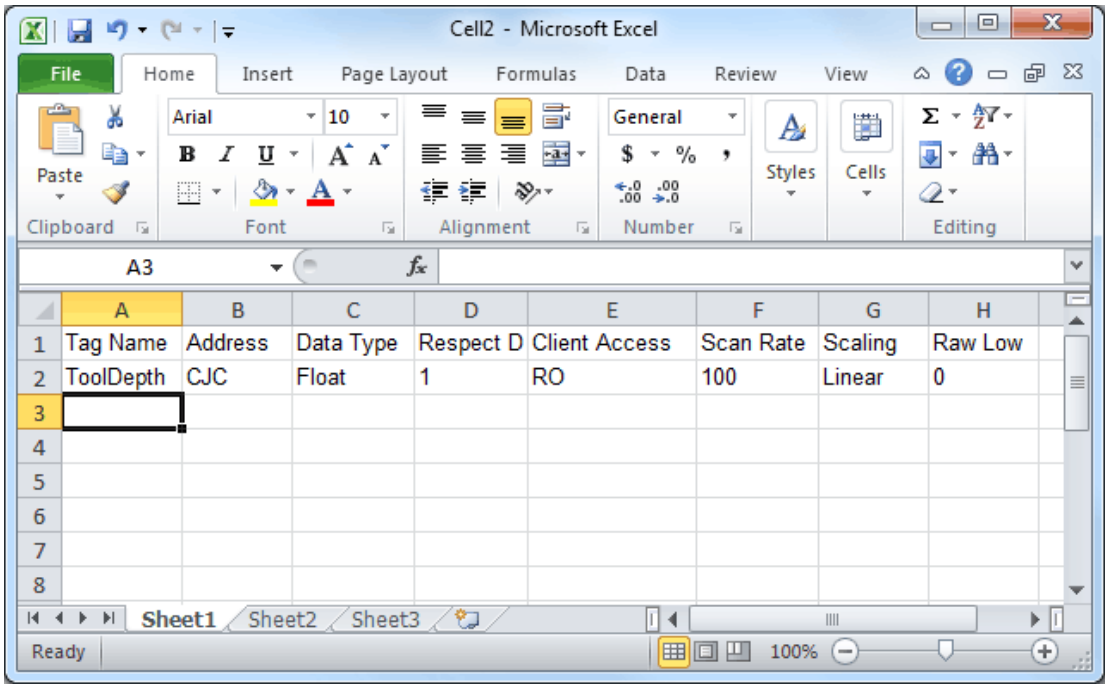
导出服务器标记列表

导出服务器标记列表时会生成一个包含标题记录的 .CSV 文本文件，标题记录下依次记录了在所选设备或标记组下定义的各个标记。标题记录中包含以下字段：

- **"标记名称"**：在 OPC 客户端引用的标记名称。
  标记名称可能包含与标记名称以句点分隔的组名称前缀。例如，标记名称 "Group1.Tag1" 会创建一个名为 "Group1" 的组，其中包含 "Tag1"。
- **"地址"**：标记引用的设备位置。
- **"数据类型"**：用于标记的数据类型，如服务器标记的数据类型下拉列表中所示。
- **"相应数据类型"**：这将强制标记遵照其定义的数据类型，而不是 OPC 客户端请求 (1, 0)。
- **"客户端访问"**：读取/写入访问 (只读和读/写)。
- **"扫描率"**：当与大多数非 OPC 客户端搭配使用时，标记地址的扫描率以毫秒为单位。
- **"缩放"**：缩放模式 (无、线性和平方根)。
- **"原始低"**：原始值较低。
- **"原始高"**：原始值较高。
- **"缩放低"**：缩放后的低值。
- **"缩放高"**：缩放后的高值。
- **"缩放数据类型"**：应用缩放后，用于标记的数据类型。
- **"钳位低"**：强制生成的缩放值保持在"缩放低"的限制内 (1, 0)。
- **"钳位高"**：强制生成的缩放值保持在"缩放高"的限制内 (1, 0)。
- **"工程单位"**：单位字符串。
- **"说明"**：标记的说明。
- **"求反值"**：应用缩放时，在传递给客户端之前，对生成的值取反 (1, 0)。

注意：每条标记记录中含有各个字段的数据。

Microsoft Excel 是在服务器外部编辑大量标记的极佳工具。导出模板 CSV 文件后，可直接将其加载到 Excel 进行编辑。加载到 Excel 的 CSV 文件如下图所示：



将 CSV 标记列表导入到服务器

标记列表经过编辑后，可通过单击**“文件”|“导入 CSV”**将其重新导入服务器。此选项仅在选择设备或标记组时可用。

使用其他字符作为分隔符

利用不使用逗号或分号分隔符的 CSV 文件时，用户应进行以下操作：

- 将项目保存为 xml 格式。然后，对 XML 文件执行大量配置，而不使用 CSV。
- 在 CSV 文件中对分隔符执行搜索替换，将分隔符替换为逗号或分号。必须将 OPC 服务器正在使用的分隔符 (逗号或分号) 设置为替换字符。

另请参阅：选项 - 常规

系统标记

系统标记向客户端应用程序提供常规错误反馈，允许在设备主动收集数据时进行操作控制，并允许 OPC 客户端应用程序在需要时更改通道或设备的标准属性。

通道级别和设备级别的可用系统标记数量取决于正在使用的驱动程序的性质。此外，应用程序级的系统标记允许客户端应用程序监控服务器的状况。系统标记可以根据其目的，按照状况和控制或属性操作进行分组。说明如下：

- **状况标记**：状况标记为只读标记，提供有关服务器操作的数据。
- **参数控制标记**：参数控制标记可用于修改服务器应用程序的操作特征。为 OPC 应用程序提供了大量的灵活性。通过使用属性控制标记，用户可以通过切换通信链路或更改目标设备的设备 ID 来实现冗余。用户还可以通过特殊监控屏幕提供对标记的访问，如有需要，设备工程师可以通过该监控屏幕对服务器的通信参数进行更改。

下表包括对以下内容的说明：

应用程序级系统标记
串行端口驱动程序的通道级系统标记
以太网驱动程序的通道级系统标记
串行和以太网驱动程序的设备级系统标记

应用程序级系统标记

语法示例：*<通道名称>.<设备名称>._System._ActiveTagCount*

标记	类	说明
_ActiveTagCount	状况标记	_ActiveTagCount 标记表示当前在服务器中处于活动状态的标记的数量。 此为只读标记。
_ClientCount	状况标记	_ClientCount 标记表示当前连接至服务器的客户端的数量。 此为只读标记。
_Date	状况标记	_Date 标记表示服务器运行所在系统的当前日期。此字符串的格式由操作系统日期/时间设置定义。 此为只读标记。
_DateTime	状况标记	_DateTime 标记表示服务器运行所在系统的 GMT 日期和时间。该字符串的格式为 '2004-05-21T20:39:07.000'。 此为只读标记。
_DateTimeLocal	状况标记	_DateTimeLocal 标记表示服务器运行所在系统的本地日期和时间。该字符串的格式为 '2004-05-21T16:39:07.000'。 此为只读标记。
_Date_Day	状	_Date_Day 标记表示服务器运行所在系统的每月当天。

标记	类	说明
	况标记	此为只读标记。
_Date_DayOfWeek	状况标记	_Date_Day 标记表示服务器运行所在系统的每周当天。该字符串的格式为数字 0 (星期日) 到 6 (星期六)。 此为只读标记。
_Date_Month	状况标记	_Date_Month 标记表示服务器运行所在系统的当前月份。该字符串的格式为数字 (如 9，而不是 "September")。 此为只读标记。
_Date_Year2	状况标记	_Date_Year2 标记表示服务器运行所在系统的当前年份的后两位。 此为只读标记。
_Date_Year4	状况标记	_Date_Year4 标记表示服务器运行所在系统的当前年份。 此为只读标记。
_ExpiredFeatures	状况标记	_ExpiredFeatures 标记提供所有限时使用功能均已过期的服务器功能列表。这些功能不再运行。 此为只读标记。
_FullProjectName	状况标记	_FullProjectName 标记表示当前加载项目的完全限定的路径和文件名。 此为只读标记。
_IsDemo	状况标记	由于在版本 6.0 或更高版本中运行时不会进入“限时”模式，因此 _IsDemo 标记不再可用。请参阅 _TimeLimitedFeatures、_LicensedFeatures 和 _ExpiredFeatures 标记来监控服务器功能的状况。
_LicensedFeatures	状况标记	_LicensedFeatures 标记提供具有有效许可证的所有正在使用的服务器功能列表。这些功能不受时间限制并会在任何限时功能过期后继续正常运行。 此为只读标记。
_OpcClientNames	状况标记	_OpcClientNames 标记是“字符串数组”，其中列出了所有连接至服务器并通过 IOPCCommon::SetClientName 方法注册其名称的 OPC 客户端。 此为只读标记。
_ProductName	状况标记	_ProductName 标记表示底层通信服务器的名称。 此为只读标记。
_ProductVersion	状况标记	_ProductVersion 标记表示底层通信服务器的版本。 此为只读标记。
_ProjectName	状况标记	_ProjectName 标记表示当前加载的项目文件名称，并不包括路径信息。 此为只读标记。
_ProjectTitle	状况标记	_ProjectTitle 标记为“字符串”标记，表示当前加载的项目的标题。 此为只读标记。

标记	类	说明
_Time	状况标记	_Time 标记表示服务器运行所在系统的当前时间。此字符串的格式由操作系统日期/时间设置定义。 此为只读标记。
_Time_Hour	状况标记	_Time_Hour 标记表示服务器运行所在系统的当前小时。 此为只读标记。
_Time_Hour24	状况标记	_Time_Hour24 标记表示表示服务器运行所在系统的当前小时 (24 小时格式)。 此为只读标记。
_Time_Minute	状况标记	_Time_Minute 标记表示服务器运行所在系统的当前分钟。 此为只读标记。
_Time_PM	状况标记	_Time_PM 标记表示服务器运行所在系统的当前上午/下午 (AM/PM) 状况。为布尔型标记：0 (False) 表示上午 (AM)，1 (True) 表示下午 (PM)。 此为只读标记。
_Time_Second	状况标记	_Time_Second 标记表示服务器运行所在系统的当前秒。 此为只读标记。
_TimeLimitedFeatures	状况标记	_TimeLimitedFeatures 标记提供所有限时但时间有剩余 (以秒为单位) 的服务器功能的列表。剩余时间到期后，功能将停止运行。 此为只读标记。
_TotalTagCount	状况标记	_TotalTagCount 标记表示当前可访问的标记总数。这些标记可以处于活动或非活动状态。 **注意**：此计数不代表项目中配置的标记数量。 此为只读标记。

串行端口驱动程序的通道级系统标记

语法示例：*<通道名称>._System._BaudRate*

标记	类	说明
_AvailableNetworkAdapters	状况标记	_AvailableNetworkAdapters 标记列出了可用的 NIC，其中包括唯一 NIC 卡和分配有多个 IP 的 NIC。此外，此标记还会显示任何处于活动状态的 WAN 连接，例如拨号连接。此标记作为字符串标记提供，可用于确定可在此 PC 上使用的网络适配器。返回的字符串将包含所有 NIC 名称及其 IP 分配。分号将用于分隔每个唯一 NIC，以便可在 OPC 应用程序中分析这些名称。对于串行驱动程序，仅在选择“以太网封装”时使用此标记。 此为只读标记。
_BaudRate	参数控制标记	_BaudRate 标记允许随意更改驱动程序的波特率。_BaudRate 标记被定义为长整型值，因此应以此格式写入新波特率。有效波特率如下：300、600、1200、2400、4800、9600、19200、38400、56000、56700、115200、128000 和 256000。 此为读/写标记。
_ComId	参	_ComId 标记允许随意更改驱动程序的通信端口选择。作为字符

标记	类	说明
	数控制标记	串标记，所需通信端口必须作为字符串值写入标记，可使用下列可能的选择：COM 1、COM 2、COM 3、COM 4、- - -、COM 16 和“以太网封装”。选择“以太网封装模式”时，用户还需要设置远程终端服务器的 IP 号。该操作在设备级完成，如下所示。 此为读/写标记。
_DataBits	参数控制标记	_DataBits 标记允许随意更改驱动程序的数据位。_DataBits 标记被定义为有符号 8 位值。有效数据位选择包括 5、6、7 和 8。 此为读/写标记。
_Description	状况标记	_Description 标记指示所参考通道的当前用户定义的文本描述。 此为只读标记。
_EnableDiagnostics	参数控制标记	_EnableDiagnostics 标记允许启用和禁用驱动程序的诊断系统。诊断系统在启用时会稍微加大驱动程序的负担。因此，服务器允许启用或禁用诊断，以提高驱动程序的性能。禁用时，“诊断”标记将不可用。有关详细信息，请参阅统计信息标记。 此为读/写标记。
_EncapsulationPort	参数控制标记	_EncapsulationProtocol 标记用于控制以太网连接的目标。有效范围为 0 到 65535。 此为读/写标记。
_EncapsulationProtocol	参数控制标记	_EncapsulationProtocol 标记控制用于以太网连接的协议。选项包括 TCP/IP 和 UDP。 此为读/写标记。
_FloatHandlingType	参数控制标记	_FloatHandlingType 标记允许更改当前通道级浮点处理。该标记存在于通道级 _System 文件夹中。有关详细信息，请参阅通道属性 - 高级。 此为读/写标记。
_FlowControl	参数控制标记	_FlowControl 标记允许随意更改驱动程序的流量控制设置。作为字符串标记，所需流量控制设置须以此格式写入为标记。流量控制的可能选择包括：“无”、DTR、RTS、"DTR, RTS"、“始终 RTS”和“RTS 手动”。并非所有驱动程序都支持“RTS 手动”操作模式。 此为读/写标记。
_InterDeviceDelayMS	参数控制标记	_InterDeviceDelayMS 标记指定接收到同一通道上当前设备发出的数据后，通道向下一个设备发送请求的延迟时间量。有效范围为 0 至 60000 毫秒。默认设置为 0。 **注意：**此标记仅在使用利用“设备间延迟”的协议的通道上可用。 此为读/写标记。
_NetworkAdapter	参数控	_NetworkAdapter 标记允许驱动程序随意更改当前正在使用的 NIC 适配器。作为字符串标记，新近所需的 NIC 适配器名称须以字符串格式写入此标记。所写字符串必须与所需 NIC 的具体说明

标记	类	说明
	制标记	相符，才能使更改生效。NIC 名称可以从上面列出的 _AvailableNetworkAdapters 标记中获得。对于串行驱动程序，仅在选择“以太网封装”时使用此标记。 **注意**：更改 NIC 选择时，系统会强制驱动程序断开所有当前设备连接，并重新连接。 此为读/写标记。
_Parity	参数控制标记	_Parity 标记允许随意更改驱动程序的奇偶性。作为字符串标记，所需奇偶性设置必须作为字符串值写入标记，可使用下列可能的选择：“无”、“奇”、“偶”。 此为读/写标记。
_ReportComErrors	参数控制标记	_ReportComErrors 标记允许报告低级通信错误，例如，要启用或禁用奇偶性和帧错误。此标记被定义为布尔型标记，并可设置为 True 或 False。设置为 True 时，驱动程序将向服务器事件系统报告所有低级通信错误。设置为 False 时，驱动程序将忽略低级通信错误并且不会报告。如果通信事务包含错误，驱动程序仍会拒绝该事务。如果环境中包含大量电噪声，则可禁用此功能，以防在事件日志中填入错误消息。 此为读/写标记。
_RtsLineDrop	参数控制标记	_RtsLineDrop 标记允许在驱动程序尝试传送消息后针对用户选择的时间段降低“RTS 行”。此标记仅对支持“手动 RTS”模式的驱动程序有效。_RtsLineDrop 被定义为长整型值。有效范围为 0 至 9999 毫秒。“手动 RTS”模式专为与无线调制解调器配合使用而设计。 此为读/写标记。
_RtsLinePollDelay	参数控制标记	_RtsLinePollDelay 标记允许在驱动程序发送的每个消息后设置用户可配置的暂停。此标记仅对支持“手动 RTS”模式的驱动程序有效。_RtsLinePollDelay 被定义为长整型值。有效范围为 0 至 9999 毫秒。“手动 RTS”模式专为与无线调制解调器配合使用而设计。 此为读/写标记。
_RtsLineRaise	参数控制标记	_RtsLineRaise 标记允许在驱动程序尝试传送消息之前针对用户选择的时间段升高“RTS 行”。此标记仅对支持“手动 RTS”模式的驱动程序有效。_RtsLineRaise 被定义为长整型值。有效范围为 0 至 9999 毫秒。“手动 RTS”模式专为与无线调制解调器配合使用而设计。 此为读/写标记。
_SharedConnection	状况标记	_SharedConnection 标记表示正在与另一个通道共享的端口设置。 此为只读标记。
_StopBits	参数控制标记	_StopBits 标记允许随意更改驱动程序的停止位。_StopBits 标记被定义为有符号 8 位值。有效数据位选择为 1 和 2。 此为读/写标记。
_UnsolicitedEncapsulationPort	参数控	_UnsolicitedEncapsulationPort 标记控制处于打开状态而允许连接的以太网端口。有效范围为 0 到 65535。

标记	类	说明
	制标记	此为读/写标记。
_UnsolicitedEncapsulationProtocol	参数控制标记	_UnsolicitedEncapsulationProtocol 标记控制连接到“主动提供的封装端口”所用的以太网协议。选项包括 TCP/IP 和 UDP。 此为读/写标记。
_WriteOptimizationDutyCycle	参数控制标记	_WriteOptimizationDutyCycle 标记允许随意更改写入/读取率的占空比。占空比控制驱动程序将要对其所执行的每次读取进行的写入次数。_WriteOptimizationDutyCycle 被定义为无符号长整型值。有效范围为每次读取 1 至 10 次写入。有关详细信息，请参阅通道属性 - 写入优化。 此为读/写标记。

以太网驱动程序的通道级系统标记

语法示例：*<通道名称>._System._NetworkAdapter*

标记	类	说明
_AvailableNetworkAdapters	状况标记	_AvailableNetworkAdapters 标记列出了可用的 NIC，其中包括唯一 NIC 卡和分配有多个 IP 的 NIC。此外，此标记还会显示任何处于活动状态的 WAN 连接，例如拨号连接。此标记作为字符串标记提供，可用于确定可在此 PC 上使用的网络适配器。返回的字符串包含所有 NIC 名称及其 IP 分配。分号用于分隔每个唯一 NIC，以便可在 OPC 应用程序中分析这些名称。对于串行驱动程序，仅在选择“以太网封装”时使用此标记。 此为只读标记。
_Description	状况标记	_Description 标记指示所参考通道的当前用户定义的文本描述。 此为只读标记。
_EnableDiagnostics	参数控制标记	_EnableDiagnostics 标记允许启用和禁用驱动程序的诊断系统。诊断系统在启用时会稍微加大驱动程序的负担。因此，服务器允许启用或禁用诊断，以提高驱动程序的性能。禁用时，“诊断”标记将不可用。有关详细信息，请参阅统计信息标记。 此为读/写标记。
_EncapsulationPort	参数控制标记	_EncapsulationPort 标记控制用于以太网连接的端口。有效范围为 0 到 65535。 此为读/写标记。
_EncapsulationProtocol prop	参数控制标记	_EncapsulationProtocol 标记控制用于以太网连接的协议。选项包括 TCP/IP 和 UDP。 此为读/写标记。
_FloatHandlingType	参数控制	_FloatHandlingType 标记允许更改当前通道级浮点处理。该标记存在于通道级 _System 文件夹中。有关详细信息，请参阅通道属性 - 高级。

标记	类	说明
	标记	此为读/写标记。
_InterDeviceDelayMS	参数控制标记	_InterDeviceDelayMS 标记指定接收到同一通道上当前设备发出的数据后，通道向下一个设备发送请求的延迟时间量。有效范围为 0 至 60000 毫秒。默认设置为 0。 **注意**：此标记仅在使用利用“设备间延迟”的协议的通道上可用。 此标记为读/写标记。
_NetworkAdapter	参数控制标记	_NetworkAdapter 标记允许驱动程序随意更改当前正在使用的 NIC 适配器。作为字符串标记，新近所需的 NIC 适配器名称须以字符串格式写入此标记。所写字符串必须与所需 NIC 的具体说明相符，才能使更改生效。NIC 名称可以从上面列出的 _AvailableNetworkAdapters 标记中获得。对于串行驱动程序，仅在选择“以太网封装”时使用此标记。 **注意**：更改 NIC 选择时，系统会强制驱动程序断开所有当前设备连接，并重新连接。 此为读/写标记。
_UnsolicitedEncapsulationPort	参数控制标记	_UnsolicitedEncapsulationPort 标记控制处于打开状态而允许连接的以太网端口。有效范围为 0 到 65535。 此为读/写标记。
_UnsolicitedEncapsulationProtocol	参数控制标记	_UnsolicitedEncapsulationProtocol 标记控制连接到“主动提供的封装端口”所用的以太网协议。选项包括 TCP/IP 和 UDP。 此为读/写标记。
_WriteOptimizationDutyCycle	参数控制标记	_WriteOptimizationDutyCycle 标记允许随意更改写入/读取率的占空比。占空比控制驱动程序将要对其所执行的每次读取进行的写入次数。_WriteOptimizationDutyCycle 被定义为无符号长整型值。有效范围为每次读取 1 至 10 次写入。有关详细信息，请参阅通道属性 - 写入优化。 此为读/写标记。

串行和以太网驱动程序的设备级系统标记

语法示例：*<通道名称>.<设备名称>._System._Error*

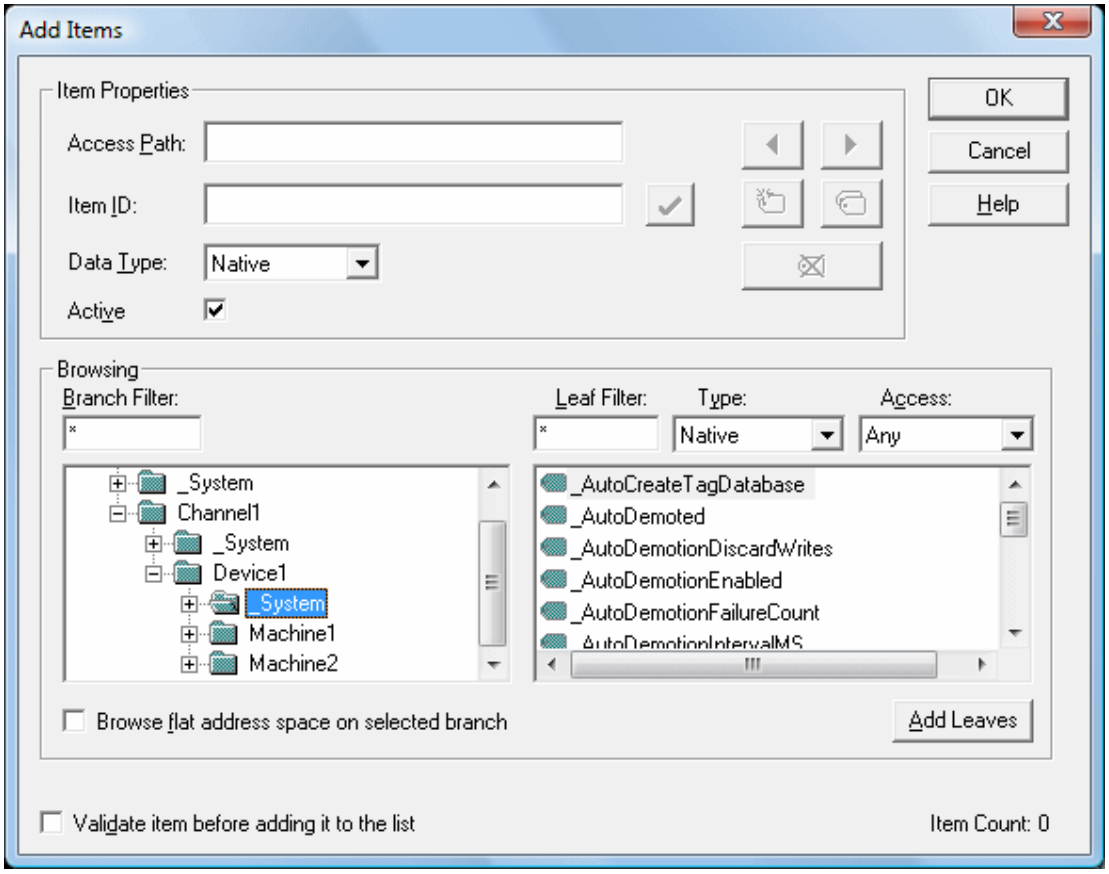
标记	类	说明
_AutoCreateTagDatabase	参数控制标记	_AutoCreateTagDatabase 标记为布尔型标记，用于为附加此标记的设备启动驱动程序的自动 OPC 标记数据库功能。当此标记设置为 True 时，通信驱动程序将尝试自动为此设备生成 OPC 标记数据库。对于不支持“自动 OPC 标记数据库生成”的驱动程序，此标记不会显示。 此为读/写标记。
_AutoDemoted	状况标记	_AutoDemoted 标记为布尔型标记，用于返回设备当前的自动降级状态。设置为 False 时，该设备不会降级且将由驱动程序扫描。设置为 True 时，该设备处于降级状态并且不会由驱动程序扫描。 此为只读标记。

标记	类	说明
_AutoDemotionDiscardWrites	参数控制标记	_AutoDemotionDiscardWrites 标记为布尔型标记，用于指定是否应在降级期间丢弃写入请求。当此标记设置为 False 时，无论 _AutoDemoted 状态为何，都会执行所有写入请求。当此标记设置为 True 时，会在降级期间丢弃所有写入。 此为读/写标记。
_AutoDemotionEnabled	参数控制标记	_AutoDemotionEnabled 标记为布尔型标记，允许设备在无响应时，于特定时间段内自动降级。当此标记设置为 False 时，设备将永不降级。当此标记设置为 True 时，如果达到 _AutoDemotedFailureCount，则会对设备进行降级。 此为读/写标记。
_AutoDemotedFailureCount	参数控制标记	_AutoDemotedFailureCount 标记指定降级设备所需的连续失败次数。_AutoDemotedFailureCount 被定义为长整型数据值。有效范围为 1 到 30。仅当 _AutoDemotionEnabled 设置为 True 时，才能写入此标记。 此为读/写标记。
_AutoDemotionIntervalMS	参数控制标记	_AutoDemotionIntervalMS 标记指定在重新尝试与设备进行通信前，设备处于降级状态的时间，以毫秒为单位。_AutoDemotionIntervalMS 被定义为长整型数据值。有效范围为 100 至 3600000 毫秒。仅当 _AutoDemotionEnabled 设置为 True 时，才能写入此标记。 此为读/写标记。
_ConnectTimeout	参数控制标记	_ConnectTimeout 标记允许随意更改与对设备进行 IP 连接关联的超时。当正在使用本地以太网驱动程序或串行驱动程序处于“以太网封装”模式时，此标记才可用。_ConnectTimeout 被定义为长整型数据类型。有效范围为 1 到 30 秒。 此为读/写标记。
_DemandPoll	状况/控制标记	_DemandPoll 标记发出对与该设备关联的所有活动客户端项目进行设备读取的指令。这等同于通过客户端对那些项目执行异步设备读取。其优先级高于任何应针对正处于活动扫描状态的项目发生的计划读取。 写入时，_DemandPoll 标记会变为 True (1)。当最终活动标记信号表示已完成读取请求时，会返回 False (0)。在标记值返回 False 前，后续写入 _DemandPoll 标记都将失败。需求轮询会遵循通道的读/写占空比。 此为读/写标记。
_Description	状况标记	_Description 标记指示所参考设备的当前用户定义的文本描述。 此为只读标记。
_DeviceId	参数控制标记	_DeviceId 标记允许随意更改设备的 ID。_DeviceId 的数据格式取决于设备的类型。对于大多数串行设备，此标记为长整型数据类型。对于以太网驱动程序，_DeviceId 采用字符串标记格式，以允许输入 IP 地址。在任一情况下，将新设备 ID 写入此标记将导致驱动程序更改目标字段设备。仅当写入此标记的设备 ID 格式正确且在给定驱动程序的有效范围内时，此情况才会发生。 此为读/写标记。
_Enabled	参数控制标	_Enabled 标记为布尔型标记，允许打开或关闭设备的活动状态。当此标记设置为 False 时，此设备中的所有其他用户定义标记和数据会被标记为无效，并且不会接受对设备的写入。当此标记设置为 True 时，会与该设备进行正常通信。

标记	类	说明
	记	此为读/写标记。
_EncapsulationIp	参数控制标记	_EncapsulationIp 标记允许随意指定和更改远程终端服务器的 IP。此标记仅在支持设备属性 - 以太网封装模式的串行驱动程序上可用。_EncapsulationIp 被定义为字符串数据类型，允许输入 IP 地址编号。服务器将拒绝无效的 IP 地址输入。此标记仅对“以太网封装”模式下的串行驱动程序有效。 此为读/写标记。
_EncapsulationPort	参数控制标记	_EncapsulationPort 标记允许指定和更改将远程终端服务器的端口号。_EncapsulationPort 被定义为长整型数据类型。有效范围为 0 到 65535。为使“以太网封装”正确进行，在此标记中输入的端口号必须与所需远程终端服务器的端口号相匹配。此标记仅对“以太网封装”模式下的串行驱动程序有效。 此为读/写标记。
_EncapsulationProtocol	参数控制标记	_EncapsulationProtocol 标记允许指定和更改用于“以太网封装”的 IP 协议。_EncapsulationProtocol 被定义为字符串数据类型。将 "TCP/IP" 或 "UDP" 写入标记用于指定 IP 协议。为使“以太网封装”正确进行，所用协议必须与远程终端服务器的协议相匹配。此标记仅对“以太网封装”模式下的串行驱动程序有效。 此为读/写标记。
_Error	状况标记	_Error 标记为布尔型标记，用于返回设备当前的错误状态。设置为 False 时，设备将正常运行。设置为 True 时，驱动程序会在与此设备进行通信时检测到错误。如果设备已完成请求超时而重试无响应的循环，则其将进入错误状态。 **注意**：有关详细信息，请参阅设备属性 - 定时。 此为只读标记。
_FailedConnection	状况标记	_FailedConnection 标记指定连接失败。此操作仅可用于特定驱动程序。 此为只读标记。
_InterRequestDelay	参数控制标记	_InterRequestDelay 标记允许随意更改设备事务之间的时间间隔。_InterRequestDelay 被定义为长整型数据类型。有效范围为 0 到 30000 毫秒。此标记仅适用于支持此功能的驱动程序。 此为读/写标记。
_RequestAttempts	参数控制标记	_RequestAttempts 标记允许更改通信尝试次数。_RequestAttempts 被定义为长整型值。有效的尝试次数范围为 1 到 10。此标记同样适用于所有驱动程序。 此为读/写标记。
_RequestTimeout	参数控制标记	_RequestTimeout 标记允许随意更改与数据请求关联的超时。_RequestTimeout 被定义为长整型值。有效范围为 100 至 30000 毫秒。此标记同样适用于所有驱动程序。 此为读/写标记。
_NoError	状况标记	_NoError 标记为布尔型标记，用于返回设备当前的错误状态。设置为 Ture 时，设备将正常运行。设置为 False 时，驱动程序会在与此设备进行通信时检测到错误。如果设备已完成请求超时而重试无响应的循环，则其将进入错误状态。

标记	类	说明
		注解：有关详细信息，请参阅设备属性 - 定时。 此为只读标记。
_ScanMode	状况标记	_ScanMode 标记允许客户端指出用于更新的方法。该标记被定义为字符串值，并对应于用户指定的“扫描模式”设置 (位于设备属性)。“遵循客户端指定的扫描速率”具有值 "UseClientRate"，“不超过 *x* 请求数据”具有值 "UseFloorRate"，而“以 *x* 请求所有数据”具有值 "ForceAllToFloorRate"。默认设置为“遵循客户端指定的扫描速率”。 此为只读标记。
_ScanRateMs	状况标记	_ScanRateMs 标记对应于 _ScanMode 标记，并在“扫描模式”设置为“不超过扫描速率请求数据”或“以扫描速率请求所有数据”时使用。此标记被定义为双字型标记。默认设置为 1000 毫秒。 此为只读标记。
_SecondsInError	状况标记	_SecondsInError 标记为双字型标记，用于显示设备处于错误状态的秒数。如果设备未处于错误状态，此标记则会显示 0。 此为只读标记。
_Simulated	参数控制标记	_Simulated 标记为布尔型标记，用于提供有关当前设备的模拟状态的反馈。读取为 True 时，此设备处于模拟模式。在模拟模式下，服务器会返回此设备的正常数据，但不会尝试与实际物理设备进行通信。当标记读取为 False 时，将激活与物理设备的通信。更改标记值允许客户端启用/禁用模拟模式。 这是读/写标记。

使用 OPC 客户端时，可在给定设备的服务器浏览空间的 _System 分支下找到“系统”标记。下图取自所提供的 OPC Quick Client，其中显示“系统”标记如何在 OPC 客户端中呈现。



在 DeviceName 分支下找到的 _System 分支始终可用。如果利用上述示例给出的 DDE 应用程序和 DDE 默认值引用系统标记，链接将显示为："<DDE 服务名称>|_ddedata!Channel1.Device1._System._Error"。

_Enabled 标记提供了一种非常灵活的控制 OPC 应用程序的方法。在某些情况下，特别是在调制解调器应用程序中，可以快捷禁用除当前连接至调制解调器以外的所有设备。此外，使用_Enable 标记将允许应用程序在物理设备正在工作时关闭特定设备，从而消除服务器"事件日志"中无害且无用的通信错误。

另请参阅：
属性标记
调制解调器标记
统计信息标记
存储并转发标记

属性标记

属性标记用于为客户端应用程序提供标记属性的只读权限。将属性名称附加到服务器标记数据库中所定义的完全限定标记地址中，以访问标记属性。有关详细信息，请参阅标记属性 - 常规。

如果完全限定的标记地址为 "Channel1.Device1.Tag1"，则可通过附加说明属性 "Channel1.Device1.Tag1_Description"。

支持的属性标记名称

标记名称	说明
_Name	_Name 属性标记指示所参考标记的当前名称。
_Address	_Address 属性标记指示所参考标记的当前地址。
_Description	_Description 属性标记指示所参考标记的当前说明。
_RawDataType	_RawDataType 属性标记指示所参考标记的原始数据类型。
_ScalingType	_ScalingType 属性标记指示所参考标记的换算类型 (无、线性或平方根)。
_ScalingRawLow	_ScalingRawLow 属性标记指示所参考标记的原始低范围。在应用换算的情况下，如果将换算设置为无，则此值包含默认值。

标记名称	说明
_ScalingRawHigh	_ScalingRawHigh 属性标记指示所参考标记的原始高范围。在应用换算的情况下，如果将换算设置为无，则此值包含默认值。
_ScalingScaledDataType	_ScalingScaledDataType 属性标记指示所参考标记的换算数据类型。在应用换算的情况下，如果将换算设置为无，则此值包含默认值。
_ScalingScaledLow	_ScalingScaledLow 属性标记指示所参考标记的换算低范围。在应用换算的情况下，如果将换算设置为无，则此值包含默认值。
_ScalingScaledHigh	_ScalingScaledHigh 属性标记指示所参考标记的换算高范围。在应用换算的情况下，如果将换算设置为无，则此值包含默认值。
_ScalingClampLow	_ScalingClampLow 属性标记指示是否应限制所参考标记的换算低值。在应用换算的情况下，如果将换算设置为无，则此值包含默认值。
_ScalingClampHigh	_ScalingClampHigh 属性标记指示是否应限制所参考标记的换算高值。在应用换算的情况下，如果将换算设置为无，则此值包含默认值。
_ScalingUnits	_ScalingUnits 属性标记指示所参考标记的换算单位。在应用换算的情况下，如果将换算设置为无，则此值包含默认值。

另请参阅：
统计信息标记
调制解调器标记
系统标记

统计信息标记

统计信息标记用于为客户端应用程序提供有关服务器信道通信操作的反馈。统计信息标记仅在启用诊断时可用。*有关详细信息，请参阅信道诊断和 OPC 诊断查看器。*

语法示例：*<信道名称>._Statistics。_FailedReads*

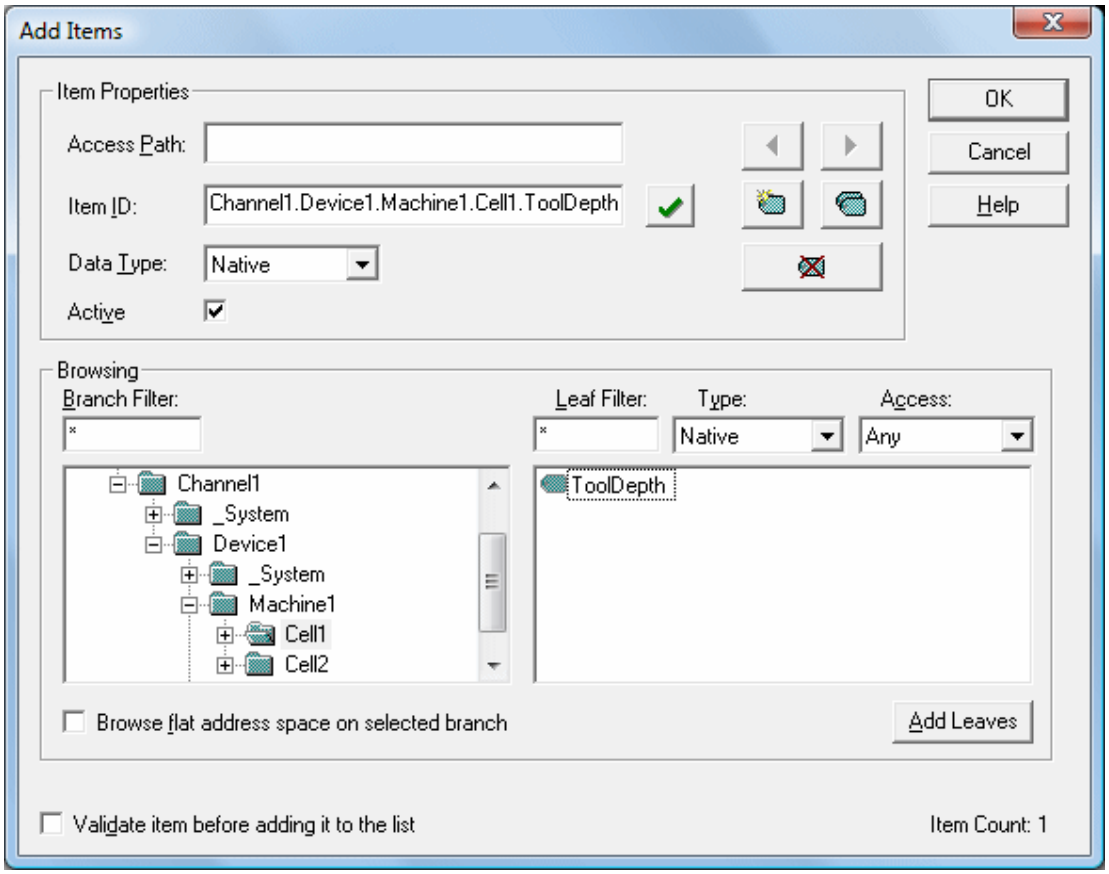
支持的统计信息标记名称

标记名称	说明
_SuccessfulReads	_SuccessfulReads 标记包含自应用程序启动或自上次调用 _Reset 标记以来，该信道成功完成的读取次数计数。此标记的格式为无符号 32 位整型，并且最终可滚动更新。此标记为只读形式。
_SuccessfulWrites	_SuccessfulWrites 标记包含自应用程序启动或自上次调用 _Reset 标记以来，该信道成功完成的写入次数计数。此标记的格式为无符号 32 位整型，并且最终可滚动更新。此标记为只读形式。
_FailedReads	_FailedReads 标记包含自应用程序启动或自上次调用 _Reset 标记以来，此信道读取完成失败的次数计数。只有当信道请求根据为设备配置的超时和重试次数失败后，才增加此计数。此标记的格式为无符号 32 位整型，并且最终可滚动更新。此标记为只读形式。
_FailedWrites	_FailedWrites 标记包含自应用程序启动或自上次调用 _Reset 标记以来，该信道写入完成失败的次数计数。此计数仅在信道基于设备配置的超时和重试次数而请求失败后增加。此标记的格式为无符号 32 位整型，并且最终可滚动更新。此标记为只读形式。
_RxBytes *	_RxBytes 标记包含自应用程序启动或自上次调用 _Reset 标记以来，信道从连接设备接收的字节数量计数。此标记的格式为无符号 32 位整型，并且最终可滚动更新。此标记为只读形式。
_TxBytes	_TxBytes 标记包含自应用程序启动或自上次调用 _Reset 标记以来，信道发送至连接设备的字节数量计数。此标记的格式为无符号 32 位整型，并且最终可滚动更新。此标记为只读形式。
_Reset	_Reset 标记可用于重置所有诊断计数器。_Reset 标记的格式设置为布尔型标记。在 _Reset 标记中写入非零值将导致诊断计数器重置。此标记为读/写形式。
_MaxPendingReads	_MaxPendingReads 标记包含自应用程序启动 (或 _Reset 标记调用) 以来，信道待处理的读取请求的最大数量计数。此标记的格式为无符号 32 位整型。此标记为只读形式。

标记名称	说明
_MaxPendingWrites	_MaxPendingWrites 标记包含自应用程序启动 (或 _Reset 标记调用) 以来，信道待处理的写入请求的最大数量计数。此标记的格式为无符号 32 位整型。此标记为只读形式。
_NextReadPriority	_NextReadPriority 是下一个通道的待处理读取队列中读取的优先级别会反映出一通道级的系统标记。可能的值为-1：读取挂起。0：下一次读取是排程级别的需求轮询的结果或显式读取从客户端。1- *n*：下一次读取是已安排读取的结果。此标记为只读形式。
_PendingReads	_PendingReads 标记包含信道当前待处理的读取请求计数。此标记的格式为无符号 32 位整型。此标记为只读形式。
_PendingWrites	_PendingWrites 标记包含信道当前待处理的写入请求计数。此标记的格式为无符号 32 位整型。此标记为只读形式。

* 此统计信息项不会在模拟模式下更新 (*请参阅设备属性*)。

统计信息标记仅在启用诊断时可用。要从 OPC 客户端进行访问，可从给定信道服务器浏览空间的 _Statistics 分支浏览诊断标记。下图取自 OPC Quick Client，其中显示“诊断”标记如何在 OPC 客户端中呈现。



仅当信道启用诊断功能时，才会显示 _Statistics 分支 (位于信道分支下方)。要从 DDE 应用程序参考诊断标记，对于以上示例和 DDE 默认值，链接将显示为："<DDE 服务名称>|_ddedata!Channel1._统计信息。_SuccessfulReads"。

“诊断”标记的值也可通过使用“通信诊断查看器”在服务器中查看。如果在“信道属性”下启用“诊断数据捕获”，请右键单击该信道，然后选择**“诊断”**。

另请参阅：

系统标记

属性标记

调制解调器标记

选择使用调制解调器时，会自动为信道创建以下标记。

语法示例：*<信道名称>.<设备名称>._调制解调器._Dial*

支持的调制解调器标记名称

标记名称	说明	访问
_Dial	向此标记写入任何值，都会开始拨打当前电话号码。除非当前状态为 3 (空闲)，否则会忽略写入。如果没有初始化当前电话号码，则会报告错误。在 Mode 标记设置为 2 (仅允许来电) 时尝试发出拨号命令会产生错误。	读/写
_DialNumber	DialNumber 标记会在应用任何拨号首选项转换 (如添加区域代码) 后显示实际拨打的电话号码。此标记旨在用于调试。在手动输入电话号码时，它可以向操作员提供有用的反馈。	只读
_Hangup	向此标记写入任何值，都会挂起当前连接。当外部设备调用服务器后，Hangup 标记将结束当前连接。如果状态 <= 3 (空闲)，则忽略对 Hangup 标记的写入，这意味着当前没有打开的连接。	读/写
_LastEvent	无论状态何时更改，更改的原因都将在此标记中设置为数字。*有关事件编号和含义的列表，请参阅最后一个事件值*。	只读
_Mode	此标记可将线路配置为仅呼叫、仅应答或呼叫并应答。 将 1 写入 Mode 标记会将线路设置为仅呼出，即在此模式下不应答任何来电。将 2 写入 Mode 标记会将线路设置为仅呼入，请求拨出 (写入 Dial 标记) 将被忽略。默认设置为 0，允许呼出和呼入。 此值只能在状态 <= 3 (空闲) 时更改。	读/写
_PhoneNumber	这是当前要拨打的电话号码。用户可以随时写入此值，但更改仅在状态 <= 3 (空闲) 时有效。如果用户在状态大于 3 时写入电话号码，则号码将加入队列。一旦状态下降到 3 或更低，新的号码将被转移到标记。队列大小为 1，因此只保留最后一个电话号码。 电话号码必须采用规范格式才能应用拨号首选项。如果采用规范格式，则可将要拨打的号码 (应用拨打首选项后) 显示为 DialNumber。 规范格式如下： +<国家代码>[空格](<区域代码>)[空格]<电话号码> 示例：+1 (207) 846-5881 **注意**：美国的国家代码为 1。 如果号码不是规范形式，则不应用拨号首选项。拨打的号码与输入的号码完全相同。用户还可以输入 Phonebook 标记名称而非电话号码。在这种情况下，使用 Phonebook 标记的当前值。	读/写
_Status	这是分配给信道的调制解调器的当前状态。有关状态值和含义的列表，请参阅 状态值。	只读
_StringLastEvent	此标记包含 LastEvent 标记值的文本表示。有关事件编号和含义的列表，请参阅 LastEvent 字符串值。	只读
_StringStatus	此标记包含 Status 标记值的文本表示。有关事件编号和含义的列表，请参阅 状态字符串值。	只读

状态值

目前正在使用 32 位状态变量的五个最低位。

位	含义
0	TAPI 已初始化
1	线路断开
2	已连接
3	呼叫
4	应答

读取为整数时，Status 标记的值始终为下列项之一：

值	含义
0	未初始化，信道不可用
1	已初始化，没有线路断开
3	线路断开，状态为空闲
7	已连接
11	呼叫
19	应答

状态字符串值

状态值	StringStatus 文本
0	未初始化，信道不可用
1	已初始化，没有线路断开
3	空闲
7	已连接
11	呼叫
19	应答

最后一个事件值

LastEvent	更改原因
-1	<空白> [尚未发生任何事件]
0	TAPI 已初始化
1	线路闭合
2	线路断开
3	线路已连接
4	用户已将线路永久删除
5	已在远程站点将线路永久删除
6	无应答
7	线路忙碌
8	无拨号音
9	检测到来电
10	用户拨号
11	无效电话号码
12	线路硬件错误导致线路闭合

最后一个事件字符串值

LastEvent	StringLastEvent
-1	<空白> [尚未发生任何事件]
0	TAPI 已初始化

LastEvent	StringLastEvent
1	线路闭合
2	线路断开
3	线路已连接
4	用户已将线路永久删除
5	已在远程站点将线路永久删除
6	无应答
7	线路忙碌
8	无拨号音
9	检测到来电
10	用户拨号
11	无效电话号码
12	线路硬件错误导致线路闭合
13	无法拨号

通信序列化标记

语法示例：*<通道名称>._CommunicationSerialization._VirtualNetwork*

标记	说明
_NetworkOwner 类：状况标记	_NetworkOwner 标记指示通道当前是否拥有对网络通信的控制。更改频率反映了可拥有通道控制权的时间长度。 此标记为只读形式。
_Registered 类：状况标记	_Registered 标记指示通道当前是否已注册到虚拟网络。设置 _VirtualNetwork 后，在操作允许的情况下，通道将从其当前已注册到的网络上取消注册 (如 _RegisteredTo 中所指示)。也就是说，如果通道在切换过程中拥有控制权，则它在未释放控制权之前无法取消注册。取消注册后，通道将注册到新的虚拟网络。如果 _VirtualNetwork 为“无”，则此标记为 FALSE。 此标记为只读形式。
_RegisteredTo 类：状况标记	_RegisteredTo 标记指示通道当前注册到的虚拟网络。设置 _VirtualNetwork 后，在操作允许的情况下，通道将从其当前已注册到的网络上取消注册。也就是说，如果通道在切换过程中拥有控制权，则它在未释放控制权之前无法取消注册。取消注册后，通道将注册到新的虚拟网络。此标记指示切换网络时是否存在延迟，即 _VirtualNetwork 和 _RegisteredTo 在一段时间内可能有所不同。如果 _VirtualNetwork 为“无”，则此标记“不适用”。 此标记为只读形式。
_StatisticAvgNetworkOwnershipTimeSec 类：状况标记	_StatisticAvgNetworkOwnershipTimeSec 标记指示从启动应用程序开始 (或从最后一次写入 _StatisticsReset 起) 通道可保持拥有控制权的平均时间。此标记可帮助确定忙碌通道/瓶颈。此标记的格式为 32 位浮点，并且最终可滚动更新。 此标记为只读形式。
_StatisticNetworkOwnershipCount 类：状况标记	_StatisticNetworkOwnershipCount 标记指示从启动应用程序开始 (或从最后一次写入 _StatisticsReset 起) 通道被授予通信控制权的次数。此标记的格式为无符号 32 位整型，并且最终可滚动更新。 此标记为只读形式。
_StatisticNetworkOwnershipTimeSec	_StatisticNetworkOwnershipTimeSec 标记指示从启动应用程序

标记	说明
类：状况标记	开始 (或从最后一次写入 _StatisticsReset 起) 通道可保持所有权的时长 (以秒为单位)。此标记的格式为 32 位浮点，并且最终可滚动更新。 此标记为只读形式。
_StatisticsReset	_StatisticsReset 标记可用于重置所有统计信息计数器。_StatisticsReset 标记的格式为布尔型标记。在 _StatisticsReset 标记中写入非零值，将导致统计计数器重置。 此标记为读/写形式。
_TransactionsPerCycle	_TransactionsPerCycle 标记指示在轮流使用虚拟网络中的其他通道时，发生在该通道中的读取/写入事务处理的次数。它允许从客户端应用程序中更改通道级别设置。此标记的格式为有符号 32 位整型 (长整型)。有效范围为 1 到 99。默认设置为 1。 此标记为读/写形式。
_VirtualNetwork 类：参数标记	_VirtualNetwork 标记允许通道的虚拟网络选择进行即时更改。作为字符串标记，所需虚拟网络必须作为字符串值写入标记，可使用下列可能的选择：无、网络 1、网络 2、...、网络 500。要禁用通信序列化，请选择“无”。 此标记为读/写形式。

通信管理

自动降级

自动降级属性允许驱动程序在设备未响应的情况下使设备暂时处于关闭扫描状态。通过将无响应设备置于离线状态，驱动程序可在特定时间段内停止与无响应设备的通信，从而继续优化其与同一通道上其他设备的通信。特定时间段结束后，驱动程序将重新尝试与无响应设备进行通信。如果设备响应，则该设备会进入开启扫描状态；否则，设备将再次开始其关闭扫描时间段。

有关详细信息，请参阅设备属性 - 自动降级。

网络接口选择

可选择 NIC 卡，用于在以太网封装模式下运行的任何以太网驱动程序或串行驱动程序。网络接口功能可用于通过 NIC 名称或当前为其分配的 IP 地址来选择特定的 NIC 卡。可用 NIC 的列表中包括唯一 NIC 卡和分配有多个 IP 地址的 NIC。此选择将显示可能处于活动状态 (例如拨号连接) 的 WAN 连接。

以太网封装

“以太网封装”模式旨在为那些通过以太网与终端服务器相连的串行设备提供通信。终端服务器本质上是虚拟串行端口：终端服务器可将以太网上的 TCP/IP 消息转换为串行数据。消息转换为串行形式后，用户可将支持串行通信的标准设备连接到终端服务器。用户可使用终端服务器设备在所有工厂操作中放置 RS-232 和 RS-485 设备，同时，仍然允许单个本地化 PC 访问远程挂载设备。此外，“以太网封装”模式还可以根据需要将各个网络 IP 地址分配到设备。通过使用多个终端服务器，用户可以通过以太网从单个 PC 访问数百个串行设备。

有关详细信息，请参阅如何... 和设备属性 - 以太网封装。

调制解调器支持

此服务器支持使用调制解调器连接到远程设备，其中远程设备通过使用在创建拨号网络连接后获得的信道级特殊调制解调器标签建立。这些信道级调制解调器标记可用于拨号远程设备、监视连接时的调制解调器状况，并在完成时终止调用。

注意：不是所有的串行驱动程序都支持使用调制解调器。*要确定调制解调器支持情况，请参阅特定驱动程序的帮助文档。*

在访问调制解调器系统标记时，信道名称可以用作基群或主题名称。调制解调器必须通过“控制面板”设置进行配置后，才可用于操作系统。

调制解调器正确安装后，可以通过在信道属性中选择**“调制解调器”**作为“物理媒体”来启用它。

有关特定设置信息，请参阅 Windows 和调制解调器文档。

重要事项：很多新型商用调制解调器皆可用于拨号网络服务器连接并可进行最快速最清晰的信号协商。与串行自动化设备通信时，调制解调器需要以特定波特率 (位/秒) 和奇偶性进行连接。因此，强烈建议您使用外部调制解调器 (可配置为使用特定波特率和奇偶性设置进行拨号)。要确定适用于特定应用程序的最佳调制解调器，请参阅“技术支持”。有关如何在项目中使用调制解调器的示例，请参阅在服务器项目中使用调制解调器。

在服务器项目中使用调制解调器

调制解调器可将串行数据从 RS-232 端口转换成可通过电话线传输的信号电平。为此，它们会将串行数据的每个字节分解成用于生成传送信号的位。大多数调制解调器可为发送的每个字节的数据最多转换 10 位信息。设备必须能够使用 10 位或以下才能通过调制解调器进行通信。要确定特定设备正在使用的位数，请参考下面的公式。

起始位 + 数据位 + 奇偶校验 + 停止位 = 总位数

例如，Modbus RTU 驱动程序配置为使用 8 个数据位、偶校验、1 个停止位和 1 个起始位。代入公式后，将为 1 + 8 + 1 + 1，等于 11 位。普通调制解调器无法向此 Modbus 设备传送数据。如果“奇偶校验”更改为“无”，它将为 1 + 8 + 0 + 1，等于 10 位。普通调制解调器可以向此 Modbus 设备传送数据。

某些驱动程序无法配置为使用 10 位或以下的数据格式，因此无法使用标准调制解调器。但是，它们需要可以处理 11 位数据发送的调制解调器。对于属于此类别的驱动程序，请咨询设备制造商以获得关于相应调制解调器供应商的建议。调制解调器运行适用于所有串行驱动程序，无论驱动程序是否支持调制解调器运行。

配置启动调制解调器

该服务器使用 Windows TAPI 接口访问连接到 PC 的调制解调器。TAPI 接口为 Windows 程序提供了一个通用接口，可以通过 PC 中存在的一系列调制解调器来访问。必须先安装调制解调器制造商为 Windows OS 提

供的一组驱动程序，服务器才能在项目中使用调制解调器。可以使用 Windows 控制面板来安装新的调制解调器。

有关调制解调器安装和设置的信息，请参阅 Windows 和调制解调器的文档。

正确安装调制解调器后，用户便可开始在服务器项目中使用它。接收端或设备调制解调器必须正确配置才能使用。用户必须确认接收调制解调器与驱动程序提供的配置文件是否匹配。

电缆

在使用项目之前，必须在接收调制解调器与设备之间配置电缆连接。需要三条电缆：用于直接连接的现有设备通信电缆、空调制解调器适配器和空调制解调器电缆。空调制解调器电缆连接到调制解调器，而所有引脚都连接到电缆两端的相同引脚。设备通信电缆用于连接目标设备，通常会将引脚 2 和 3 反接。由于当前用于直接连接的设备的通信电缆正在工作，因此可以通过连接一个空调制解调器适配器在接收调制解调器上使用。类似地，PC 调制解调器电缆从 PC 连接到启动调制解调器。电缆连接就位后，便可在应用程序中使用调制解调器。

注意：在大多数计算机商店都可以找到空调制解调器适配器。

示例：服务器端调制解调器配置

调制解调器完成配置和安装后，可以与服务器一起使用。

1. 要启动，请加载直接连接项目并双击信道名称。在**“信道属性”**中，打开“串行通信”组。
2. 在**“物理媒介”**下拉菜单中，选择**“调制解调器”**。
3. 在**“调制解调器设置”**中，选择计算机上可用的调制解调器。

   注意：如果计算机上没有可用的调制解调器，则用户无法从“物理媒介”下拉菜单中选择调制解调器。如果发生这种情况，请退出服务器并尝试使用操作系统提供的“调制解调器配置”工具重新安装调制解调器。

4. 要配置启动调制解调器的特性，请使用**“调制解调器设置”**中的属性。*有关详细信息，请参阅信道属性 - 串行通信。*
5. 完成后，单击**“应用”**。然后，单击**“确定”**以保存并退出“信道属性”。

在应用程序中使用调制解调器

一旦启用调制解调器运行，便会向数据客户端提供一组预定义的标记。这些调制解调器标记可控制和监视连接的调制解调器，并包含在信道名称 (已成为用于访问调制解调器标记的活动 OPC 访问路径) 下。由于服务器几乎不知道调制解调器控制需要什么应用程序，因此它并不会暗示任何类型的控制。通过使用预定义的调制解调器标记，用户可以应用应用程序的脚本功能来控制服务器如何使用所选调制解调器。

电话簿

电话簿是电话簿标记 (电话号码) 的集合，可用于代替指定写入调制解调器系统标记的 "_PhoneNumber" 标记中的电话号码。对于已将**“物理媒介”**设置为“调制解调器”的所有信道，都会自动创建电话簿。与电话簿标记关联的数据是将由服务器进行拨号的电话号码。客户端写入电话簿标记的操作会使服务器使用与该标记相关联的电话号码进行拨号。

数据类型	权限
字符串	读/写

电话簿标记是通过在电话簿中创建新条目来创建的。要添加新的电话簿条目，请单击项目树中的“电话簿”节点，然后单击“新建电话号码”图标。

这将打开“电话号码”属性编辑器。

“名称”：指定电话号码条目的名称。它将成为 "_Phonebook" 系统标记组中 OPC 浏览数据的一部分。其长度不能超过 256 个字符。尽管使用描述性名称通常是一个好方法，但在浏览 OPC 服务器的标记空间时，一些 OPC 客户端应用程序的显示窗口可能不够大。电话号码中的名称必须是唯一的。

号码：指定从 OPC 客户端应用程序调用相关电话簿标记时要拨打的电话号码。可输入最多 64 位的字符串。

说明：输入文字以将注释附加到电话号码条目中。其长度不能超过 255 个字符。

注意：服务器全天在线运行时，可以随时更改这些参数。对属性的更改立即生效；但是，已经连接到此标记的 OPC 客户端在发布并重新获取标记之前不会受到影响。

自动拨号优先级

已为信道启用“自动拨号”时，初始连接请求将首先尝试拨打电话簿中遇到的第一个条目。如果该尝试未成功，则会尝试电话簿中的下一个号码，以此类推。该序列将继续，直到建立调制解调器连接，或客户端将全部参考发布到由信道提供的数据为止。“自动拨号”用于拨号的顺序优先级由用户定义，可通过选择电话簿条目并单击其中一个“更改优先级”图标进行更改，如下所示。也可以通过打开所选条目的上下文菜单来更改它们。

示例

对于创建的“电话簿”条目和设置为 "Site1" 的名称：

语法示例：*<信道名称>._Phonebook.Site1*

自动拨号

当在服务器项目中指定使用调制解调器时，“自动拨号”功能将会自动执行客户端应用程序所需的操作。如果没有“自动拨号”功能，这些操作 (其中包括连接、断开连接和分配电话号码) 就会由外部客户端应用程序通过使用信道级别调制解调器标记执行。例如，要开始建立连接的过程，客户端会将拨号字符串写入 “<*信道名称*>._Modem._PhoneNumber"，并将值写入 "<*信道名称*>._Modem._Dial"。当不再需要远程设备中的数据时，客户端会结束呼叫，方法是写入“<*信道名称*>._Modem._Hangup"。

在尝试建立连接时，“自动拨号”功能会通过自动拨打电话簿中定义的电话号码来减轻客户端的这些职责。当没有任何客户端标记参考依赖于调制解调器连接时，连接会自动断开。要访问“自动拨号”属性，请单击**“信道属性”|“串行通信”**。

有关详细信息，请参阅信道属性 - 串行通信。

调制解调器连接和连接断开

当客户端连接到服务器运行时，并从启用了“自动拨号”功能的信道所连接的设备请求数据时，便开始了建立调制解调器连接过程。首先会尝试拨打在电话簿中遇到的第一个电话号码来发出初始连接请求。如果该尝试未成功，则会尝试电话簿中的下一个号码，以此类推。该序列将继续，直到建立调制解调器连接，或客户端将全部参考发布到由信道提供的数据为止。

注意：重新建立连接时，将使用上次生成成功连接的电话簿条目。如果之前的电话簿条目均未连接成功 (或如果该条目在连接成功后已被删除)，则将使用用户定义的电话号码序列。服务器重新初始化或重新启动期间，不会保留用于重新拨号的号码。

另请参阅：电话簿

定时

定时设置 (例如，在继续拨打下一个电话号码之前等待连接的时长) 由 TAPI 调制解调器配置确定，而不是由任何特定调制解调器自动拨号设置确定。

注意：对于某些驱动程序而言，串行端口一旦打开，就不允许其关闭。使用这些驱动程序建立的连接只有在所有客户端参考均被释放后才会断开连接 (除非将 TAPI 设置配置为在经历某一段空闲时间后断开连接)。

客户端访问

调制解调器标记可用于对调制解调器进行客户端级别的控制。如果启用了“调制解调器自动拨号”功能，但是调制解调器标记的写入权限受到限制，此时只能使用一种访问形式。调制解调器标记的值会按客户端在调制解调器控制范围内时的方式更新。

从配置更改自动拨号设置

运行时会根据以下规则对设置的更改做出反应：

- 如果在客户端已向调制解调器拨号并建立连接后启用“自动拨号”，则会忽略更改，直到调制解调器断开连接为止。如果客户端在断开连接时仍从信道请求数据，则初始连接序列开始。

- 如果在没有任何调制解调器连接且客户端正在从信道请求数据的情况下启用“自动拨号”，则初始连接序列开始。
- 如果在存在现有自动拨号连接的情况下禁用“自动拨号”，则不会执行任何操作，且连接会断开。

另请参阅：信道属性 - 串行通信

设计项目

以下示例使用服务器随附的 Simulator 驱动程序来演示创建、配置和运行项目的过程。Simulator 驱动程序是基于内存的驱动程序，能为演示提供静态数据和变化数据。因为其不支持在其他通信驱动程序中找到的配置选项范围，所以有些示例可能会使用其他驱动程序的图像来展示特定的产品特征。有关特定主题的详细信息，请从下表中选择一个链接。

运行服务器
开始新项目
添加和配置通道
添加和配置设备
添加用户定义标记
生成多个标记
添加标记缩放
保存项目
打开加密的项目
测试项目

有关软件和硬件要求的信息，请参阅系统要求。

运行服务器

此服务器既可作为服务，也可作为桌面应用程序来运行。当以默认设置作为服务运行时，服务器始终处于在线状态。当作为桌面应用程序运行时，OPC 客户端会在连接和收集数据时自动调用服务器。为使任一进程正常工作，用户必须先创建和配置项目。在开始时，服务器会自动加载上一次使用的项目。

最初，用户必须手动调用服务器。为此，请双击桌面图标或从位于系统托盘中的“管理”菜单中选择**“配置”**。界面的外观取决于用户所做的更改。

服务器开始运行后，便可创建项目。

有关服务器元素的详细信息，请参阅基本服务器组件。有关用户界面的详细信息，请参阅导航配置。

开始新建项目

用户必须配置服务器以确定在运行期间提供什么内容。服务器项目包括通道、设备、标记组和标记的定义。这些因素存在于项目文件的上下文中。与许多应用程序一样，可定义、保存和加载一些项目文件。

某些配置选项是全局的，并且应用于所有项目。这些全局选项在**“工具”|“选项”**对话框中配置，其中包括常规选项和运行时连接选项。这些设置存储在名为 "settings.ini" 的 Windows INI 文件中，该文件存储在安装时选择的应用程序数据目录中。虽然全局选项通常存储在 Windows 注册表中，但 INI 文件支持在不同计算机之间复制这些全局设置。

最初，软件是在默认项目打开的情况下打开的。与任何其他文件一样，该文件可编辑、保存和关闭。

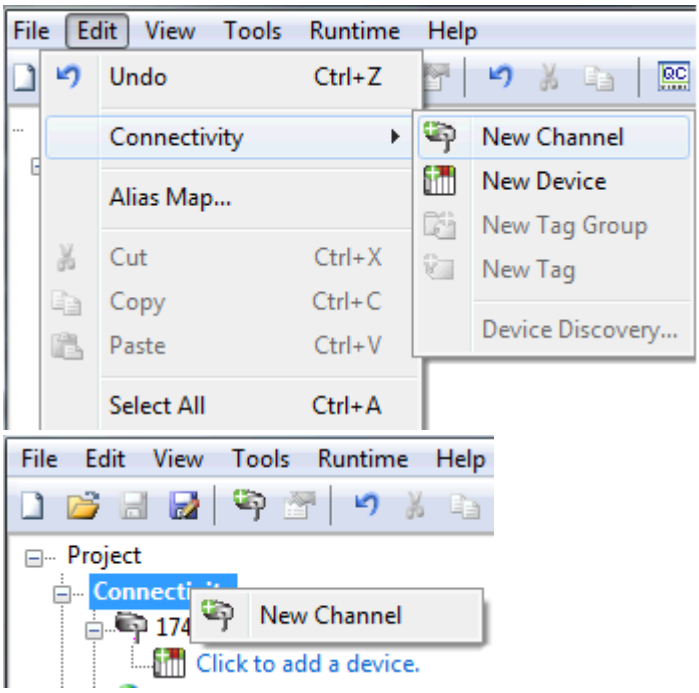
1. 要定义新项目，请选择**“文件”|“新建”**。
2. 如果系统提示您关闭、保存或离线编辑。
3. 选择**“文件”|“另存为”**。
4. 输入密码以保护加密的项目文件。
5. 选择存储文件的位置。
6. 单击**“保存”**。
7. 通过“添加通道”开始配置项目文件。

***另请参阅：**选项 - 常规、保存项目*

添加和配置信道

创建新项目时，用户必须先确定应用程序所需的通信驱动程序：这称为服务器中的信道。可在一个项目内定义若干信道，具体视安装的一个或多个驱动程序而定。有关详细信息，请参阅以下说明。

1. 首先，请通过以下方式将新的信道添加到项目中：
   单击**“编辑”|“连接”|“新建信道”**- 或者 -
   单击工具栏上的**“新建信道”**图标 - 或者 -
   右键单击树中的**“连接”**节点并选择**“新建信道”**



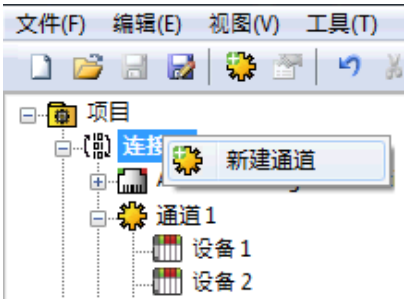
2. 在“信道向导”中，保留其默认设置的信道名称 "Channel1"。然后单击**“下一步”**。
3. 在**“设备驱动程序”**中，选择要应用于信道的通信驱动程序。然后单击**“下一步”**。在本示例中，将使用 Simulator 驱动程序。
4. 对于 Simulator 驱动程序，下一页面是**“信道汇总”**。其他设备可能有其他信道向导页面，可用于配置其他属性 (如通信端口、波特率和奇偶校验)。有关详细信息，请参阅信道属性 - 串行通信。
5. 完成后，单击**“完成”**。

另请参阅：如何优化服务器项目、服务器汇总信息

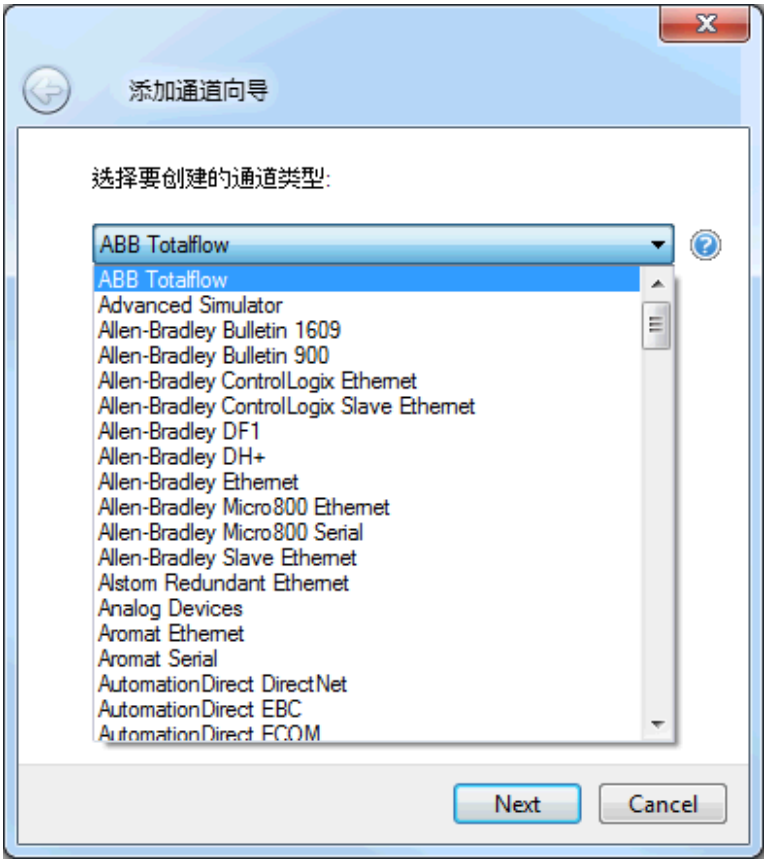
信道创建向导

“信道创建向导”可引导您逐步完成配置信道 (由所使用的协议定义) 的过程。一旦对信道进行定义，分配给该信道的所有设备便可使用其属性和设置。具体属性由协议或所选驱动程序决定。

1. 在树状视图中，右键单击**“连接”**节点，然后选择**“新建信道”**(或选择**“编辑”| “连接”| “新建信道”**)。

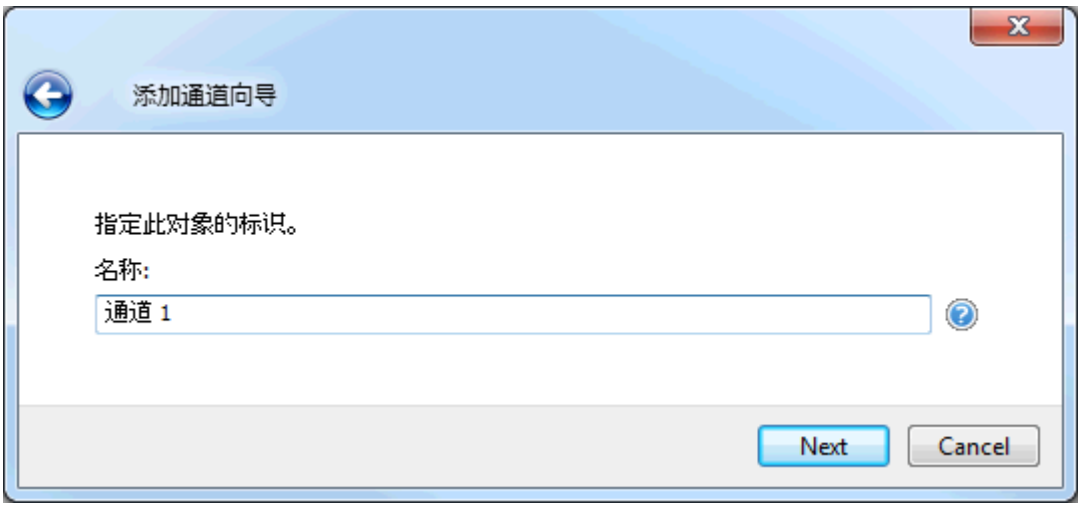


2. 从可用驱动程序下拉列表中选择将要创建的信道类型。



3. 单击**“下一步”**。

4. 输入信道名称以帮助对其进行识别 (用于标记路径、事件日志消息和别名使用)。



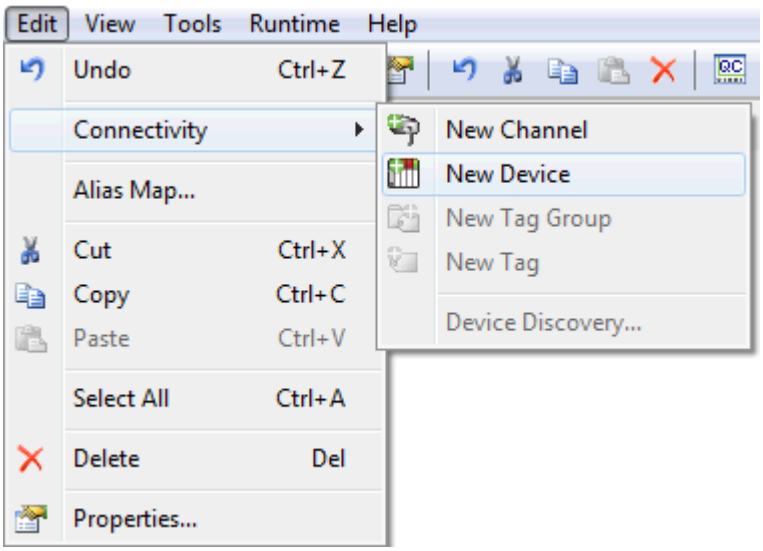
5. 单击**"下一步"**。
6. 根据选项和环境配置信道属性。
7. 查看新的信道摘要并选择**"上一步"**对其进行更改或单击**"结束"**将其关闭。

添加和配置设备

一旦定义了信道，就可以添加设备。设备用于标识通信链路的物理节点或站点，可以通过它将连接的定义与应用程序中的特定关注点联系起来。从这一角度来看，设备是向数据库对象描述连接的正确术语。因此，"设备"是指网络上的特定设备，支持多个设备节点，并允许用户模拟联网设备。

注意：在本例中，将使用 Simulator 驱动程序。设备向导中的选项取决于驱动程序。

1. 首先，请选择将设备添加到的信道。
2. 首先，请通过以下方式将新的设备添加到项目中：
   单击**"编辑"(Edit) |"连接"(Connectivity) | "新建设备"(New Device)** - 或者 -
   单击工具栏上的**"新建设备"(New Device)** 图标 - 或者 -
   右键单击树中的**"连接性"**节点并选择**"新建信道"**。



3. 在设备向导中，将名称保留为默认设置的 “Device1”，然后单击**“下一步”**。
4. 在**“型号”**中，选择模拟中设备的寄存器大小为 8 或 16 位，然后单击**“下一步”**。

   注意：其他设备驱动程序可能要求用户选择设备型号。在本例中，选择的寄存器大小为 16 位。

5. 在 **ID** 中，选择设备 ID (实际通信协议所需的唯一标识符)。然后单击**“下一步”**。

   注意：设备 ID 格式和样式取决于所使用的通信驱动程序。对于 Simulator 驱动程序，设备 ID 是数字值。

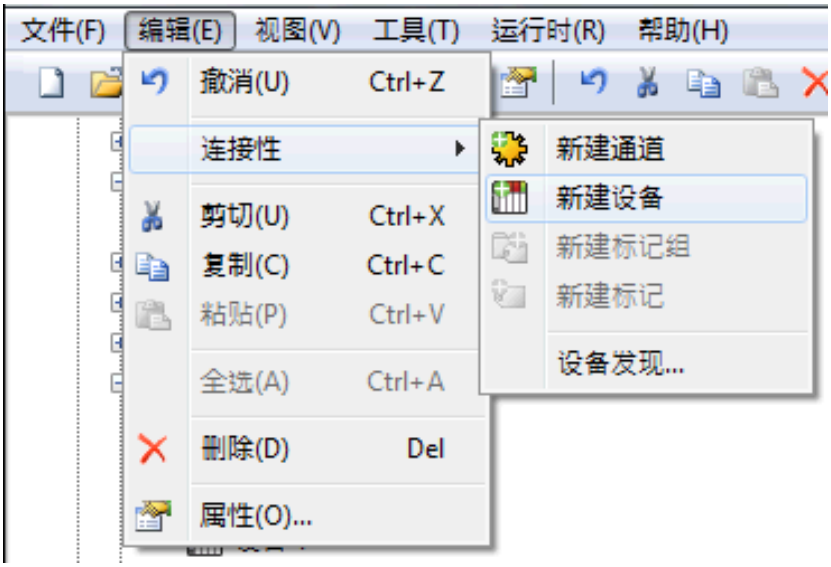
6. 在**“扫描模式”**下，指定设备的扫描速率。然后单击**“下一步”**。
7. 对于 Simulator 驱动程序，下一页面是**“设备汇总”**。其他驱动程序可能还有设备向导页面用于配置其他属性 (例如：计时)。有关详细信息，请参阅设备属性。
8. 完成后，单击**“完成”**。

注意：在联机全时操作模式下，服务器可随时提供 OPC 数据。但此时由于项目尚未保存，配置可能会丢失。在保存之前，用户可以向服务器添加标记。有关详细信息，请参阅添加用户定义的标记。

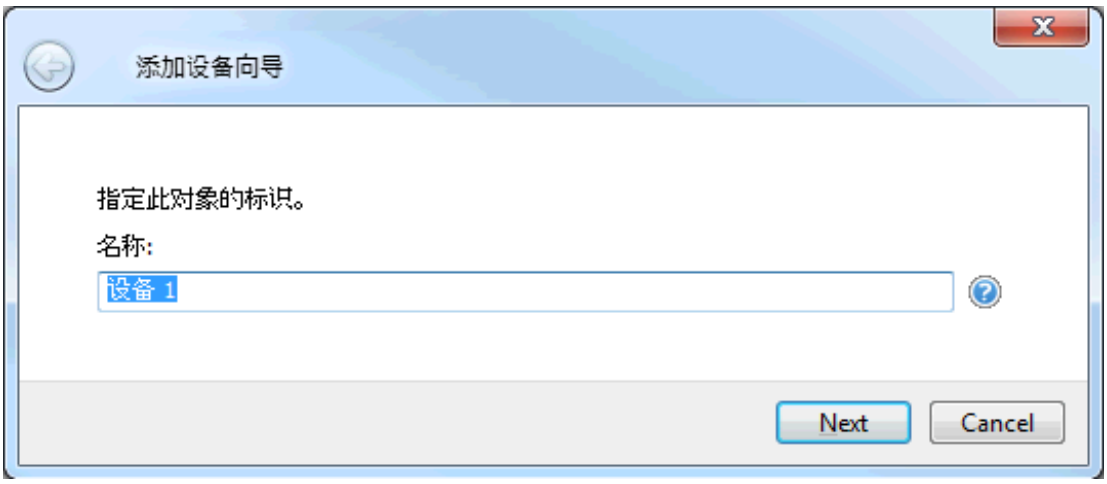
设备创建向导

可通过“设备创建向导”了解如何配置设备以进行通信和数据收集。具体属性由协议或所选驱动程序决定。

1. 从树状视图中，查找并选择用于添加设备的通道。
2. 右键单击并选择**“新建设备”(New Device)** 或选择 **“编辑”(Edit) | “连接”(Connectivity) | “新建设备”(New Device)**)。



3. 输入设备名称以帮助对其进行识别 (用于标记路径、事件日志消息和别名使用)。



4. 单击**"下一步"**。
5. 根据选项和环境配置设备属性。
6. 查看新的设备摘要并选择**"上一步"(Back)** 对其进行更改或单击**"结束"(Finish)** 将其关闭。

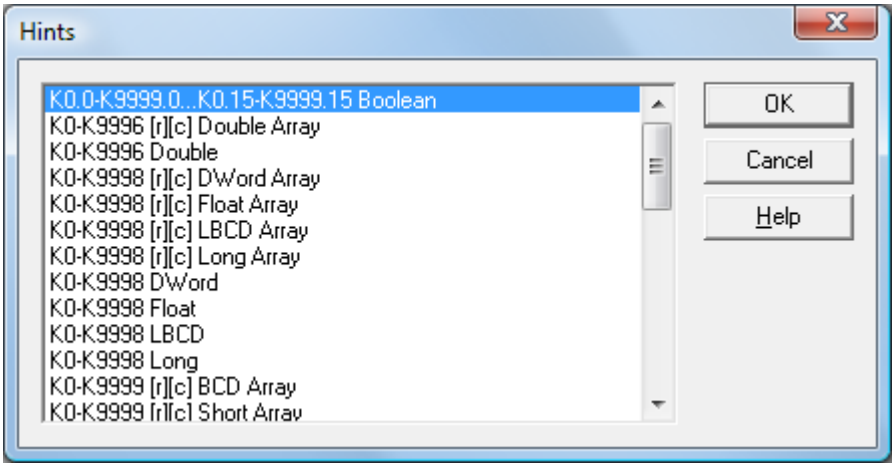
添加用户定义标记 (示例)

服务器可以采用两种方式将数据从设备传输到客户端应用程序。最常见的方法需要用户在服务器项目中定义一组标记，并使用先前为每个标记分配的名称，作为客户端与服务器之间各个链接的项。此方法使所有用户定义标记都能用于在 OPC 客户端中进行浏览。

用户定义标记支持缩放。有关详细信息，请参阅添加标记缩放。 某些情况下支持浏览并选择多个标记。有关详细信息，请参阅浏览标记。

1. 首先，请从"连接"树节点中选择设备名称。在此示例中，所选的设备是 "Device1"。
2. 单击**"编辑"|"连接"|"新建标记"**。或者，右键单击设备并选择**"新建标记"**。
3. 在**"标记属性"-"常规"**中，编辑属性以与以下内容匹配：
   - **"标记名称"**MyFirstTag
   - **"地址"**R000
   - **"说明"(可选)**我的第一个 Simulator 标记
   - **"数据类型"**字
   - **"客户端访问"**读/写
   - **"扫描速率"**100 毫秒。此属性不适用于 OPC 标记。

     注意：有关详细信息，请参阅标记属性 - 常规。
4. 如有必要，使用**"提示"**来确定驱动程序的正确设置。要调用提示，请单击"标记属性"中的问号图标。



注意“地址”、“数据类型”和“客户端访问”字段取决于通信驱动程序。例如，在 Simulator 驱动程序中，"R000" 是支持字数据类型且有读/写访问权限的有效地址。

5. 有关其他信息，请单击**“帮助”**。这将调用驱动程序帮助文档中的“地址说明”主题。
6. 按**“应用”**，将标记提交到服务器。标记现在应在服务器中可见。
7. 在此示例中，必须再添加一个标记，以在标记属性 - 缩放中使用。为此，请单击**“标记属性”-“常规”**中的**“新建”**图标。此操作会使属性恢复其默认设置。
8. 输入下列内容：
   - **“标记名称”**MySecondTag
   - **“地址”**K000
   - **“说明”**我的第一个缩放标记
   - **“数据类型”**短整型
   - **“客户端访问”**读/写
9. 下一步，按**“应用”**，将新标记提交到服务器。标记现在应在服务器中可见。

错误消息

输入标记信息时，用户可能会看到从服务器或驱动程序不时发出的错误消息。当用户尝试用与现有标记相同的名称添加标记时，服务器会生成错误消息。通信驱动程序生成错误的三个可能原因：

1. 有任何错误输入到地址的格式或内容中 (包括在某设备特定数据项的范围之内)。
2. 选定的数据类型不可用于地址。
3. 选定的客户端访问级别不可用于地址。

有关特定错误消息的详细信息，请参阅错误说明。

动态标记寻址

动态标记寻址只在客户端应用程序中定义标记。用户只需在直接访问设备地址的客户端中创建一个标记项，而无需在对服务器中创建的另一个标记项进行寻址的客户端中创建标记项。在客户端连接时，服务器会为该位置创建一个虚拟标记并自动开始数据扫描。

有关详细信息，请参阅动态标记。

提示：

1. 服务器可以为项目中的每个设备创建一个特殊布尔型标记，客户端可用其来确定该设备是否运行正常。要使用此标记，请将链接中的项指定为“错误”。如果设备通信正常，此标记为零，否则，此标记为一。

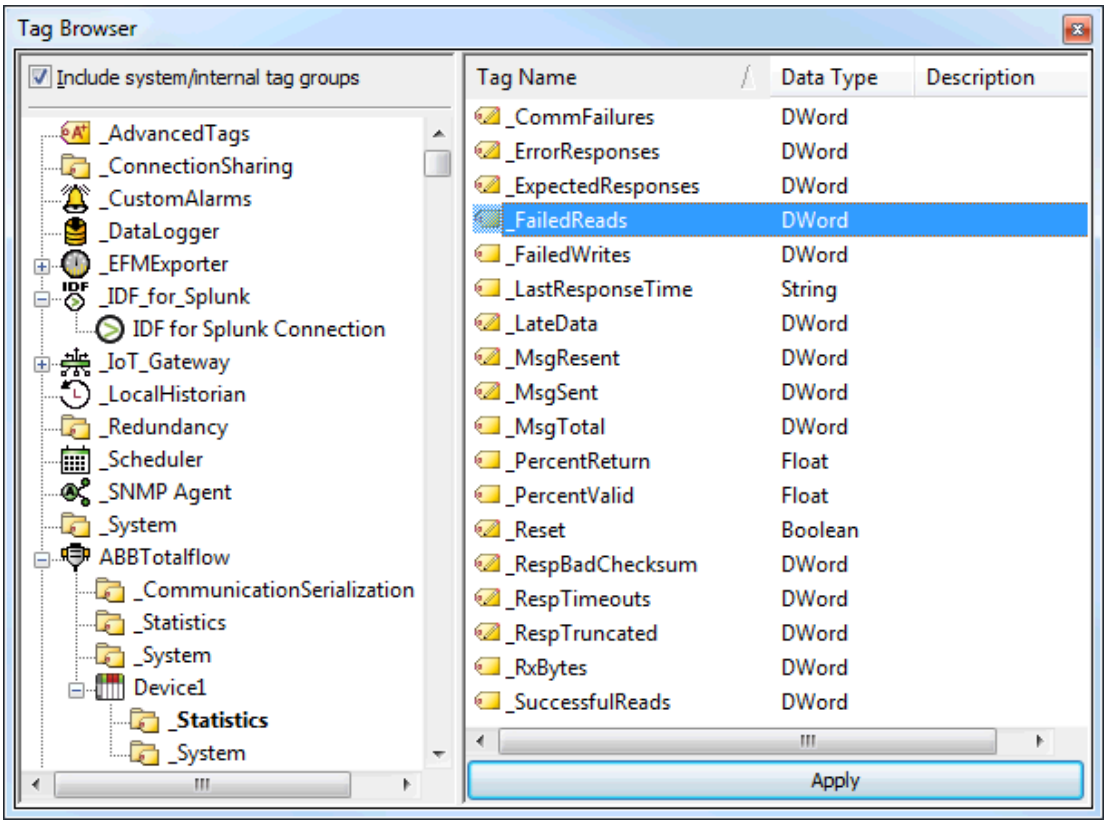
2. 如果省略了数据类型，驱动程序将根据所参考的设备和地址选择默认数据类型。所有位置的默认数据类型都记录在驱动程序的帮助文档中。如果指定的数据类型对于设备位置无效，则服务器会拒绝标记，并且事件日志中会发布错误。

3. 如果将设备地址用作链接的项 (以便地址与服务器中用户定义标记的名称相匹配)，链接会参考用户定义标记所指向的地址。在服务器全天在线运行的情况下，用户现在可以开始在 OPC 客户端中使用该项目。

浏览标记

服务器支持对可用标记进行浏览，在某些情况下，还可以选择将多个标记添加到项目中。

1. 访问“标记浏览器”对话框。



2. 如果**“包括系统/内部标记组”**可用，则启用此选项以使这些组可供选择。

3. 如果**“分支级别标记选择”**可用，则启用此选项以使左侧树视图 (可选择右侧的所有关联标记) 中的分支节点可供选择。

4. 导航左侧窗格中的树，以查找包含要添加的标记的分支。

5. 除非已启用**“分支级别标记选择”**，否则选择右侧窗格中的标记。如果支持添加多个标记，则可以使用标准键盘功能 (shift、Ctrl) 选择多个标记。

6. 单击**“应用”**。

另请参阅：添加用户标记

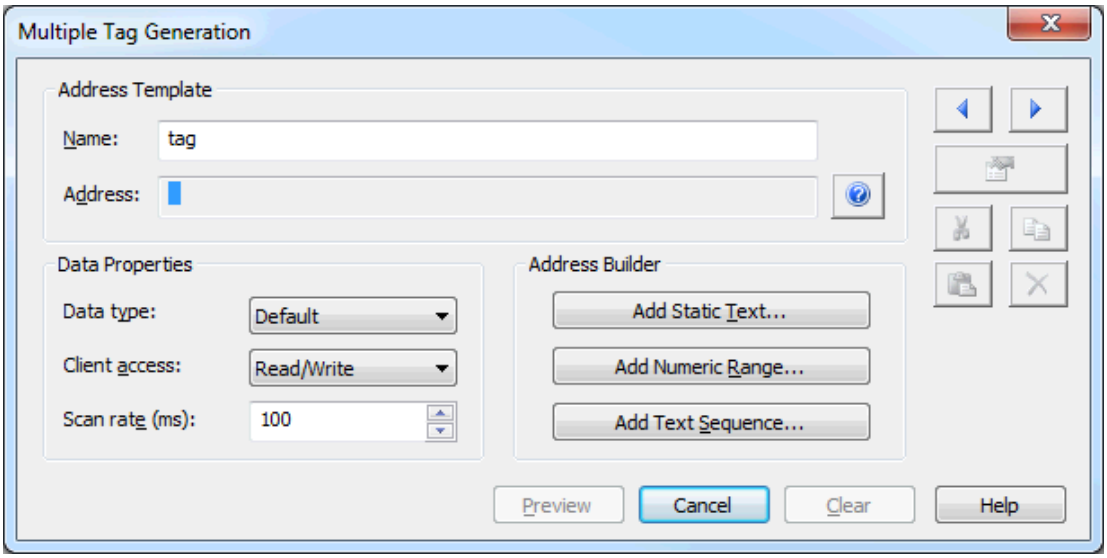
生成多个标记

“多标记生成工具”使用用户定义驱动程序命名法来动态创建标记。有关使用该工具的信息，请参阅以下说明。

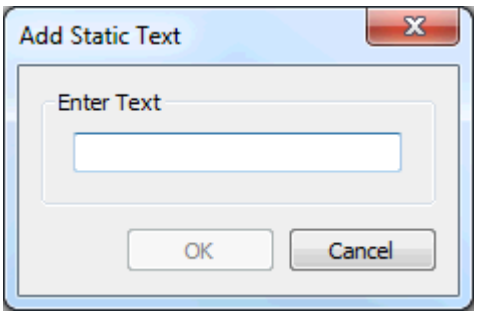
有关其属性的详细信息，请参阅多标记生成。

1. 首先，请选择设备，然后单击**“编辑”|“连接”|“新建标记”**。或者，右键单击设备并选择**“新建标记”**。
2. 在**“标记属性”**中，选择**“多标记生成”**图标 (位于“标识”属性的右下角)。

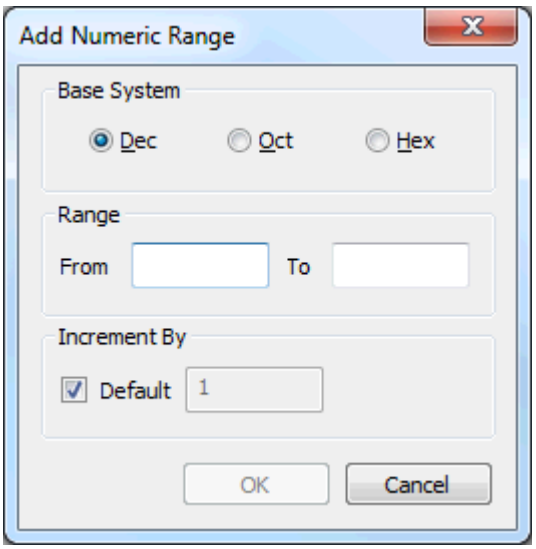
3. 在**“多标记生成”**中，定义标记名称，然后根据需要配置**“数据属性”**的属性。



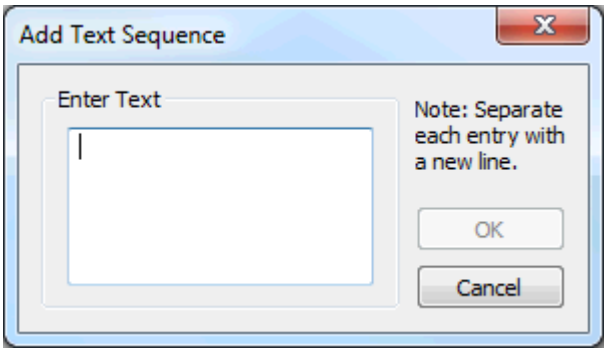
4. 单击**“添加静态文本”**。在此组中，根据需要输入文本。完成后，按**“确定”**。



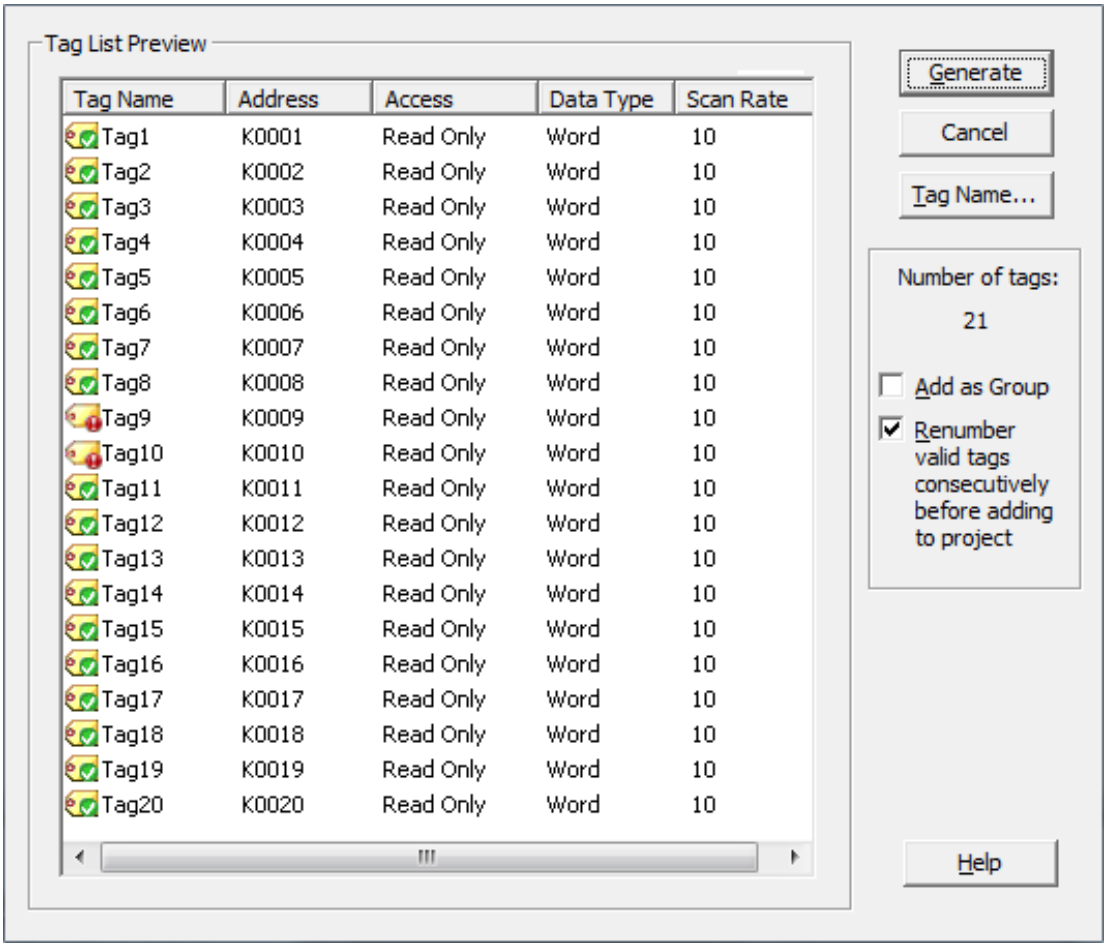
5. 单击**“添加数字范围”**。在此组中输入基本系统、范围和增量。完成后，按**“确定”**。



6. 单击**“添加文本序列”**。在此组中，根据需要输入文本。每个条目各占一行。完成后，按**“确定”**。

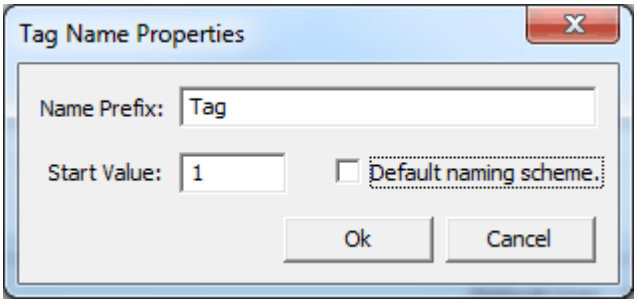


7. 单击**“预览”**。



注意：有效标记会显示绿色复选标记。无效标记会显示红色的 *x*。

8. 要将标记添加为组，请使用**“添加为组”**。
9. 要更改标记的名称或起始值，请选择**“标记名称”**。完成后，单击**“确定”**。



10. 要生成标记，请单击**“生成”**。如果生成成功，用户将返回到“多标记生成”对话框。
11. 单击**“关闭”**。然后单击**“确定”**。在标记显示窗口中应该能看到生成的标记。

另请参阅：多标记生成

添加标记缩放

在服务器中创建新标记时，用户可以选择应用标记缩放。此操作允许将设备的原始数据缩放到应用程序的相应范围。有两种类型的缩放：线性和平方根。有关详细信息，请参阅标记属性 - 缩放。

1. 首先，请打开标记的**“标记属性”**。
2. 打开**“缩放”**组。
3. 对于类型，请选择**“线性”**或**“平方根”**。
4. 通过高低值和限制来指定设备的期望数据范围。缩放数据类型还可让用户指定生成的缩放值在 OPC 客户端应用程序上的显示方式。

5. 在**“单位”**中，为 OPC 客户端指定字符串，以描述所生成工程值的格式或单位。要使用“单位”字段，需要能够访问“数据访问 2.0”标记属性数据的 OPC 客户端。如果客户端不支持这些功能，则无需配置此字段。
6. 如上所示输入数据后，单击**“确定”**。

保存项目

应该存在一个项目，该项目配置了可保存的用户定义标记。项目的保存方式取决于该项目是运行时项目还是离线项目。

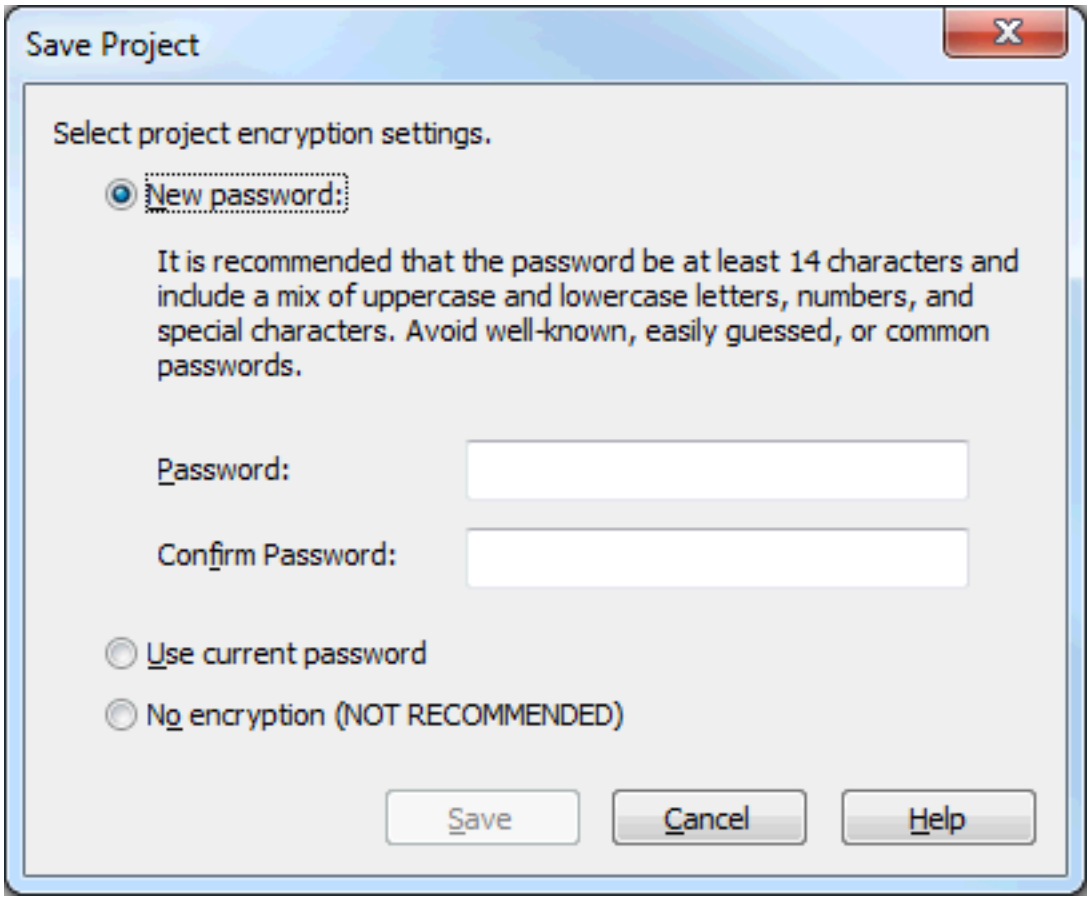
编辑运行时项目以后，利用服务器的全天在线运行可在项目保存到磁盘后立即从客户端访问标记。因为会更改实际项目，用户可以通过单击**“文件”|“保存”**进行保存。

项目文件有多个有效文件格式：.OPF、.SOPF 和 .JSON。.OPF 格式是未加密的二进制项目文件格式。.JSON (JavaScript 对象符号) 格式虽然使用便捷，但是人类可读且基于文本，是一个不太安全的选项，只有在其他安全措施就位时才可使用。建议用户将项目另存为 .SOPF 文件，因为此文件类型经过加密，是保存项目文件的最安全方式。

用户可以覆盖现有项目或将编辑内容另存为新项目，还可以选择将新项目加载为默认运行时项目。

通过选择**“文件”|“打开”**打开已保存的项目，以查找和选择项目文件。

编辑离线项目时，用户可以选择保存到同一个项目或另存为新项目。完成后，单击**“运行时”|“连接”**并将新项目加载为默认运行时项目。



如果保存新项目时启用了项目文件加密 (默认启用)，则必须设置密码。输入密码或选择**“不加密”** (不推荐)，然后单击**“确定”**。要修改密码，或者开启/关闭项目加密，请前往**“项目属性”|“常规”|“项目文件加密”**。单击**“取消”**可在不保存项目的情况下中止。

提示：建议密码至少为 14 个字符，并包括大写和小写字母、数字和特殊字符的组合。避免众所周知、容易猜到的或常见的密码。

通过密码加密的文件在保存时会另存为 .SOPF 文件。.JSON 和 .OPF 文件不支持加密项目。

注意：OPC 客户端应用程序会在客户端需要数据时，自动调用 OPC 服务器。但是，OPC 服务器需要知道，以此方式对其进行调用时要运行什么项目。服务器将加载上一次加载或配置的项目。要确定服务器将加载的项目，请查看在**"文件"**中的**"最近使用的"**文件列表。加载的项目是列出的第一个项目文件。

默认情况下，项目文件将保存到以下目录。

对于 64 位操作系统版本，项目文件保存在 (默认情况下) 以下目录中：
C:\Users\<用户名>\Documents\Kepware\KEPServerEX\V6

对于 32 位 OS 版本，项目文件保存在 (默认情况下) 以下目录中：
C:\Users\<用户名>\Documents\Kepware\KEPServerEX\V6

服务器会自动将项目的副本保存到以下目录中：

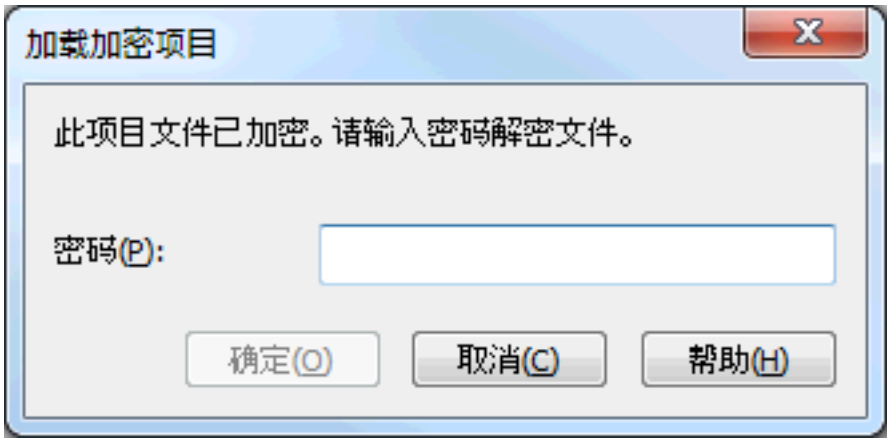
对于 64 位操作系统版本，项目文件保存在 (默认情况下) 以下目录中：
C:\ProgramData\Kepware\KEPServerEX\V6

对于 32 位 OS 版本，项目文件保存在 (默认情况下) 以下目录中：
C:\ProgramData(x86)\Kepware\KEPServerEX\V6

提示：如果文件已保存到另一个位置，搜索 *.opf、*.sopf 或 *.json 可查找可用的项目文件。

打开加密的项目

对于启用项目文件加密后保存的项目文件，当打开该文件时，系统会提示用户输入密码。

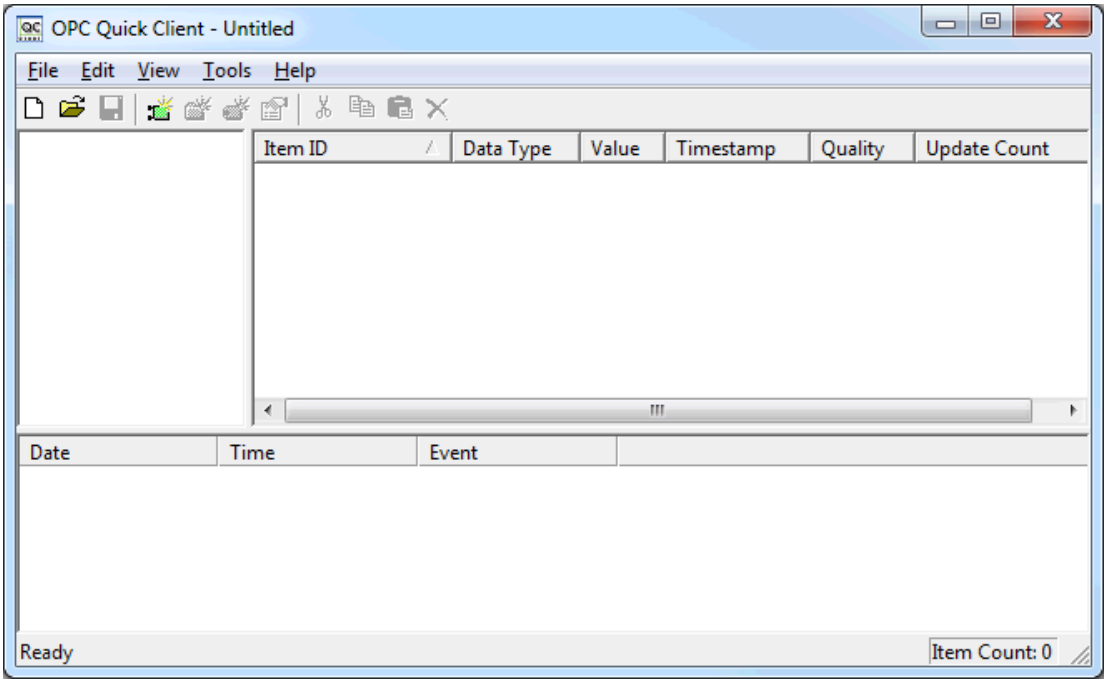


输入用于加密项目文件的密码，然后单击**"确定"**(单击**"取消"**将中止文件打开操作)。

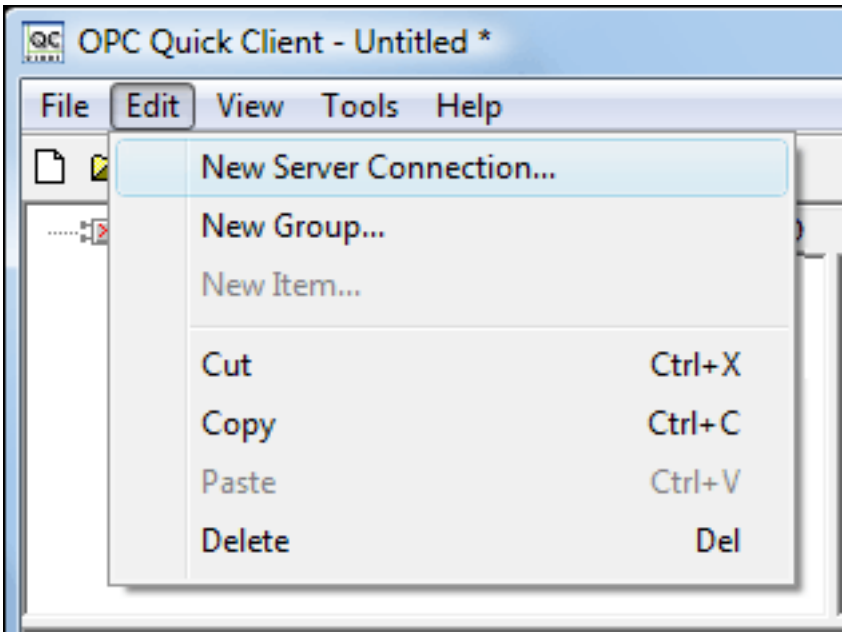
测试项目

服务器包括一个全功能 OPC Quick Client，其支持任何 OPC 客户端应用程序中的所有可用操作。Quick Client 可以访问服务器应用程序中的所有可用数据，并可用于读取和写入数据、执行结构化测试套件和测试服务器性能。它还提供有关服务器返回的任何 OPC 错误的详细反馈。

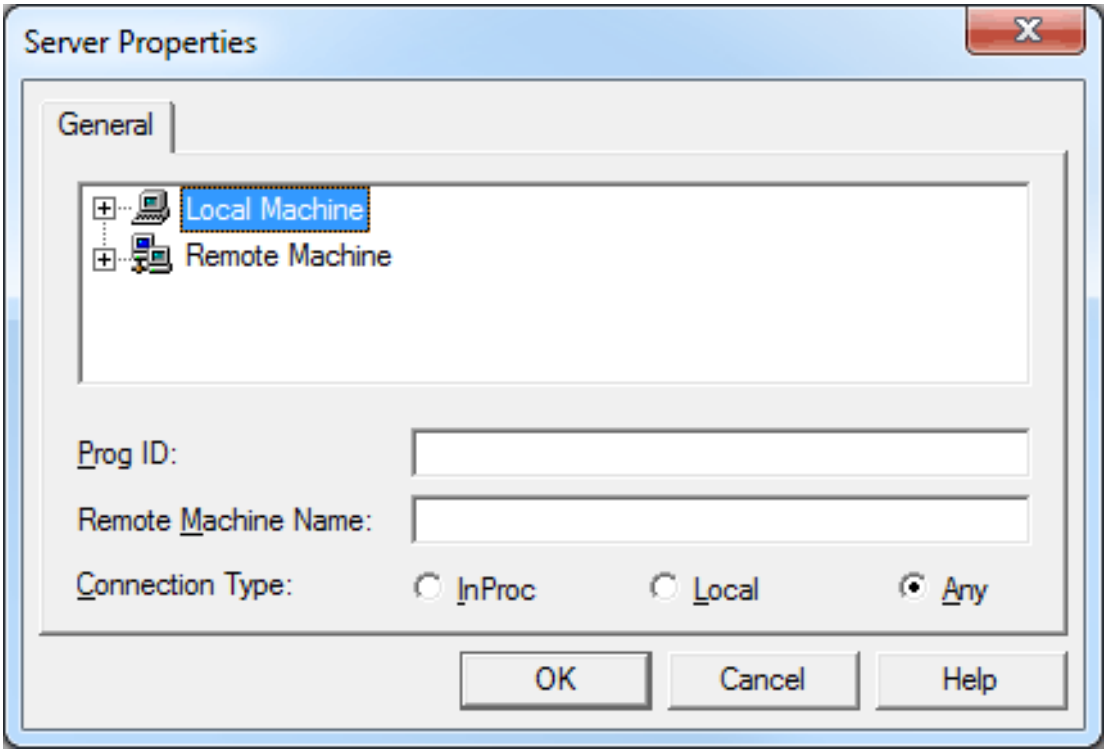
1. 首先，在与服务器相同的程序组中找到 OPC Quick Client 程序。然后，运行 OPC Quick Client。



2. 通过单击**“编辑”|“新建服务器连接”**来建立连接。

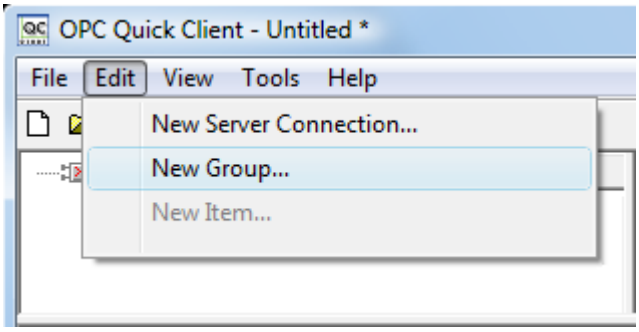


3. 在**“服务器属性”**中，请在本地或通过 DCOM 远程连接 OPC 服务器。默认情况下，会用服务器的“程序 ID”(OPC 客户端用其参考特定 OPC 服务器) 预配置此对话框。



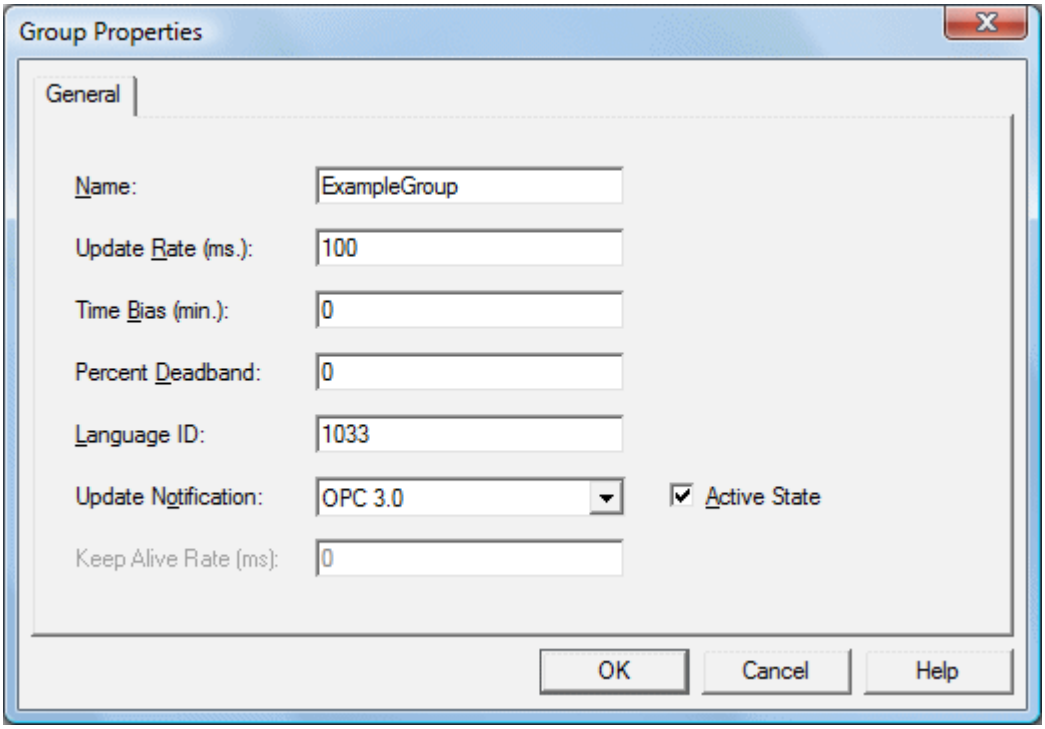
注意：建立连接后，可能会发生两种情况。如果服务器正在运行，OPC Quick Client 会与服务器建立连接。如果服务器未运行，它将自动启动。

4. 将组添加到连接。为此，请选择服务器连接，然后单击**"编辑"|"新建组"**。



注意："组"充当从服务器访问的所有标记的容器，并控制标记的更新方式。所有 OPC 客户端都使用组来访问 OPC 服务器数据。组中包含了若干属性，OPC 客户端可通过这些属性确定从标记读取数据的频率、标签处于活动状态还是非活动状态、死区是否适用等等。这些属性使 OPC 客户端能控制 OPC 服务器的运行方式。有关组属性的详细信息，请参阅 OPC Quick Client 帮助文档。

5. 为了本示例的目的，请编辑组属性使其与下图内容相符。



注意：“更新速率”、“死区百分比”和“活动状态”属性控制为组标记返回数据的时间以及是否返回。属性说明如下：

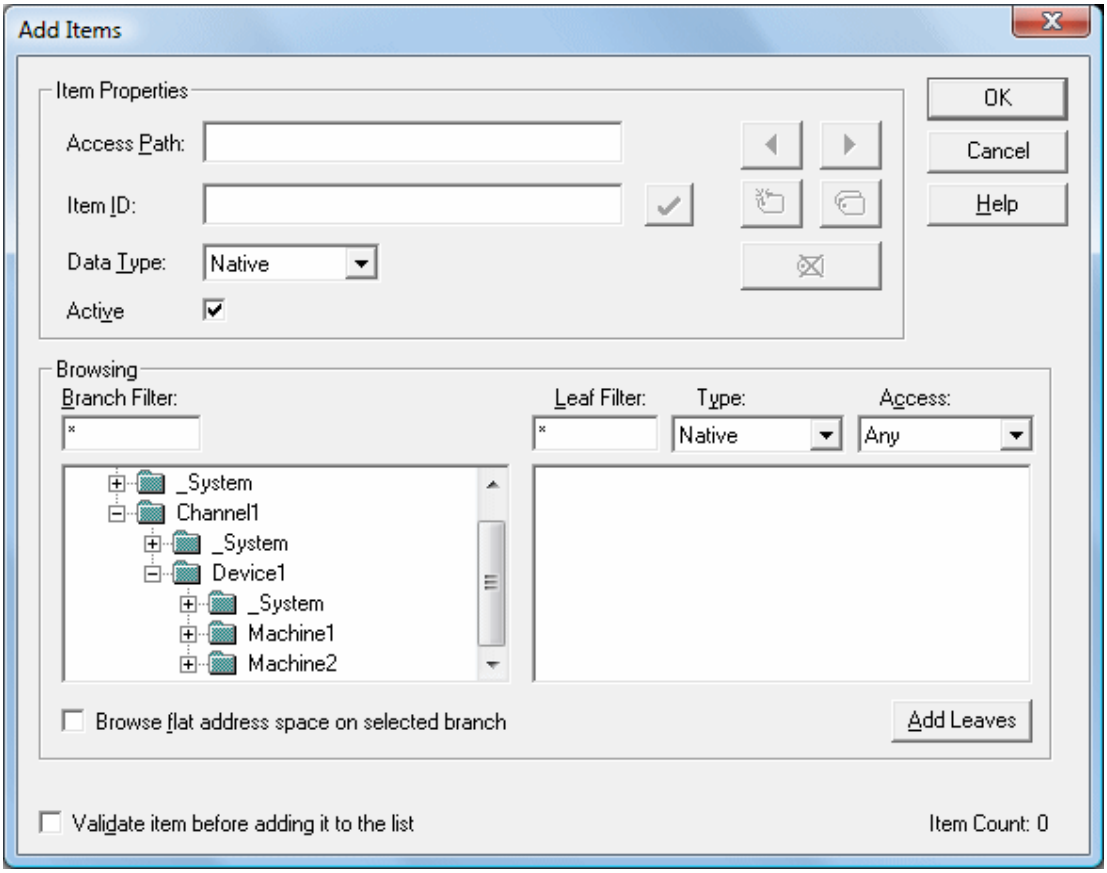
- **“名称”**：此属性用于客户端参考，实际上可以留空。
- **“更新速率”**：用于设置从实际设备扫描数据的频率以及作为扫描结果，将数据返回 OPC 客户端的频率。
- **“死区百分比”**：当更改超过所请求的更改百分比时，此属性只需通过检测更改就可以消除或减少数据中的干扰内容。百分比更改是给定标签数据类型的一个因子。
- **“活动状态”**：此属性会将此组中的全部标记打开或关闭。

6. 完成后，单击**“确定”**。

访问标记

必须先将 OPC 服务器标记添加到组，然后才能访问它们。OPC 数据访问规范将标记浏览接口定义为允许 OPC 客户端直接访问并在 OPC 服务器中显示可用标记。通过允许 OPC 客户端应用程序浏览 OPC 服务器的标记空间，单击所需标记将其自动添加到组。

1. 首先，请选择用于放置标记的组。单击**“编辑”|“新建项”**。

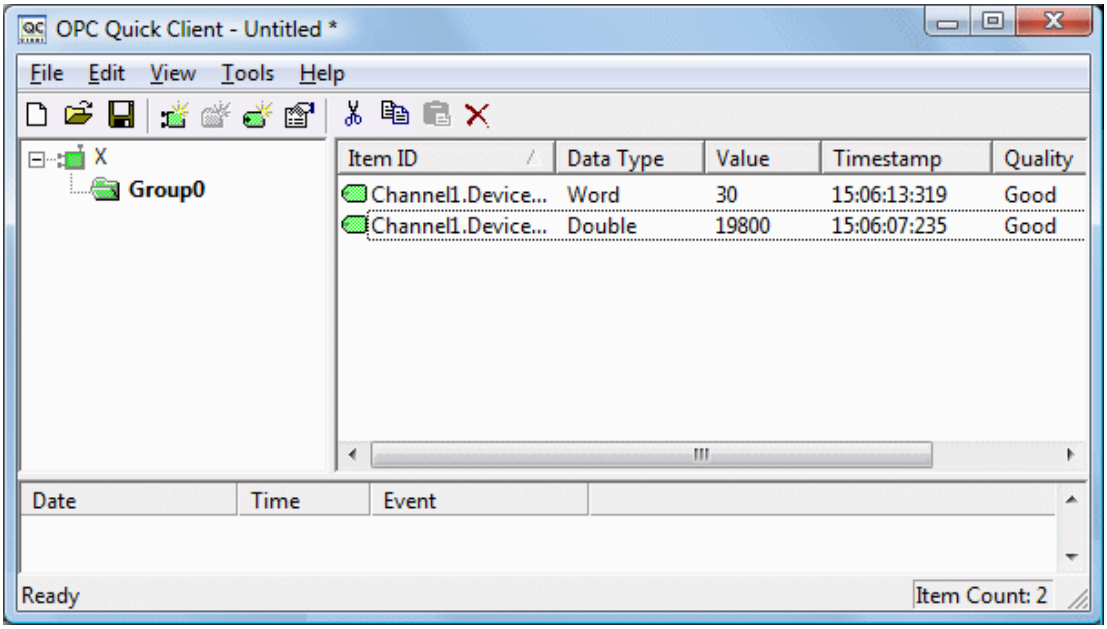


注意：“添加项”对话框还提供了“浏览”部分的树状视图，并可用于在 OPC 服务器中浏览，以便查找在服务器中配置的标记。在使用 "Example1" 项目时，用户可以通过展开视图的分支来访问先前定义的标记。

2. 当树层次结构处于上图所示状态时，用户就可以双击标记名称，开始将标记添加到 OPC 组中。将标记添加到组中后，在“添加项”对话框底部显示的**“项计数”**会增加以指示所添加项的数量。如果 "MyFirstTag" 和 "MySecondTag" 都被添加，则项计数应为 2。

3. 完成后，单击**“确定”**。

注意：用户现在应该能够利用两个已定义的标记访问服务器的数据。

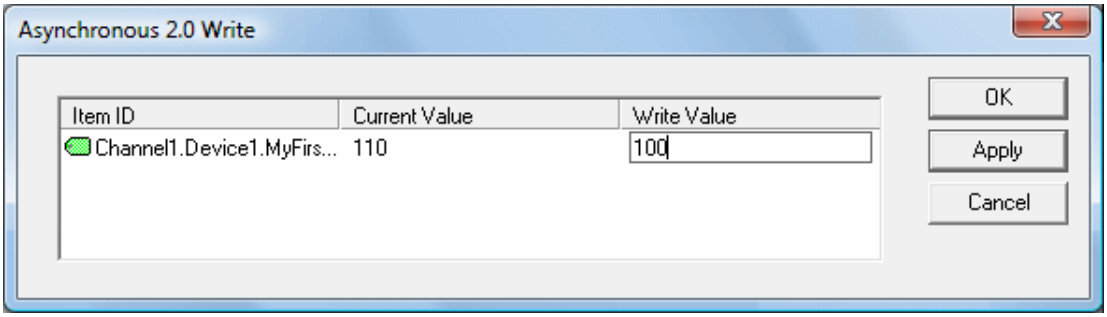


注意：第一个标记 "MyFirstTag" 应包含可变值。此时，第二个标记应为零。如果用户仅需要测试 OPC 项的读数，他们现在已完成。但是，如果用户需要更改 OPC 项，可使用一种写入方法来将新数据发送到 OPC 项。

将数据写入 OPC 服务器

OPC Quick Client 支持两种将数据写入 OPC 服务器的方法：同步写入和异步写入。同步写入会在 OPC 服务器上执行写入操作，并等待其完成。异步写入会在 OPC 服务器上执行写入，但不等待写入完成。将数据写入 OPC 项时，可以选择任意一种方法：不同的写入方式是 OPC 客户端应用程序设计中的重要因素。

1. 首先，请选择项。然后，右键单击并选择**“同步”**或**“异步写入”**。为了本示例的目的，右键单击 "MyFirstTag"，然后选择**“异步写入”**。



注意：尽管显示**“异步 2.0 写入”**对话框，但值会继续更新。

2. 要为此项输入新值，请单击**“写入值”**，然后输入其他值。
3. 单击**“应用”**以写入数据。此操作允许用户继续写入新值，而单击**“确定”**可写入新值并关闭对话框。
4. 单击**“确定”**。

注意：如果未输入任何新数据，单击**“确定”**不会将数据发送到服务器。

结论

到目前为止，构建和测试 OPC 项目涉及到的所有基本步骤都已讨论完毕。建议用户继续测试服务器和 OPC Quick Client 的各种功能以获得更好的认识和理解。有关 OPC Quick Client 的详细信息，请参阅其帮助文档。

用户现在可以开始 OPC 应用程序的开发。如果使用 Visual Basic，请参阅提供的示例项目。这两个项目提供如何在 Visual Basic 应用程序中直接使用 OPC 技术的一个简单示例和一个复杂示例。

如何...

有关详细信息，请从下表中选择一个链接。

允许桌面交互作用
创建和使用别名
优化服务器项目
处理数组数据
正确命名通道、设备、标记和标记组
连接到服务器的 DNS/DHCP 设备重新上电时，解决通信问题
选择合适的网线
使用别名来优化项目
服务器上使用 DDE
使用动态标记寻址
使用以太网封装
使用非规范化浮点值

允许桌面交互作用

有些通信接口要求服务器与桌面进行交互。例如，DDE 和 FastDDE 会使用 Windows 消息层。在选择如何与桌面进行通信时，务必考虑操作系统这个因素。

Windows Vista、Windows Server 2008 及更新的操作系统

在 Windows Vista、Windows Server 2008 及更新的操作系统中，服务运行于隔离的会话中，而登录到控制台的用户无法访问该会话。这些操作系统要求将进程模式设置为“交互式”。可在与当前用户相同的用户帐户中运行“运行时”。有关更改进程模式的信息，请参阅设置 - 运行时进程。

Windows XP、Windows Server 2003 及之前版本的操作系统

在 Windows XP、Windows Server 2003 及之前版本的操作系统中，可仍将进程模式设置为“系统服务”。然而，必须允许运行时服务与桌面交互。这是首选的操作模式，因为用户无需登录到供服务器启动的控制台。有关允许服务与桌面交互的信息，请参阅以下说明。

注意：仅当服务器在“服务模式”下运行时才适用这些服务设置。

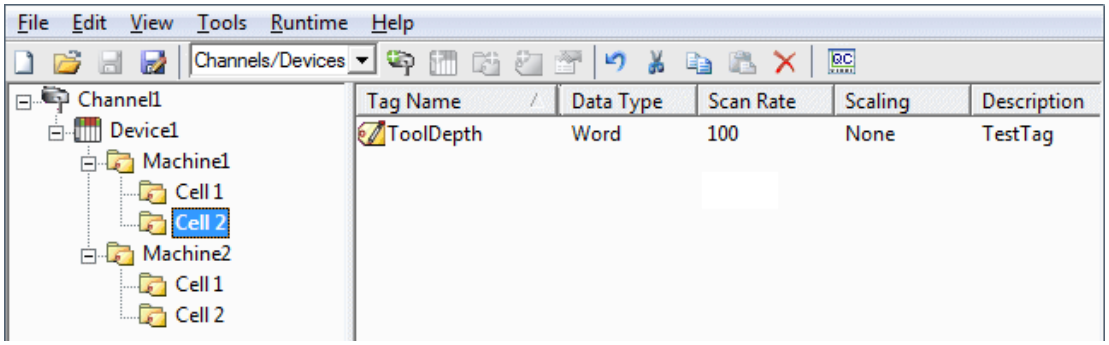
1. 首先，请启动**“服务”**管理单元 (属于 **Microsoft 管理控制台**的一部分)。要执行此操作，请单击 **“开始” | “运行”**。
2. 键入“services.msc”并单击**“确定”**。然后，从服务列表中根据名称查找服务器。打开其上下文菜单并选择**“属性”**。
3. 打开**“登录”**组并启用**“允许服务与桌面交互”**。
4. 单击**“应用”**。
5. 单击**“确定”**以退出。
6. 查找“管理”图标。打开其上下文菜单并选择**“停止运行时服务”**。
7. 然后，重新打开上下文菜单并选择**“启动运行时服务”**。

另请参阅：
访问管理菜单

创建和使用别名

复杂标记引用示例

下图显示了服务器中的复杂标记引用。



例如，要创建指向 "ToolDepth" 标记的应用程序的 DDE 链接，必须将 DDE 链接输入为“<DDE 服务名称>|_ddedata!Channel1.Device1.Machine1.Cell2.ToolDepth”。

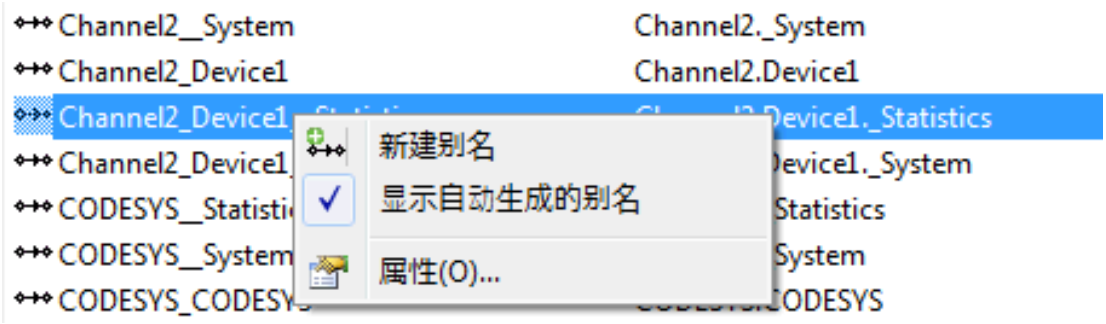
尽管 DDE 链接的 *<应用程序>|<主题>!<项目>* 格式仍然存在，然而，主题中需要包含可选标记组和通道名称时，该内容会变得更加复杂。别名映射允许在 DDE 客户端应用程序中使用较短版本的参考。

有关详细信息，请参阅什么是别名映射。

为复杂地址路径创建别名

有关通过创建别名来简化复杂标记地址路径的信息，请遵循以下说明。

1. 在树视图中，选择要编辑的别名，然后双击以打开别名节点。
2. 在详细信息视图中，单击鼠标右键并选择**“新建别名”**(或选择**“编辑”|“别名”|“新建别名”**)。



3. 浏览至包含要引用项目的组或设备。

4. 输入表示复杂标记引用的别名。现在，此别名即可用来在客户端应用程序中对在服务器中找到的标记进行寻址。*有关保留字符的信息，请参阅如何... 正确命名通道、设备、标记和标记组。*
5. 复杂主题和项目名称 "_ddedata! Channel1.Device1.Machine1.Cell2" 可通过使用别名 "Mac1Cell2" 进行替换。应用到上述示例中，应用程序中的 DDE 链接即可输入为“<DDE 服务名称>|Mac1Cell2！ ToolDepth”。

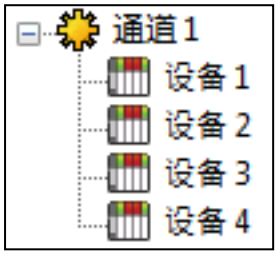
注意：虽然用户创建的别名可以与通道共享名称，但不建议这样做。如果客户端项目引用了使用共享名称的动态地址，则会失败。例如，如果将别名命名为 "Channel1" 并映射到 "Channel1.Device1"，则客户端中引用 "Channel1.Device1.<地址>" 的项目将无效。必须移除或重命名此别名，客户端的引用才能成功。

另请参阅：别名属性

优化服务器项目

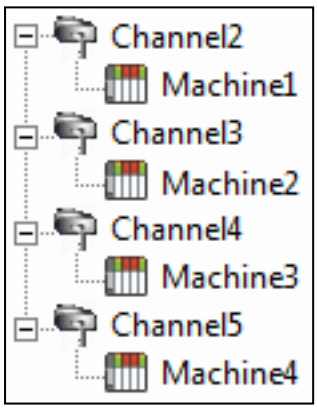
该服务器的每个驱动程序几乎都支持至少 100 个通道；这意味着需要 100 个 COM/串行端口或 100 个源套接字才能实现以太网通信。要确定可用于每个设备的支持通道数，请参阅 服务器汇总信息下的驱动程序信息。

该服务器将通信协议称为通道。应用程序中定义的每个通道都表示服务器中一个单独的执行路径。一旦定义了通道，必须在该通道下定义一系列设备。每一台此类设备都代表一个可从中收集数据的单一设备。虽然这种定义应用程序的方法提供了高水平的性能，但它不能充分利用驱动程序或网络。下面显示了使用单个通道配置时应用程序所呈现效果的示例。



每个设备均出现在单个通道下。在此配置中，驱动程序必须尽快从一个设备移动到下一个设备，以有效速率收集信息。随着更多设备的添加或从单个设备请求的信息的增加，整体更新速率会受到不利影响。

如果驱动程序只能定义一个单通道，如上所示的示例为唯一可用的选项。但是，使用多个通道可通过同时向网络发出多个请求来分发数据收集工作负载。下面显示了使用多个通道来提高性能时相同应用程序所呈现效果的示例。



当前，每个设备已在其自身的通道下定义。在这个新配置中，单个执行路径专用于从每个设备收集数据。如果应用程序拥有的设备数较少，则可对其进行精确优化，如此处所示。

即使应用程序拥有的设备数大于通道数，也可改善性能。虽然理想情况是每个通道对应 1 台设备，但应用程序也将受益于其他通道。尽管在全部通道上分散设备负载会使服务器再次从一台设备移动到另一台设备，但是，这样可以用极少的设备在单通道上进行处理。

- 这一过程可用于建立与一台以太网设备的多个连接。尽管 OPC 服务器可允许大多数驱动程序使用 100 个通道，但设备会最终确定允许的连接数。此约束来自于大多数设备限制支持的连接数。与设备建立的连接越多，在每个连接上处理请求的时间就越短。这意味着，添加连接时，性能可能会大打折扣。

处理阵列数据

许多适用于此服务器的驱动程序允许客户端访问数组格式的数据。数组允许客户端应用程序在一个请求中请求连续数据的特定集合。数组是一种特定的数据类型。用户不能有一个具有“字”和“双字”数据类型组合的数组。此外，数组被写入一个事务中。要在服务器中使用数组，客户端应用程序必须至少支持读取数组数据的功能。

在 DDE 客户端中处理数组数据

仅在使用 CF_TEXT 或高级 DDE 剪贴板格式时，数组数据才在客户端可用。

对于使用高级 DDE 的客户端应用程序，将在 SPACKDDE_DATAHDR_TAG 结构中指定数组中的元素数。此协议仅支持一维数组。当向服务器传输数组数据时，应使用此结构。

使用 CF_TEXT 的客户端可支持一维或二维数组。各行中的数据使用 TAB (0x09) 字符分隔，各行使用 CR (0x0d) 字符和 LF (0x0a) 字符终止。当客户端想存入数据值数组时，写入的字符串应具有此分隔符格式。

无论以何种格式传输数组标记，都不需要写入整个数组，但起始位置是固定的。如果试图向未声明为数组的标记存入数组格式数据，则仅需写入数组中的第一个值。如果尝试存入多于标记数组大小的数据，则仅能写入标记数组大小的数据。如果在尝试存入数据时留空某些数据值，则在重新写入设备时，服务器将使用该数

组元素的最后一个已知值。如果该寄存器中的值已更改，但尚未在服务器中更新，则该值将被旧值覆盖。因此，在向数组中写入数据时应谨慎。

在 OPC 客户端中处理数组数据

在支持数组的 OPC 客户端中，OPC 项数据值实际为变体数组数据类型。OPC 客户端将解析数组元素数据：某些客户端将创建用于显示的子标记。例如，如果 OPC 客户端在其名为"Process" 的数据库中创建了一个标记，并且关联的 OPC 项是由 5 个元素组成的一维数组，则它可以创建 5 个标记 (名为 "Process_1"、"Process2" 等)。其他客户端 (如 OPC Quick Client) 可能显示为以逗号分隔值 (CSV) 数据。

正确命名通道、设备、标记和标记组

命名通道、设备、标记或标记组时，以下字符为保留或受限字符：

- 句点
- 双引号
- 前导下划线
- 前导或尾随空格

注意：在特定情况下可使用某些受限字符。有关详细信息，请参阅下表。

1. 在别名中使用句点来分隔原始通道名称和设备名称。例如，有效的名称为“Channel1.Device1”。
2. 可在第一个字符后使用下划线。例如，有效的名称为“Tag_1”。
3. 名称中可使用空格。例如，有效的名称为“Tag 1”。

连接到服务器的 DNS/DHCP 设备重新上电时，解决通信问题

某些驱动程序支持对连接进行 DNS/DHCP 解析，以允许用户分配唯一的域/网络名供标识用。启动及连接网络时，设备会从网络 DNS 服务器请求 IP 地址。将域名解析为 IP 地址进行连接的过程比较耗时。为提高速度，操作系统会缓存所有已解析的 IP/域名并重新使用。默认情况下，已解析的名称会在缓存中保存两个小时。

如果与设备的域/网络相关联的 IP 地址名称发生了改变，服务器将无法重新连接该设备。如果造成这一改变的原因是对设备进行了重新上电，则会获取新的 IP。造成这一改变的原因还可能是手动改变了设备的 IP。在这两种情况下，使用中的 IP 地址均会消失。

因为服务器每 30 秒会自动刷新缓存，会对 IP 进行强制解析。如果这无法解决该问题，用户可在 PC 的命令提示中键入命令字符串“ipconfig/flushdns”来手动刷新缓存。

有关详细信息，请参阅以下 Microsoft 支持文章：禁用和修改客户端 DNS 缓存。

选择合适的网线

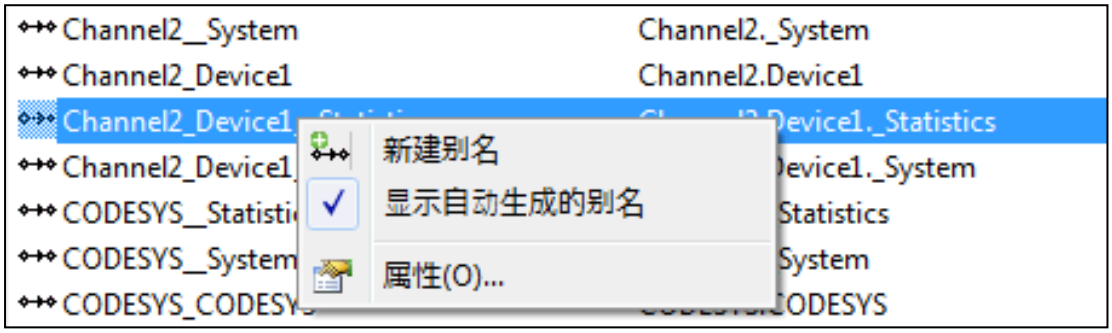
如果之前没有使用启用以太网的设备或串口转以太网转换器的经验，那么用户很难选择合适的网线。通常，有两种方法可用于确定网线设置是否正确。如果通过网络集线器或交换机连接设备或转换器，用户需要使用**跳接线**。以前人们通过电话接线员使用的电路板对设备进行跳接或相互连接，跳接线由此而得名。但如果是从 PC 直接连接设备，用户则需要使用**交叉电缆**。这两种网线都可以通过电子元件或 PC 供应商购买。

使用别名来优化项目

要获得项目的最佳性能，建议将每个设备置于其自己的通道上。如果在创建项目后需对其进行优化以便通信，可能难以将客户端应用程序更改为引用新的项目名称。不过，通过使用别名映射，用户可以允许客户端向新“配置”发出旧请求。要启动，请按照以下说明。

1. 首先，为每个设备都创建一个新通道。将设备置于新通道下，并删除原始通道。
2. 在树状视图的“别名”下，为**别名映射**中的每台设备创建**新别名**。别名是以句点分隔的原始通道和设备名称。例如，"Channel1.Device1"。

有关保留字符的信息，请参阅如何正确命名通道、设备、标记和标记组。



注意：在将响应返回客户端应用程序并显示该项目不存在的错误之前，服务器会根据别名映射验证项目的任何请求。

在服务器上使用 DDE

在应用程序中使用 DDE

动态数据交换 (DDE) 是一种 Microsoft 通信协议，它提供一种方法在 Windows 操作系统上运行的应用程序之间交换数据。DDE 客户端程序打开通向 DDE 服务器应用程序的通道，并使用应用程序 (服务) 名称、主题名称和项目名称的层次结构来请求项目数据。

DDE 客户端连接到服务器界面，必须允许运行时与桌面交互。
有关详细信息，请参阅 如何允许桌面交互作用。

示例 1：本地访问寄存器 (使用默认主题)

语法为 *<应用程序>|<主题>!<项>*，其中：

- **应用程序** DDE 服务名称
- **主题** _ddedata *
- **项** Modbus.PLC1.40001

*这是不使用别名映射条目的所有 DDE 数据的默认主题。

注意： 语法的一个示例是"MyDDE | _ddedata ！ Modbus.PLC1.40001"。

示例 2：本地访问寄存器 (使用别名作为主题)

语法为 *<应用程序> | <主题>！ <项>*，其中：

- **应用程序** DDE 服务名称
- **主题** ModPLC1 *
- **项** 40001

* 这是使用别名映射条目的主题。

注意： 语法的一个示例是"MyDDE |ModPLC1 ！ 40001" . 有关其他可能的语法，请参阅 DDE 客户端的特定帮助文档。

另请参阅：
项目属性 - DDE

项目属性-FastDDE & SuiteLink
什么是别名映射？

使用动态标记寻址

此服务器还可用于动态地从服务器引用物理设备数据地址。服务器动态创建所请求的项的标记。用户无法从一个客户端浏览通过另一个客户端动态添加的标记。动态添加标记之前，用户应注意以下事项：

- 正确的语法必须用于数据地址。有关特定驱动程序的语法的详细信息，请参阅帮助文档。
- 如果用户没有指定所请求项目的数据类型，则应用程序将其设置为默认设置。有关特定的驱动程序支持的数据类型的详细信息，请参阅帮助文档。

注意：在下面的示例中，Simulator 驱动程序的通道名称为 'Channel1'，设备名称为 'Device1'。

示例 1：在非 OPC 客户端中使用动态标记寻址

要在模拟设备中从寄存器 'K0001' 获取数据，请使用 "Channel1.Device1.K001." 的项 ID此注册的默认数据类型为短整型。由于非 OPC 客户端不提供服务器的更新速率，因此动态标记的默认更新速率为 100 毫秒。发送动态请求后，数据类型和更新速率都可以被覆盖。

要覆盖标记默认值，请在该项的末尾使用商用 AT 符号 ('@')。如果要将寄存器添加为双字 (无符号 32 位) 数据类型，请使用 "Channel1.Device1.K0001@DWord." 的项 ID要将默认更新速率更改为 1000 毫秒，请使用 "Channel1.Device1.K0001@1000."要更改这两个默认值，请使用"Channel1.Device1.K0001@DWord,1000"。

注意：客户端应用程序必须能够在其地址空间中接受特殊字符，如 "@"。

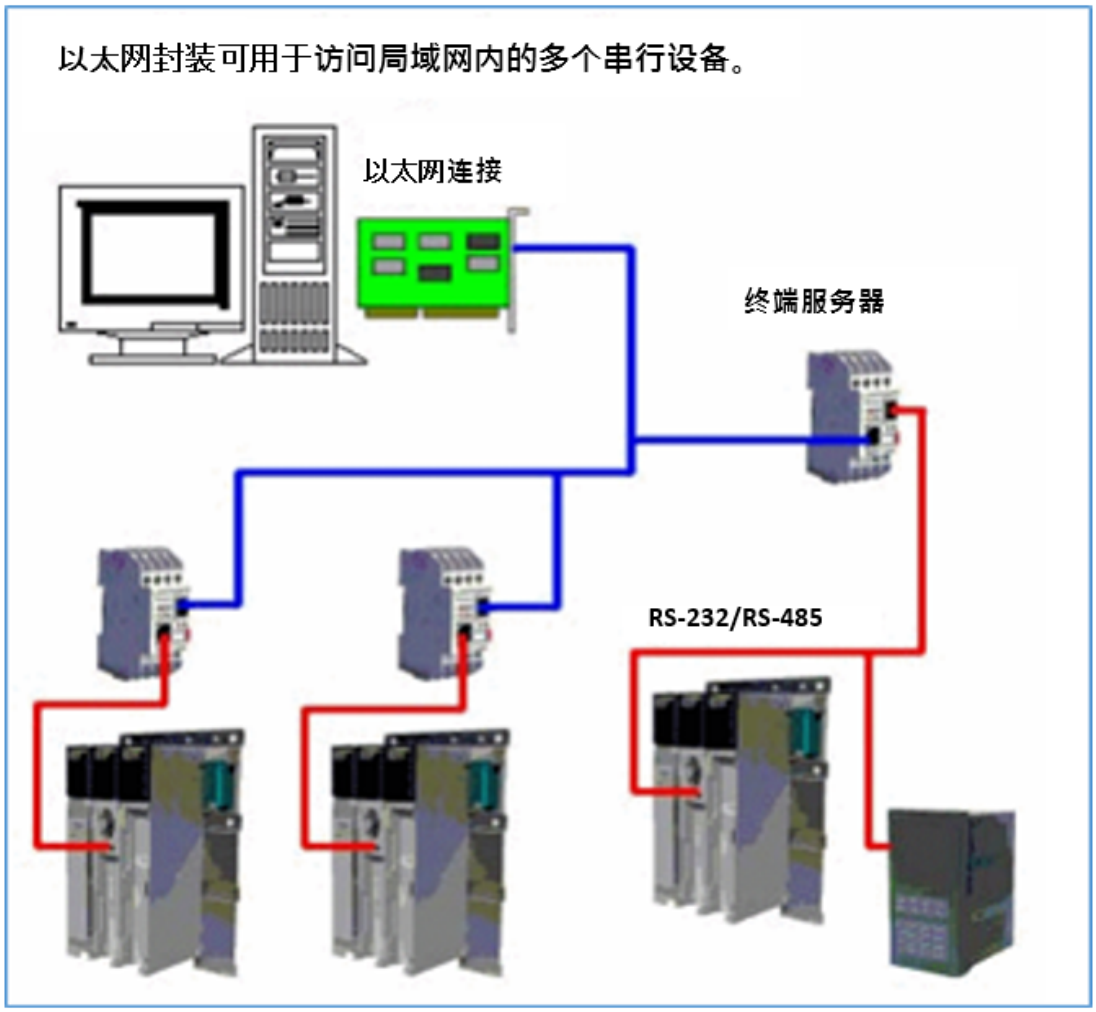
示例 2：在 OPC 客户端中使用动态标记寻址

在 OPC 客户端中，如果在添加 OPC 项时客户端应用程序不提供指定数据类型的方式，则可以使用相同的语法来覆盖数据类型。由于在 OPC 中使用项的更新速度，没有必要要覆盖它。

注意：客户端应用程序必须能够在其地址空间中接受特殊字符，如 "@"。

使用以太网封装

“以太网封装”模式旨在为通过以太网与终端服务器相连的串行设备提供通信。终端服务器本质上是将以太网上的 TCP/IP 消息转换为串行数据的虚拟串行端口。消息转换为串行形式后，用户可将支持串行通信的标准设备连接到终端服务器。下图显示了如何使用“以太网封装”模式。



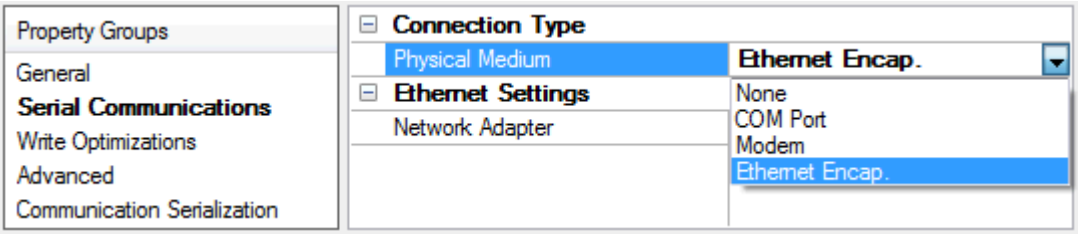
注意：对于支持"以太网封装"的主动驱动程序，用户必须在通道级别配置端口和协议设置。这样便可将驱动程序绑定到指定的端口并处理从多个设备传入的请求。由于通道接受来自所有设备的传入请求，所以不会在通道上输入 IP 地址。

"以太网封装"可用于无线网络连接 (例如 802.11 b 和 CDPD 数据包网络)，并且经过开发可以支持多种串行设备。通过使用终端服务器设备，用户可在所有工厂操作中放置 RS-232 和 RS-485 设备，但仍然允许单个本地化 PC 访问远程挂载设备。此外，"以太网封装"模式还可以根据需要将各个网络 IP 地址分配到各个设备。在使用多个终端服务器的同时，用户还可以从单个 PC 访问数百个串行设备。

配置"以太网封装"模式

要启用"以太网封装"模式，请打开**"通道属性"**，然后选择**"串行通信"**组。在**"连接类型"**下拉菜单中，选择**"以太网封装"**。

注意：只有支持"以太网封装"的驱动程序才允许选择该选项。



注意：服务器的多通道在每个驱动程序协议上最多支持 16 个通道。这使得用户可以指定一个通道使用本地 PC 串行端口，指定另外一个通道使用"以太网封装"模式。

当选择“以太网封装”模式时，串行端口设置 (如波特率、数据位和奇偶校验) 不可用。在为“以太网封装”模式配置通道之后，用户必须配置设备以便进行以太网操作。当新设备添加到通道时，“以太网封装”设置可用于选择以太网 IP 地址、以太网端口号和以太网协议。

注意：所使用终端服务器的串行端口必须配置为符合要连接到终端服务器的串行设备的要求。

设备需求轮询

设备需求轮询对于需要从其客户端应用程序对轮询设备进行完全控制的客户很有用。这在 SCADA 行业尤其有用，例如：石油和天然气、水/废水处理、电力和其他可能会遇到严重通信延迟的工作环境。

许多客户端 SCADA 系统或者没有可配置的扫描速率，或者其扫描速率的最小值对于 SCADA 操作员所需的数据更新来说太长。为了绕过此限制，SCADA 系统可以对服务器中可用的设备请求轮询标记执行写入操作。在这一方案中，服务器中的每个设备都公开一个 _DemandPoll 标记，当客户端写入该设备时，会轮询所有引用的标记。轮询期间，_DemandPoll 标记变为 True (1)。当最终的启用标记指示读取请求已完成时，返回 False (0)。在标记值返回 False 前，后续写入 _DemandPoll 标记都将失败。需求轮询会遵循通道的读/写占空比。可以开发客户端 SCADA 脚本 (例如：“刷新”按钮脚本) 来写入 _DemandPoll 标记并触发轮询。轮询结果将传递给客户端应用程序。有关详细信息，请参阅系统标记。

注意：上述过程不是 OPC 合规行为。如果这是一个问题，建议将通信分散到两个设备上。一个设备可以使用传统的 OPC 更新间隔，另一个设备可以将扫描模式设置为“不扫描，仅按需求轮询”，且只在写入 _DemandPoll 标记时进行轮询。

无论是否使用设备需求轮询，受标记扫描速率限制的客户端也可能遇到操作员等待时间，这是由于服务器遵从 OPC 客户端的组更新速率。为了规避这一 OPC 合规行为，用户可以配置“忽略组更新速率，数据可用后立即将其返回”设置。这将立即返回轮询结果并忽略更新间隔。有关详细信息，请参阅项目属性 - OPC DA 合规。

另请参阅：设备属性 - 扫描模式

配置 API 服务

配置 API 允许 HTTP RESTful 客户端添加、编辑、读取和删除服务器内的信道、设备和标记等对象。配置 API 提供以下功能：

- 标准用户可读 JSON 数据格式的对象定义
- 支持服务器中某些对象的触发和监控操作
- 通过 HTTP 基本验证和 HTTP over SSL (HTTPS) 的安全性
- 支持基于用户管理器和安全策略插件的用户级访问
- 具有可配置详细程度和保留级别的事务处理日志记录

注意：此文档假定您熟悉 HTTP 通信和 REST 概念。

初始化 - 配置 API 作为 Windows 服务进行安装，并随系统自动启动。
操作 - 配置 API 支持服务器与 REST 客户端之间的连接和命令。
关闭 - 如果必须停止配置 API，请使用 Windows 服务控制管理器来终止配置 API 服务。

安全性

配置 API 的 REST 客户端必须使用 HTTP 基本验证。用户凭证将在服务器用户管理器中定义。

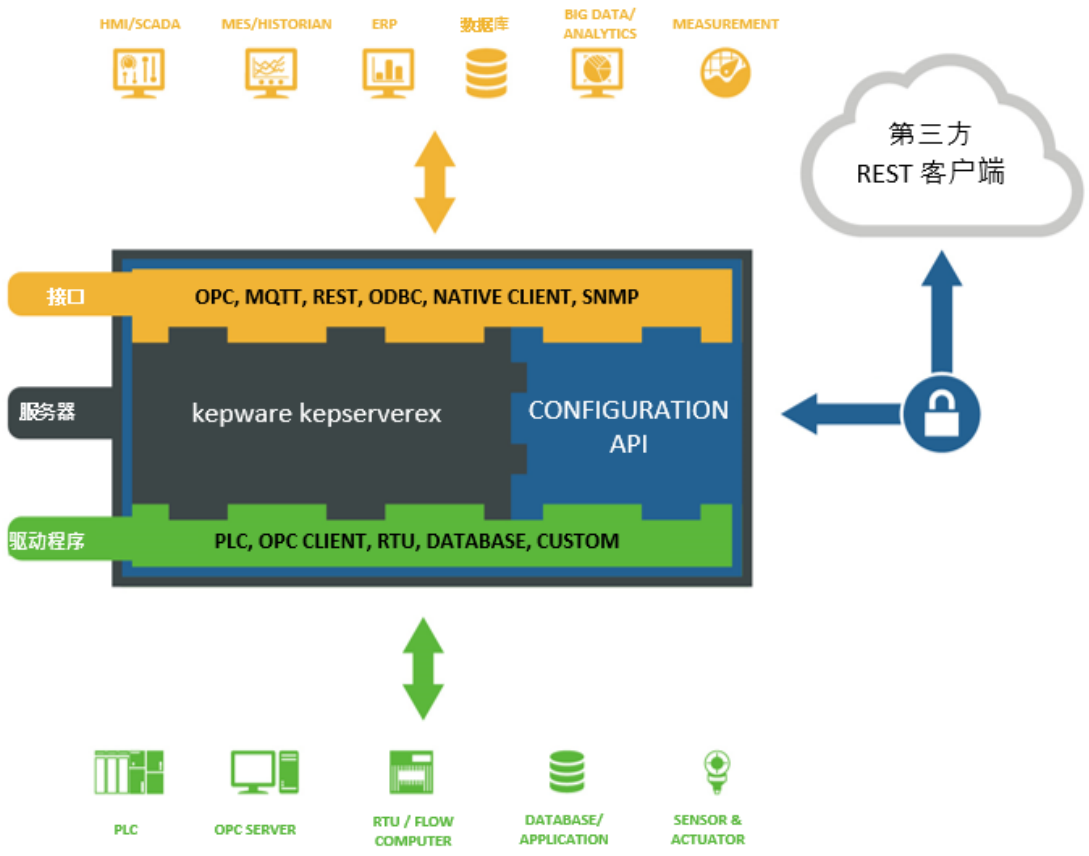
文档

有关属性、数据范围、端点映射方案的其他信息，以及配置 API 登录页面上每个端点的可接受操作，请参阅 http://localhost:57412/config/ (适用于默认配置)。

*默认情况下，登录页面中提供的文档采用 HTML 编码。*要获取 JSON 编码的文档，请包括一个以 "application/json" 开头的“接受”请求。

配置 API 服务 - 体系结构

下图显示了组件的布局。已在连接服务器的同一台计算机上安装了“配置 API 服务”。



配置 API 服务 - 并发客户端

配置 API 可同时为多个 REST 客户端提供服务。为防止客户端编辑过期配置，服务器运行时会维护数字项目 ID。每次通过配置 API 或本地配置客户端编辑对象时，项目 ID 均会发生更改。当前项目 ID 会在每个 GET 响应中返回。在所有 PUT 请求中，当前项目 ID 都必须由客户端指定。

最佳做法是发出 GET 请求、保存当前项目 ID，然后将该 ID 用于以下 PUT 请求。如果仅使用一个客户端，客户端可能会将属性 "FORCE_UPDATE": true 放在 PUT 请求主体中，以强制配置 API 服务器忽略项目 ID。

配置 API 服务 - 内容检索

可通过发出 HTTP GET 请求来从服务器中检索内容。在请求中指定的 URI 可将下列其中一个方面作为目标：

1. 在线文档 (例如 /config/v1/doc 或 /config/v1/doc/drivers)
2. 事件日志条目 (例如 /config/v1/event_log)
3. 事务处理日志条目 (例如 /config/v1/transaction_log)
4. 项目配置 (例如 /config/v1/project 或 /config/v1/project/channels/Channel1)

在确定项目配置目标时，REST 客户端可以指定应返回内容的类型。在此上下文中，“内容”一词是指有关集合或对象实例的数据的一个或多个类别。

默认情况下，使用可以标识集合的端点发出 GET 请求后，服务器将返回一个 JSON 数组，其中该数组包含集合中各个实例的值，而集合内的各个值是包含实例属性的 JSON 对象。

默认情况下，使用可以标识对象实例的端点发出 GET 请求后，服务器将返回一个包含该实例属性的 JSON 对象。

这些请求的默认行为可以进行改变，方法是指定一个或多个附加到 URL (例如 http://<主机名>:<端口>/config/v1/project?content=children) 中的“内容”查询参数。下表显示了可用的内容类型和针对每个端点类型的适用性：

内容类型	集合端点	对象实例端点
properties	是	是
property_definitions	no	是
property_states	no	是
type_definition	是	是
子项	是	是

下表显示了给定内容类型的 JSON 响应结构：

GET 请求 URI	JSON 响应结构

/config/v1/project?content=properties	{
	<property name>: <value>,
	<property name>: <value>,
	...
	}

/config/v1/project?content=property_definitions	[{<property definition>}, {<property definition>}, ...]
/config/v1/project?content=property_states	{ “allow”: { <property name>: true/false, <property name>: true/false,

GET 请求 URI	JSON 响应结构

	...
	},
	“enable”:
	{
	<property name>: true/false,
	<property name>: true/false,
	...
	}
	}

/config/v1/project?content=type_definition	{
	"name": <type name>,
	"collection": <collection name>,
	"namespace": <namespace name>,
	"can_create": true/false,
	"can_delete": true/false,
	"can_modify": true/false,
	"auto_generated": true/false,
	"requires_driver": true/false,
	"access_controlled": true/false,
	"child_collections": [<collection names>]
	}

/config/v1/project?content=children	{
	<collection name>: [
	{
	“name”: <object instance name>,
	“href”: <object instance uri>
	},
	...
	],
	<collection name>: [
	{
	“name”: <object instance name>,
	“href”: <object instance uri>
	},
	...
	],
	...
	}

可以在同一请求中指定多种内容类型，之间用逗号隔开。例如，http://<主机名>:<端口>/config/v1/project?content=children,type_definition。当指定多种类型时，JSON 响应将包含具有各个所请求内容类型成员的单个对象，例如：

```
{ “properties”: <属性响应结构>, “property_definitions”: <属性定义响应结构>,
“property_states”: <属性状态响应结构>, “type_definition”: <类型定义响应结构>,
“children”: <子项响应结构> }
```

类型定义

下表介绍了类型定义 JSON 对象的成员。

成员	类型	说明
名称	字符	对象类型名称。

成员	类型	说明
	串	
集合	字符串	集合名称。标识此类型对象所在的集合。此名称构成了可以使用 REST 接口寻址的有效端点。
namespace	字符串	执行对象类型的命名空间。由服务器执行的对象存在于 "servermain" 命名空间中。其他命名空间由可选组件 (例如驱动程序、插件和客户端接口) 定义。
can_create	布尔型	指示最终用户是否可以创建此类型的实例。例如，对于 "Project" 类型，此选项为 false，原因在于无法创建任何内容。
can_delete	布尔型	指示最终用户是否可以删除此类型的实例。同样，对于 "Project" 类型，无法删除实例。
can_modify	布尔型	指示最终用户是否可以修改此类型的实例。例如，服务器包含一些自动生成的对象，这些对象本身不具有任何可修改属性，而是仅用于创建子集合。
auto_ generated	布尔型	如果为 true，此类型的实例由服务器自动生成。通常，此类型对象的前三个成员必须定义为 "false"。
requires_ driver	布尔型	如果在未提供已安装驱动程序名称的情况下无法创建此类型的实例，则此选项为 true。
access_ controlled	布尔型	对于可以针对此类型实例执行的 CRUD 操作，如果服务器可提供组级别访问控制，则此选项为 true。*(请参阅 "用户管理器")*。
child_ collections	数组	集合名称数组，支持作为此类型对象下的子项。例如，如果某个类型的 "child_ collections" 中包括 "devices"，则该类型的对象实例支持将一个或多个 "Device" 实例作为子项。

属性定义

属性定义用于标识给定属性的特征，其中包括所支持的数据类型、适用范围、默认值等。属性定义对象的 JSON 结构定义如下：

成员	类型	说明
symbolic_ name	字符串	通过 <命名空间> 形式的规范名称标识属性。<属性名称>
display_ name	已本地化的字符串	属性显示在“服务器配置”属性编辑器中时所显示的名称。返回值将采用当前为服务器配置的语言。
display_ description	已本地化的字符串	属性显示在“服务器配置”属性编辑器中时所显示的说明。返回值将采用当前为服务器配置的语言。
read_only	布尔型	如果属性为信息性消息，并且在最初定义后不希望再进行更改，则此选项为 true。
type	字符串	确定属性值的数据类型。(请参阅下述"属性类型"。)
minimum_ value	数字或空 (适用于数字类型)	必须考虑属性最小值的有效性。如果为空，则没有最小值。
maximum_ value	数字或空 (适用于数字类型)	必须考虑属性最大值的有效性。如果为空，则没有最大值。
minimum_ length	数字 (仅适用于字符	字符串值可能具有的最小长度。0 表示无最小值。

成员	类型	说明
	串)	
maximum_length	数字 (仅适用于字符串)	字符串值可能具有的最大长度。-1 表示无最大值。
hints	字符串数组 (仅适用于字符串)	可分配给属性值的选项的数组。如果不存在提示，则不包括此成员。

enumeration	对象 (仅适用于枚举)	对于枚举属性，此对象将标识枚举可以具有的有效名称/值对。结构如下所示：
		{
		<name>: number,
		<name>: number,
		...
		}

allow	对象数组	针对可确定此属性是否相关的一个或多个其他属性，定义条件相关性。不允许的属性不会显示在“服务器配置”属性编辑器中。(请参阅下述“允许和启用条件”。)
enable	对象数组	针对可确定客户端更改时是否启用此属性的一个或多个其他属性，定义条件相关性。不允许启用的属性在“服务器配置”属性编辑器中呈灰显状态。(请参阅下述“允许和启用条件”。)

属性类型

下表介绍了属性定义在 "type" 成员中可能包含的不同值。“值类型”用于标识属性值应采用的 JSON 类型。

类型名称	值类型	说明
AllowDeny	布尔型	用于说明可显示包含 “Allow”=true 和 “Deny”= false 选项的下拉列表的属性。
EnableDisable	布尔型	用于说明可显示包含 “Enable”=true 和 “Disable”= false 选项的下拉列表的属性。
YesNo	布尔型	用于说明可显示包含 “Yes”=true 和 “No”= false 选项的下拉列表的属性。
String	字符串	普通字符串。此类型的属性将包括 minimum_length 和 maximum_length 指定符。
StringArray	数组	字符串数组。此类型的属性将包括应用于字符串本身的 minimum_length 和 maximum_length 说明符，而不是数组长度。
Password	字符串	包含密码的混淆字符串。当更改此类型的属性值时，希望使用纯文本密码。应该仅在使用安全连接的情况下更改密码值。
LocalFileSpec	字符串	本地 Windows 文件系统中的完全合格的文件规范。
UncFileSpec	字符串	网络位置中完全合格的文件规范。
LocalPathSpec	字符串	本地 Windows 文件系统中完全合格的路径规范。
UncPathSpec	字	网络位置中完全合格的路径规范。

类型名称	值类型	说明
	符串	
StringWithBrowser	字符串	描述包含字符串值的属性，其中字符串值通常可以从动态生成的字符串集合中进行选择。
Integer	数字	无符号 32 位整数值。
Hex	数字	旨在以十六进制符号进行显示/编辑的无符号 32 位整数值。
Octal	数字	旨在以八进制符号进行显示/编辑的无符号 32 位整数值。
SignedInteger	数字	有符号 32 位整数值。
Real4	数字	单精度浮点值。
Real8	数字	双精度浮点值。
Enumeration	数字	属性定义 "enumeration" 成员的可能数字值之一。
PropArray	对象	描述所包含成员均具有固定长度值数组的结构。
TimeOfDay	数字	此整数值包含自午夜开始计算的秒数，可用于定义特定时间。
Date	数字	此 Unix 时间值用于指定给定日期的午夜。
DateAndTime	数字	此 Unix 时间值用于指定给定日期的某一特定时间。
Blob	数组	此字节值数组表示不透明的数据集合。此类型的数据源自服务器并经过哈希处理，以防止修改。

允许和启用条件

对于包含允许和/或启用条件的定义，在 JSON 中的结构如下所示：

```
<条件>: [ { "depends_on": <属性名称> "operation": "==" or "!=" "value": <值> }, ...
]
```

每个条件标识属于依存对象的另一个属性，以及等于或不等于该属性值的判断方式。对于同一属性或几个不同属性，均可存在多个相关性。如果存在多个相关性，"operation" 始终相同。当存在多个相关性时，表达式的评估可确定条件状态为逻辑 "or" for "==" 以及 "and" for "!="。

当使用 "content=property_states" 时，返回的 JSON 将描述每个属性上述条件的评估结果 (如果存在)。

语言规范

服务器支持多种语言。它向客户端返回本地化的内容，所用语言为客户端已配置使用的语言。通过在 GET 请求标头中指定“Accept-Language”字段，客户端可以覆盖已配置的语言。

有关详细信息，请参阅 https://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html。

例如，如果服务器配置为英语，而客户端需要德语，则可以在请求标头中指定以下内容：“Accept-Language: de”。

注意：如果客户端指定了服务器不支持的语言，则使用当前配置的语言。

配置 API 服务 - 数据

配置 API 服务将从 REST 客户端接收标准 JSON 格式的请求。这些请求在服务器中使用，并细分为创建、读取、更新或删除命令。

有关属性、数据范围、端点映射方案的其他信息，以及配置 API 登录页面上每个端点的可接受操作，请参阅 http://localhost:57412/config/ (适用于默认配置)。

*默认情况下，登录页面中提供的文档采用 HTML 编码。*要获取 JSON 编码的文档，请包括一个以 "application/json" 开头的“接受”请求。

对象名称包含空格或 URL 格式所不允许的其他字符时，必须进行百分比编码才能由配置 API 正确解释。百分比编码包括将不允许的字符替换为其十六进制表示。例如，名为 "default object" 的对象，其百分比编码为 default%20object。以下字符在 URL 中不允许，必须进行编码：

空格	!	#	$	&	'	(	)	*	+	,	/	:	;	=	?	@	[	]
%20	%21	%23	%24	%26	%27	%28	%29	%2A	%2B	%2C	%2F	%3A	%3B	%3D	%3F	%40	%5B	%5D

服务器验证对象名称之前，将删除对象名称中的所有前导和尾随空格。这将导致服务器中的对象名称与用户通过配置 API 提供的对象名之间存在差异。用户可以在发送 PUT/POST 后对父对象发送 GET，以验证服务器中新的或已修改的对象名称是否与通过 API 发送的对象名称匹配。

可以通过配置 API 来更改电话簿中拨号条目的优先级顺序，方法是写入 PhonePriority 对象的 servermain.PHONEBOOK_PRIORITY 属性。此字符串属性是以逗号分隔的电话簿条目名称 (带引号) 的列表。

对端点 http://{本地主机:编号}/config/v1/project/channels/{信道名称} /phonebooks/phonebook/phonePriorities/ 例 GET 请求示返回：

```
{
"PROJECT_ID": 1270990535,
"common.ALLTYPES_DESCRIPTION": "",
"common.ALLTYPES_NAME": "PhonePriority",
 "servermain.PHONEBOOK_PRIORITY": "\"Phone3\",\"Phone1\",\"Phone2\""
}
```

对端点 http://{localhost:number}/config/v1/project/channels/{信道名称}/phonebooks/phonebook/phonePriorities/PhonePriority 的 PUT 请求示例可更改 servermain.PHONEBOOK_PRIORITY 属性：

```
{
"PROJECT_ID": 1270990535,
"servermain.PHONEBOOK_PRIORITY": "\"Phone1\",\"Phone2\",\"Phone3\""
}
```

百分比编码无法保证名称有效。要确定有效的名称值，请参阅适用于所创建特定对象的文档。

字符串属性中电话簿条目名称前后的引号必须使用反斜杠字符 (\") 转义。建议选择和配置调制解调器时使用服务器配置应用程序；而不使用 API。

如果用户同时打开了服务器配置，则以非管理员用户身份尝试使用 API 执行 POST/PUT/DELETE 会失败。错误为 401 状态码 (未授权)。一次仅一位用户可以写入运行时；如果没有足够的凭证，API 无法从服务器配置获取权限。

创建对象

可通过将 HTTP POST 请求发送至配置 API 来创建对象。创建新对象时，JSON 必须包括该对象所必需的属性 (例如，每个对象都必须具有名称)，但并非需要所有属性。未包括在 JSON 内的所有属性在创建时都设置为默认值。POST JSON 主体示例：

```
{
"<属性1_名称>": <值>,
"<属性2_名称>": <值>,
"<属性3_名称>": <值>
}
```

创建多个对象

通过在数组中包含 JSON 属性对象,可以将多个对象添加到给定集合中。POST JSON 正文示例:

```
[
        {
        "<属性1_名称>": <值>,
        "<属性2_名称>": <值>,
        "<属性3_名称>": <值>
        },
        {
        "<属性1_名称>": <值>,
        "<属性2_名称>": <值>,
        "<属性3_名称>": <值>
        }
]
```

当 POST 包含多个对象时,如果由于属性验证失败或其他错误而无法处理其中一个或多个对象,则返回 HTTP 状态代码 207 (多状态) 以及 JSON 对象数组 (其中包含请求中每个对象的状态)。例如,如果请求中包含两个对象,而第二个指定的名称与第一个相同:

```
[
        {
        "code": 201,
        "message": "Created"
        },
        {
        "code": 400,
        "message": "Validation failed on property common.ALLTYPES_NAME in object
definition at line 7: The name 'Channel1' is already used."
        }
]
```

创建具有子层次结构的对象

可以创建具有完整子对象层次结构的对象。为此,请将该层次结构包括在 POST 请求中,就像将其保存在 JSON 项目文件中时一样。例如,要创建带下属设备的通道,可以使用以下 JSON:

```
[
   {
        "common.ALLTYPES_NAME": "Channel1",
        "servermain.MULTIPLE_TYPES_DEVICE_DRIVER": "Simulator",
        "devices": [
                {
                "common.ALLTYPES_NAME": "Device1",
                "servermain.MULTIPLE_TYPES_DEVICE_DRIVER": "Simulator",
                "servermain.DEVICE_MODEL": 0
                }
        ]
   }
]
```

读取对象

可通过将 HTTP GET 请求发送至配置 API 来读取对象。每次 GET 请求都会返回所有对象属性，且每个对象包括一个 Project_ID。Project_ID 属性用来跟踪配置中的更改，且在配置 API 或服务器配置客户端发生任何更改时进行更新。此属性应在所有 PUT 请求中保存和使用，以防止操作过期数据。响应主体示例：

```
{
"<属性1_名称>": <值>,
"<属性2_名称>": <值>,
"PROJECT_ID": 12345678
}
```

另请参阅：内容检索

编辑对象

可通过将 HTTP PUT 请求发送至配置 API 来编辑对象。PUT 请求需要 JSON 主体中的 Project_ID 或 Force_Update 属性。将 Force_Update 设置为 True，可忽略 Project_ID 验证。PUT 主体示例：

```
{
"<属性1_名称>": <值>,
"<属性2_名称>": <值>,
"PROJECT_ID": 12345678,
"FORCE_UPDATE": true
}
```

通常，当 PUT 请求成功并且所有属性都被成功分配时，没有响应正文返回到客户端，只有一个 200 状态代码来表示成功。有时，PUT 请求中的属性可能未被服务器运行时分配给对象实例。此情况下将生成响应正文，如下所示：

```
{,
"not_applied":,
        {, "servermain.CHANNEL_UNIQUE_ID": 2466304381
        },
"code": 200,
"message": "Not all properties were applied. 这可能是因为已启用的客户端引用或属性被禁止/禁用/只读。
}
```

响应内容指明哪些属性未应用于对象实例，并包含各个实际使用的值。无法应用属性值的原因有以下几种：

- 此属性只读，不能更改。
- 对象的客户端引用限制更新此属性。
- 根据相关条件中其他属性的值，不允许此属性。
- 根据相关条件中其他属性的值，不启用此属性。
- 值发生转变 (例如: 四舍五入或截断)。

删除对象

可通过将 HTTP DELETE 请求发送至配置 API 来删除对象。配置 API 不允许使用一个请求删除相同级别的多个项目 (例如，删除一个信道内的所有设备)，但可以删除整个树 (例如，删除设备可删除其所有子标记)。

错误

所有配置 API 服务请求都以 JSON 格式返回错误。示例：

```
{
"code": 400,
"message": "Invalid property: 'NAME'."
}
```

另请参阅：故障排除

配置 API 服务 - 服务

如果除了标准 CRUD (创建、检索、更新、删除) 操作之外，还可以对对象调用其他操作，则对象可提供服务。服务提供了一个异步编程接口，远程客户端可通过该接口触发和监控这些操作。这些服务可在其操作的对象下的“服务”集合中找到。例如，针对项目的“项目加载”服务位于 /config/v1/project/services/ProjectLoad 端点。由于任何对象都可以提供服务，因此可以先查询服务集合是否存在，然后查询该集合中的服务。

服务体系结构

服务旨在与其操作的对象进行无状态交互。服务由两部分组成：服务和作业。作业异步执行工作，并支持客户端监视作业的完成情况以及操作期间出现的任何错误。作业完成后，服务器会按计划自动将其删除。客户端无需在作业完成后清理作业。

服务

服务是调用操作的接口。调用服务时可指定的所有参数以属性呈现。要查看可用参数，请对服务端点执行 HTTP GET。包括服务的名称和说明在内，所有属性都是可在调用服务时包含的参数。这些参数的一部分或全部可能是必需参数，具体取决于服务。

要调用服务，请对服务端点执行 HTTP PUT 请求，并在请求正文中指定所需参数。服务可能会限制并发调用的总数。当达到并发调用最大数目时，请求将被拒绝，请求响应为“HTTP 429 Too Many Requests”。如果未达到限制数，服务器响应为“HTTP 202 Accepted”，且响应正文包括指向新创建的作业的链接。

成功 PUT 响应示例：

```
{
"code": 202,
"message": "Accepted",
"href": "/config/v1/project/services/ProjectLoad/jobs/job1"
}
```

繁忙 PUT 响应示例：

```
{
"code": 429,
"message": "The server is busy. Retry the operation at a later time."
}
```

作业

作业表示服务器接受的特定请求。要检查作业状态，请对作业端点执行 HTTP GET 请求。**servermain.JOB_COMPLETE** 属性将作业的当前状态表示为布尔值。在作业完成执行之前，此属性的值保持为 false。如果由于某种原因无法执行作业，服务器会通过 **servermain.JOB_STATUS_MSG** 属性向客户端提供相应的错误消息。

作业清理

服务器可在指定时间内自动删除作业。默认情况下，当作业完成时，客户端可以在 30 秒内与之交互。如果需要更长时间，或者连接速度较慢，客户端可以在调用服务时使用 **servermain.JOB_TIME_TO_LIVE_SECOND** 参数，将生存时间最多提高至 5 分钟。每个作业都有自己的生存时间，在创建之后不能更改。由于不允许从客户端手动删除服务器上的作业，因此最好在不影响客户端获取作业信息的情况下选择最短的生存时间。

服务交互示例

以下详细说明客户端与服务交互时的过程。只有当客户端不知道所需的参数和/或服务具有什么参数时，步骤 1 才是必需的。如果这些是已知的，客户端可以直接从步骤 2 开始。

我们将 /config/v1/project/services/ProjectLoad 端点的“项目加载”服务用作示例。

1. 对服务执行 GET 请求以检索可用参数。

   操作：

   ```
   GET /config/v1/project/services/ProjectLoad
   ```

   响应：

   ```
   { "PROJECT_ID": 984022419, "common.ALLTYPES_NAME": "ProjectLoad",
   "servermain.JOB_TIME_TO_LIVE_SECONDS": 30, "servermain.PROJECT_FILENAME": ""
   "servermain.PROJECT_PASSWORD": "" }
   ```

2. 使用 PUT 响应中的链接对作业执行 GET。

   操作：

   ```
   GET /config/v1/project/services/ProjectLoad/jobs/job1
   ```

   响应：

   ```
   {
   "PROJECT_ID": 1366849387,
   "common.ALLTYPES_NAME": "job1",
   "servermain.JOB_COMPLETE":false,
   "servermain.JOB_STATUS_MSG": "",
   "servermain.PROJECT_FILENAME": "C:\\location\\of\\file.opf"
   }
   ```

3. 使用正确的参数对服务执行 PUT 请求。所需的参数因服务而异。请注意，**servermain.JOB_TIME_TO_LIVE_SECONDS** 是可选的，在本示例中已省略。如果所加载的项目未加密，则 **servermain.PROJECT_PASSWORD** 为可选并被省略。

   操作：

   ```
   PUT /config/v1/project/services/ProjectLoad
   ```

   ```
   {
   "servermain.PROJECT_FILENAME": "C:\\location\\of\\file.opf"
   }
   ```

   响应：

   ```
   {
   "code": 202,
   "message": "Accepted",
   "href": "/config/v1/project/services/ProjectLoad/jobs/job1"
   }
   ```

4. 将加密的项目另存为 .SOPF。

   操作：

   ```
   PUT /config/v1/project/
           {
                   "force_update": true,
                   "servermain.PROJECT_ENCRYPTION_ENABLE": true
           }
   ```

   响应：

```
202 - OK
```

操作：

```
PUT /config/v1/project/Services/ProjectSave/
        {
                "common.ALLTYPES_NAME": "ProjectSave",
                "servermain.PROJECT_FILENAME": "test/file.sopf",
                "servermain.PROJECT_PASSWORD": "password123"
        }
```

响应：

```
{
"code": 202,
"message": "Accepted",
"href": "/config/v1/project/services/ProjectSave/jobs/job1"
}
```

5. 轮询作业，直到 **servermain.JOB_COMPLETE** 为 true。当它为 true 时，作业即完成执行。

操作：

```
GET /config/v1/project/services/ProjectLoad/jobs/job1
```

响应：

```
{
"PROJECT_ID": 1366849387,
"common.ALLTYPES_NAME": "job1",
"servermain.JOB_COMPLETE": true,
"servermain.JOB_STATUS_MSG": "",
"servermain.PROJECT_FILENAME": "C:\\location\\of\\file.opf"
}
```

配置 API 服务 - 响应码

REST 请求可能返回下列响应码之一。响应的正文可能包含具体的错误消息，以帮助判断错误原因和可能的解决方案：

- HTTP/1.1 200 正常
- HTTP/1.1 201 已创建
- HTTP/1.1 202 已接受
- HTTP/1.1 207 多状态
- HTTP/1.1 400 错误请求
- HTTP/1.1 401 未授权
- HTTP/1.1 403 禁止
- HTTP/1.1 404 未找到
- HTTP/1.1 429 Too Many Requests
- HTTP/1.1 500 内部服务器错误
- HTTP/1.1 503 服务器运行时不可用
- HTTP/1.1 504 网关超时连接
- HTTP/1.1 520 未知错误

请参阅配置 API 服务事件日志消息

iFIX 信号调节选项

以下信号调节选项可在 iFIX 数据库管理器中获得：

3BCD
4BCD
8AL
8BN
12AL
12BN
13AL
13BN
14AL
14BN
15AL
15BN
20P
TNON

注意：通过服务器仅可获得针对静态标记的线性和对数标定。有关详细信息，请参阅标记属性 - 缩放和静态标记 (用户定义)。

3BCD 信号调节

说明	3 位二进制编码的十进制 (BCD) 值
输入范围	0-999
换算	将 3 位二进制编码的十进制值换算成数据库块的 EGU 范围。
读取算法	读取 3 位 BCD 寄存器。在缩放该值之前，先将 Raw_value 分为三个半字节 (4 位)。会检查每个半字节的值是否大于 9 (十六进制 A-F)。如果找到 A 到 F 之间的十六进制值，则会生成范围警报，表示该值不在 BCD 范围内。否则，将使用以下算法对该值进行缩放： Result=((Raw_value/999) * Span_egu) + Lo_egu。
读取算法变量	Lo_egu - 数据库块的低工程值。 Span_egu - 工程值的跨度。 Raw_value - 存储在字段设备寄存器中的值。 Result - 存储在数据库块中的缩放值。
写入算法	使用以下算法写入 3 位 BCD 寄存器： Result=(((InputData-Lo_egu) / Span_egu) * 999) + .5。
写入算法变量	Lo_egu - 低工程值。 Span_egu - 工程值的跨度。 InputData - 数据库块的当前值。 Result - 发送到处理硬件的值。

4BCD 信号调节

说明	4 位二进制编码的十进制 (BCD) 值
输入范围	0-9999
换算	将 4 位二进制编码的十进制值缩放至数据库块的 EGU 范围。
读取算法	读取 4 位 BCD 寄存器。在缩放该值之前，先将 Raw_value 分为四个半字节 (4 位)。会检查每个半字节的值是否大于 9 (十六进制 A-F)。如果找到 A 到 F 之间的十六进制值，则会生成范围警报，表示该值不在 BCD 范围内。否则，将使用以下算法对该值进行缩放： Result=((Raw_value/9999) * Span_egu) + Lo_egu。
读取算法变量	Lo_egu - 数据库块的低工程值。 Span_egu - 工程值的跨度。 Raw_value - 存储在字段设备寄存器中的值。 Result - 存储在数据库块中的缩放值。
写入算法	使用以下算法写入 4 位 BCD 寄存器： Result=(((InputData-Lo_egu) / Span_egu) * 9999) + .5。
写入算法变量	Lo_egu - 低工程值。 Span_egu - 工程值的跨度。 InputData - 数据库块的当前值。 Result - 发送到处理硬件的值。

8AL 信号调节

说明	8 位二进制数字
输入范围	0-255
换算	将 8 位二进制值缩放至数据库块的 EGU 范围。
读取算法	使用同 8BN 一样的算法读取 16 位寄存器，并返回一个状态，用于指示值是否超出范围或处于警报状态，或者返回“确定”。 Result =((Raw_value/255) * Span_egu) + Lo_egu。
读取算法变量	Lo_egu - 数据库块的低工程值。 Span_egu - 工程值的跨度。 Raw_value - 存储在字段设备寄存器中的值。 Result - 存储在数据库块中的缩放值。
写入算法	使用同 8BN 一样的算法写入 16 位寄存器，并返回一个状态，用于指示值是否超出范围或处于警报状态，或者返回“确定”。 Result=(((InputData-Lo_egu)/Span_egu) * 255) + .5。
写入算法变量	Lo_egu - 低工程值。 Span_egu - 工程值的跨度。 InputData - 数据库块的当前值。 Result - 发送到处理硬件的值。

8BN 信号调节

说明	8 位二进制数字
输入范围	0-255
换算	将 8 位二进制值换算为数据库块的 EGU 范围。忽略最高有效位。
读取算法	使用以下算法读取 16 位寄存器： Result =((Raw_value/255) * Span_egu) + Lo_egu。
读取算法变量	Lo_egu - 数据库块的低工程值。 Span_egu - 工程值的跨度。 Raw_value - 存储在字段设备寄存器中的值。 Result - 存储在数据库块中的缩放值。
写入算法	使用以下算法写入 8 位 BCD 寄存器： Result =(((InputData-Lo_egu)/Span_egu) * 255) + .5。
写入算法变量	Lo_egu - 低工程值。 Span_egu - 工程值的跨度。 InputData - 数据库块的当前值。 Result - 发送到处理硬件的值。

12AL 信号调节

说明	12 位二进制数字
输入范围	0-4095
换算	将 12 位二进制值换算至数据库块的 EGU 范围。
读取算法	使用同 12BN 一样的算法读取 16 位寄存器，并返回一个状态，用于指示值是否超出范围或处于警报状态，或者返回“确定”。 Result=((Raw_value/4095) * Span_egu) + Lo_egu。
读取算法变量	Lo_egu - 数据库块的低工程值。 Span_egu - 工程值的跨度。 Raw_value - 存储在字段设备寄存器中的值。 Result - 存储在数据库块中的缩放值。
写入算法	使用同 12BN 一样的算法写入 16 位寄存器，并返回一个状态，用于指示值是否超出范围或处于警报状态，或者返回“确定”。 Result=(((InputData-Lo_egu)/Span_egu) * 4095) + .5。
写入算法变量	Lo_egu - 低工程值。 Span_egu - 工程值的跨度。 InputData - 数据库块的当前值。 Result - 发送到处理硬件的值。

12BN 信号调节

说明	12 位二进制数字
输入范围	0-4095
换算	将 12 位二进制值换算至数据库块的 EGU 范围。忽略最高有效半字节 (4 位)。超出范围值将被视为 12 位值。例如，4096 因忽略四位最高有效位而被视为 0。
读取算法	使用以下算法读取 16 位寄存器： Result =((Raw_value/4095) * Span_egu) + Lo_egu。
读取算法变量	Lo_egu - 数据库块的低工程值。 Span_egu - 工程值的跨度。 Raw_value - 存储在字段设备寄存器中的值。 Result - 存储在数据库块中的缩放值。
写入算法	使用以下算法写入 16 位寄存器： Result=(((InputData-Lo_egu)/Span_egu) * 4095) + .5。
写入算法变量	Lo_egu - 低工程值。 Span_egu - 工程值的跨度。 InputData - 数据库块的当前值。 Result - 发送到处理硬件的值。

13AL 信号调节

说明	13 位二进制数字
输入范围	0-8191
换算	将 13 位二进制值换算至数据库块的 EGU 范围。
读取算法	使用同 13BN 一样的算法读取 16 位寄存器，并返回一个状态，用于指示值是否超出范围或处于警报状态，或者返回“确定”。 Result=((Raw_value/8191) * Span_egu) + Lo_egu。
读取算法变量	Lo_egu - 数据库块的低工程值。 Span_egu - 工程值的跨度。 Raw_value - 存储在字段设备寄存器中的值。 Result - 存储在数据库块中的缩放值。
写入算法	使用同 13BN 一样的算法写入 16 位寄存器，并返回一个状态，用于指示值是否超出范围或处于警报状态，或者返回“确定”。 Result=(((InputData-Lo_egu)/Span_egu) * 8191) + .5。
写入算法变量	Lo_egu - 低工程值。

说明	13 位二进制数字
	Span_egu - 工程值的跨度。 InputData - 数据库块的当前值。 Result - 发送到处理硬件的值。

13BN 信号调节

说明	13 位二进制数字
输入范围	0-8191
换算	将 13 位二进制值换算至数据库块的 EGU 范围。忽略 3 位最高有效位。
读取算法	使用以下算法读取 16 位寄存器： Result=((Raw_value/8191) * Span_egu) + Lo_egu。
读取算法变量	Lo_egu - 数据库块的低工程值。 Span_egu - 工程值的跨度。 Raw_value - 存储在字段设备寄存器中的值。 Result - 存储在数据库块中的缩放值。
写入算法	使用以下算法写入 16 位寄存器： Result =(((InputData-Lo_egu)/Span_egu) * 8191) + .5。
写入算法变量	Lo_egu - 低工程值。 Span_egu - 工程值的跨度。 InputData - 数据库块的当前值。 Result - 发送到处理硬件的值。

14AL 信号调节

说明	14 位二进制数字
输入范围	0-16383
换算	将 14 位二进制值换算至数据库块的 EGU 范围。
读取算法	使用同 14BN 一样的算法读取 16 位寄存器，并返回一个状态，用于指示值是否超出范围或处于警报状态，或者返回“确定”。 Result=((Raw_value/16383) * Span_egu) + Lo_egu.
读取算法变量	Lo_egu - 数据库块的低工程值。 Span_egu - 工程值的跨度。 Raw_value - 存储在字段设备寄存器中的值。 Result - 存储在数据库块中的缩放值。
写入算法	使用同 14BN 一样的算法写入 16 位寄存器，并返回一个状态，用于指示值是否超出范围或处于警报状态，或者返回“确定”。 Result=(((InputData-Lo_egu)/Span_egu) * 16383) + .5.
写入算法变量	Lo_egu - 低工程值。 Span_egu - 工程值的跨度。 InputData - 数据库块的当前值。 Result - 发送到处理硬件的值。

14BN 信号调节

说明	14 位二进制数字
输入范围	0-16383
换算	将 14 位二进制值换算至数据库块的 EGU 范围。忽略 2 位最高有效位。
读取算法	使用以下算法读取 16 位寄存器： Result=((Raw_value/16383) * Span_egu) + Lo_egu.
读取算法变量	Lo_egu - 数据库块的低工程值。 Span_egu - 工程值的跨度。 Raw_value - 存储在字段设备寄存器中的值。 Result - 存储在数据库块中的缩放值。
写入算法	使用以下算法写入 16 位寄存器： Result=(((InputData-Lo_egu)/Span_egu) * 16383) + .5.

说明	14 位二进制数字
写入算法变量	Lo_egu - 低工程值。 Span_egu - 工程值的跨度。 InputData - 数据库块的当前值。 Result - 发送到处理硬件的值。

15AL 信号调节

说明	15 位二进制数字
输入范围	0-32767
换算	将 15 位二进制值换算至数据库块的 EGU 范围。
读取算法	使用同 15BN 一样的算法读取具有报警功能的 16 位寄存器，并返回一个状态，用于指示值是否超出范围或处于警报状态，或者返回“确定”。 Result=((Raw_value/32767) * Span_egu) + Lo_egu.
读取算法变量	Lo_egu - 数据库块的低工程值。 Span_egu - 工程值的跨度。 Raw_value - 存储在字段设备寄存器中的值。 Result - 存储在数据库块中的缩放值。
写入算法	使用同 15BN 一样的算法写入具有报警功能的 16 位寄存器，并返回一个状态，用于指示值是否超出范围或处于警报状态，或者返回“确定”。 Result=(((InputData-Lo_egu)/Span_egu) * 32767) + .5.
写入算法变量	Lo_egu - 低工程值。 Span_egu - 工程值的跨度。 InputData - 数据库块的当前值。 Result - 发送到处理硬件的值。

15BN 信号调节

说明	15 位二进制数字
输入范围	0-32767
换算	将 15 位二进制值换算至数据库块的 EGU 范围。忽略最高有效位。
读取算法	使用以下算法读取 16 位寄存器： Result =((Raw_value/32767) * Span_egu) + Lo_egu。
读取算法变量	Lo_egu - 数据库块的低工程值。 Span_egu - 工程值的跨度。 Raw_value - 存储在字段设备寄存器中的值。 Result - 存储在数据库块中的缩放值。
写入算法	使用以下算法写入 16 位寄存器： Result =(((InputData-Lo_egu)/Span_egu) * 32767) + .5。
写入算法变量	Lo_egu - 低工程值。 Span_egu - 工程值的跨度。 InputData - 数据库块的当前值。 Result - 发送到处理硬件的值。

20P 信号调节

说明	6400 - 32000 钳位
输入范围	6400 - 32000
换算	将二进制值缩放至数据库块的 EGU 范围。使值处于 6400-32000 范围内。
读取算法	使用以下算法读取 16 位寄存器： Result =(((Raw_value-6400)/25600) * Span_egu) + Lo_egu。
读取算法变量	Lo_egu - 数据库块的低工程值。 Span_egu - 工程值的跨度。 Raw_value - 存储在字段设备寄存器中的值。 Result - 存储在数据库块中的缩放值。
写入算法	使用以下算法写入 16 位寄存器：

说明	6400 - 32000 钳位
	Result =(((InputData-Lo_egu)/Span_egu) * 25600) + 6400.5。
写入算法 变量	Lo_egu - 低工程值。 Span_egu - 工程值的跨度。 InputData - 数据库块的当前值。 Result - 发送到处理硬件的值。

TNON 信号调节

说明	0 - 32000 钳位
输入范围	0 - 32000
换算	将二进制值缩放至数据库块的 EGU 范围。使值处于 0-32000 范围内。
读取算法	使用以下算法读取 16 位寄存器： Result =((Raw_value/32000) * Span_egu) + Lo_egu。
读取算法 变量	Lo_egu - 数据库块的低工程值。 Span_egu - 工程值的跨度。 Raw_value - 存储在字段设备寄存器中的值。 Result - 存储在数据库块中的缩放值。
写入算法	使用以下算法写入 16 位寄存器： Result =(((InputData-Lo_egu)/Span_egu) * 32000) + .5。
写入算法变量	Lo_egu - 低工程值。 Span_egu - 工程值的跨度。 InputData - 数据库块的当前值。 Result - 发送到处理硬件的值。

iFIX 应用程序的项目启动

服务器 iFIX 接口功能得到增强，可为 iFIX 用户提供更好的启动性能。此增强功能适用于使用“模拟输出”(AO)、“数字输出”(DO) 的 iFIX 应用程序和/或先前在启动时初始化错误的“警报值”。服务器会为包含 iFIX 客户端可访问的所有项的默认服务器项目维护特殊的 iFIX 配置文件。此配置文件用于在 iFIX 请求项数据前自动启动扫描项功能。因此，在 iFIX请求时仅请求一次的数据更新 (如 AO/DO) 已具有初始值。有关在现有 iFIX 项目中此功能的使用信息，请参阅以下说明。

1. 首先，从iFIX 数据库管理器中导出PDB 数据库。
2. 重新导入已导出的文件，使数据库中的每个项再次通过服务器进行验证。
3. 在**“确认标记替换”**消息框中，选择**“替换所有标记”**。

   注意：新配置文件的创建位置即包含 "default_FIX.ini" 名称的默认服务器项目文件所在的文件夹。

4. 如果读取项目中所有项的初始值所需时间较长，则可能需要延迟 SAC 处理的启动。执行此操作可确保服务器有足够的时间在 iFIX 客户端向服务器请求数据前检索所有初始更新。有关特定 iFIX 版本的详细信息，请参阅 iFIX 文档。
5. 重新启动 iFIX 应用程序和服务器以使更改生效。

注意：对于新项目 (或向现有 iFIX 数据库中添加其他项时)，用户无需执行上述步骤。向数据库中添加项时，服务器会对该项进行验证。如果该项有效，服务器会将其添加到配置文件中。

存储并转发服务

“存储并转发服务”允许不同的服务器组件将数据存储在本地磁盘上一段时间。该服务安装了需要存储并转发功能的组件。“存储并转发服务”会根据支持存储并转发的功能自动启动和停止。

另请参阅：

ThingWorx 项目属性
“存储并转发”配置设置
存储并转发系统标记
ThingWorx 访问权限

内置诊断

发生通信问题时，用户可以利用 OPC 和信道诊断来确定问题原因。这些视图可提供服务器级别和驱动程序级别诊断。由于诊断可能会影响性能，因此用户应仅在调试或故障排除时利用诊断。有关详细信息，请从下表中选择一个链接。

OPC 诊断查看器
信道诊断

OPC 诊断查看器

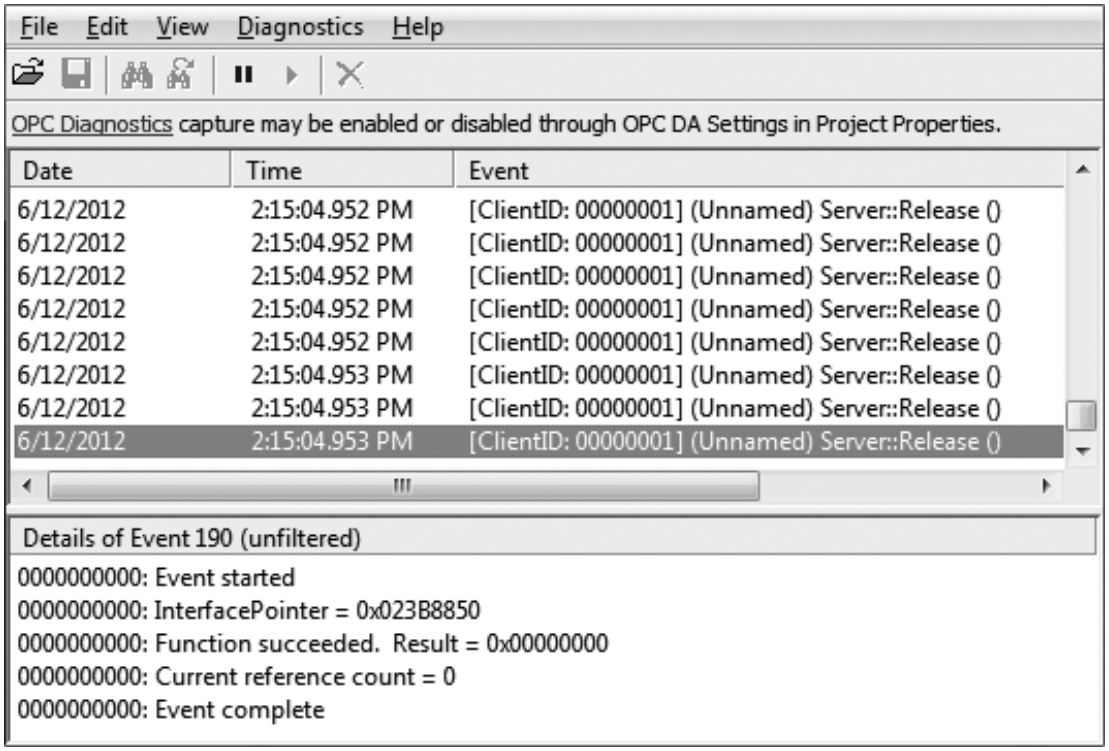
OPC 诊断查看器同时提供 OPC 客户端与服务器之间发生的 OPC 事件的实时和历史视图。事件是客户端对服务器做出的方法调用，或服务器对客户端做出的回调。

访问 OPC 诊断查看器

OPC 诊断查看器独立于主服务器配置窗口。要访问 OPC 诊断查看器，请单击**“查看”|“OPC 诊断”**。

注意：当捕获被禁用时，虽然可以访问查看器，但在其启用前没有任何诊断。

有关启用 OPC 诊断的信息，请参阅项目属性 - OPC DA、项目属性 - OPC UA 设置。



有关日志设置属性的信息，请参阅设置 - 事件日志。

实时数据模式

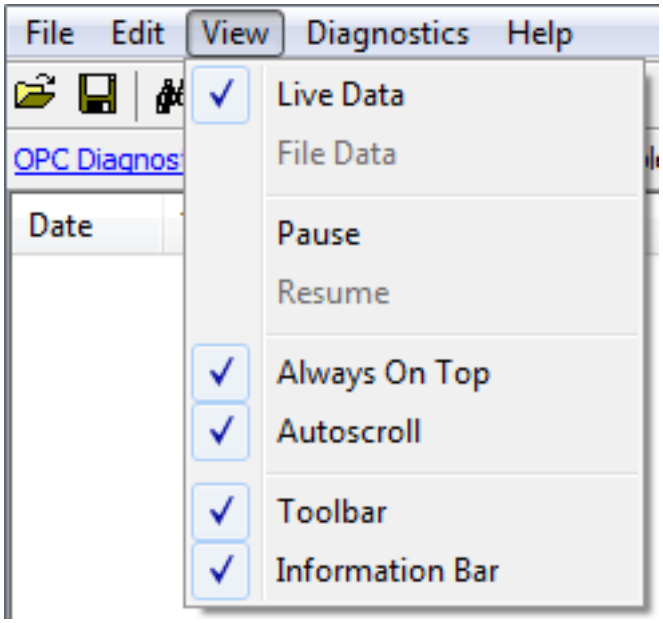
OPC 诊断查看器在“实时数据”模式下打开，可显示“事件日志”中当前可用的持久 OPC 诊断数据。查看器会实时更新。要暂停显示，请单击**“查看”|“暂停”**或选择**“暂停”**图标。虽然可以继续捕获数据，但显示不会更新。

要保存 OPC 诊断文件，请单击**“文件”|“另存为”**，然后选择 **OPC 诊断文件 (*.opcdiag)**。

文件数据模式

OPC 诊断查看器可以打开并显示已保存的 OPC 诊断文件。打开已保存的文件时，查看器会切换到“文件数据模式”，并显示加载文件中的名称和数据。用户可以通过“视图”菜单切换模式。关闭文件后，视图会切换到“实时数据”，且“文件数据”视图不可用，直到加载另一个文件之后才可用。

“查看”菜单

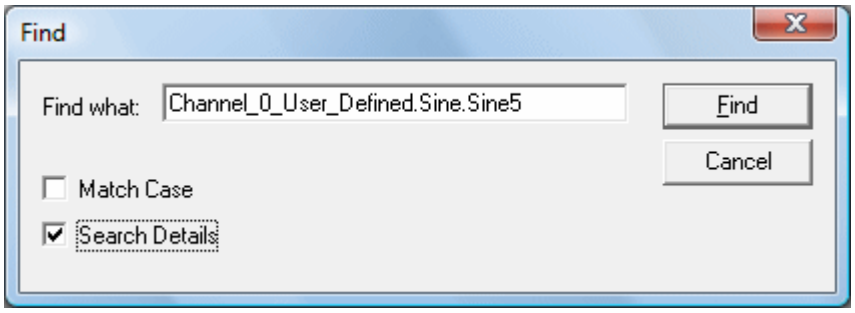


选项说明如下：

- **"实时数据"**启用后，此选项将显示"事件日志"中当前可用的任何持久 OPC 诊断数据。默认设置为启用状态。有关详细信息，请参阅实时数据模式。
- **"文件数据"**启用后，此选项将显示保存的 OPC 诊断文件中的数据。默认设置为禁用状态。有关详细信息，请参阅文件数据模式。
- **"始终置顶"**启用后，此选项将强制 OPC 诊断窗口始终位于所有其他应用程序窗口的顶部。默认设置为启用状态。
- **"自动滚动"**启用后，此选项将在收到新事件时滚动显示以确保最新事件可见。当用户手动选择事件时(或通过"查找"/"查找下一个"进行选择时)，它将关闭。
- **"工具栏"**启用后，此选项将显示由图标组成的工具栏，可用于快速访问"文件"、"编辑"和"视图"菜单中可用的选项。默认设置为启用状态。
- **"信息栏"**启用后，此选项将显示 OPC 诊断数据上方的信息栏。默认设置为启用状态。

"查找"

此对话框用于在诊断视图中搜索在客户端和服务器之间传送的密钥信息。例如，此搜索功能可用于查找特定项 ID 或组名称的所有操作。



属性说明如下：

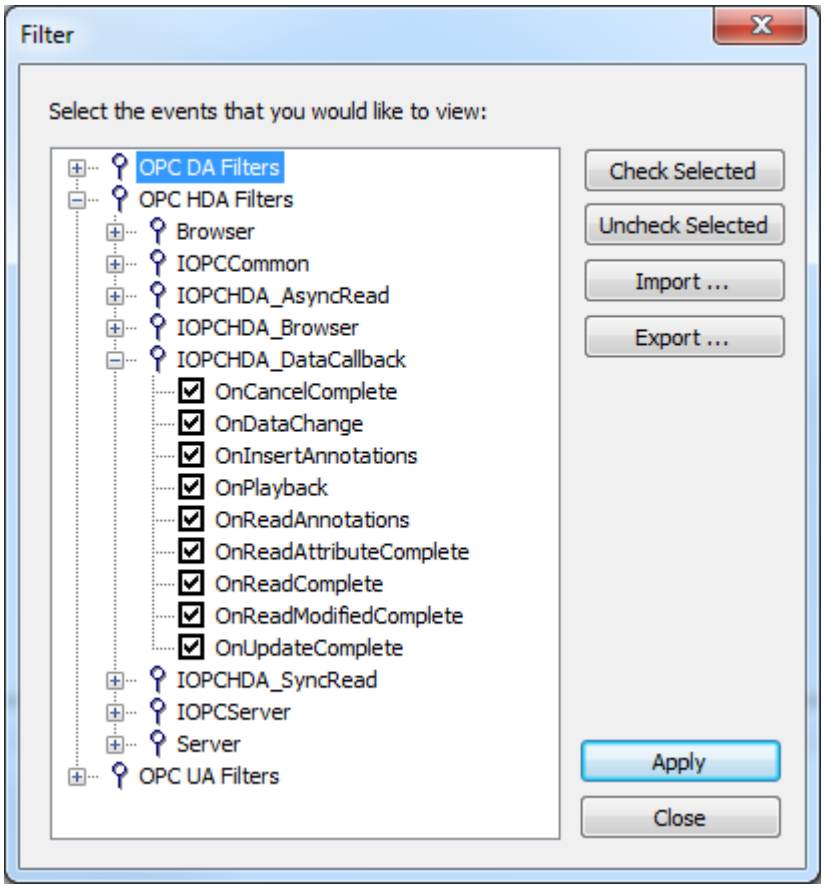
- **"查找内容"**此字段用于指定搜索条件。
- **"区分大小写"**启用后，搜索条件将区分大小写。
- **"搜索详细信息"**启用后，搜索条件将包含详细信息。

注意：当找到带有指定文本的事件或详细信息时，包含该文本的行将突出显示。要执行"查找下一个"操作(以及查找指定文本的下一个匹配项)，请按 "F3"。当找到最后一个匹配项后，消息框会指示这种情况。通过按 "Ctrl + F"，用户可以随时更改搜索条件。

"筛选"

此对话框会指定哪些事件在 OPC 诊断查看器中可见。例如，大多数客户端将对服务器进行连续的 GetStatus 调用，以确定服务器是否仍然可用。通过筛选此事件，用户即可检查诊断数据。所应用的筛选是针对视图的，而非针对捕获。无论筛选器设置如何，都会捕获所有事件类型。此外，由于可在对话框处于打开状态时应用筛选器，因此可以独立地更改并应用设置。无需关闭并重新打开对话框即可进行更改。

注意：服务器支持的每种 OPC 数据访问 1.0、2.0 和 3.0 接口的每种方法 (如 "IOPCCommon" 或 "GetErrorString") 均可用作筛选器。



选项说明如下：

- **"选中所选内容"**：单击此按钮后，会启用选定项下的所有事件进行查看。默认情况下，所有接口的所有方法都会被选中。
  - *有关详细信息，请参阅 OPC DA 事件和 OPC UA 服务。*
- **"取消选中所选内容"**单击此按钮后，会启用选定项下的所有事件类型和方法。
- **"导入"**单击此按钮后，用户可以选择要导入到筛选器中的 INI 文件。
- **"导出"**单击此按钮后，用户可以将筛选器导出为 INI 文件。

注意：

1. 由于筛选器设置在 OPC 诊断查看器关闭时会保持不变，因此用户可以稍后重新打开并查看 OPC 诊断文件。在"文件数据模式"下打开的文件可能已经过筛选。从 OPC 诊断查看器保存文件时，将仅保存在应用筛选器后所显示的事件。如果需要一个未筛选的数据文件，用户必须在保存文件之前关闭筛选。

2. 当捕获诊断信息时，服务器的性能会受到影响，因为它是发生于客户端与服务器通信之间的附加处理层。此外，在"扩展数据存储持久模式"下登录 OPC 诊断可能会占用大量的磁盘空间。Windows 事件查看器会报告任何相关错误。有关持久模式的信息，请参阅设置 - 事件日志。

OPC DA 事件

有关特定 OPC 诊断事件的详细信息，请从下表中选择一个链接。

IClassFactory
Server
IOPCCommon
IOPCServer
IConnectionPointContainer (服务器)
IConnectionPoint (服务器)
IOPCBrowse
IOPCBrowseServerAddressSpace
IOPCItemProperties
IOPCItemIO
Group
IOPCGroupStateMgt
IOPCGroupStateMgt2
IOPCItemMgt
IOPCItemDeadbandMgt
IOPCItemSamplingMgt
IOPCSyncIO
IOPCSyncIO2
IOPCAsyncIO
IDataObject
IAdviseSink
IAsyncIO2
IAsyncIO3
IConnectionPointContainer (组)
IConnectionPoint (组)
IOPCDataCallback
IEnumOPCItemAttributes

IClassFactory

IClassFactory 接口包含多种可用于处理整个对象类的方法。它是针对特定类对象的类对象实现的，并由 CLSID 标识。

- **QueryInterface**：客户端可以通过调用 IConnectionPointContainer 的 QueryInterface 来询问对象是否支持一些出站接口。如果对象通过递回有效指针做出的回答为“是”，则客户端将获知此对象可以尝试建立连接。
- **AddRef**：增加对象上接口的参考计数。给定对象上接口指针的每个新副本都应该调用此参数。
- **Release**：将接口的参考计数减少 1。
- **CreateInstance**：创建未初始化的对象。
- **LockServer**：允许在类对象客户端调用时快速创建实例，以保持服务器在内存中处于开启状态。

服务器

客户端调用 CoCreateInstance 来创建服务器对象和初始接口。

- **QueryInterface**：客户端可以通过调用 IConnectionPointContainer 的 QueryInterface 来询问对象是否支持一些出站接口。如果对象通过递回有效指针做出的回答为“是”，则客户端将获知此对象可以尝试建立连接。
- **AddRef**：增加对象上接口的参考计数。给定对象上接口指针的每个新副本都应该调用此参数。
- **Release**：将接口的参考计数减少 1。

IOPCCommon

此接口可供所有 OPC 服务器类型 (DataAccess、Alarm&Event、历史数据等) 使用。它可用于设置和查询区域设置 ID，此 ID 将对特定的客户端/服务器会话有效。一个客户端的操作不会影响其他客户端。

- **GetErrorString**：返回服务器特定错误代码的错误字符串。预期的行为是这还将包括 Win32 错误 (如 RPC 错误) 处理。
- **GetLocaleID**：为此服务器/客户端会话返回默认的区域设置 ID。
- **QueryAvailableLocaleIDs**：为此服务器/客户端会话返回可用的区域设置 ID。
- **SetClientName**：允许客户端选择向服务器注册客户端名称。这主要用于调试目的。推荐的行为是用户在此设置节点名称和 EXE 名称。
- **SetLocaleID**：为此服务器/客户端会话设置默认的区域设置 ID。此区域设置 ID 由此接口上的 GetErrorString 方法使用。服务器的默认值应为 LOCALE_SYSTEM_DEFAULT。

IOPCServer

这是 OPC 服务器的主接口。按照本规范的“安装和注册”一章中所指定的，向操作系统注册 OPC 服务器。

- **AddGroup**：向服务器添加组。组是客户端用来组织和处理数据项的逻辑容器。
- **CreateGroupEnumerator**：为服务器提供的组创建各种枚举器。
- **GetErrorString**：返回服务器特定错误代码的错误字符串。
- **GetGroupByName**：在给定专用组名称 (早期由同一客户端创建) 的情况下，返回附加接口指针。使用 GetPublicGroupByName 附加至公共组。此函数可用于重新连接到已发布所有接口指针的专用组。
- **GetStatus**：返回服务器的当前状态信息。
- **RemoveGroup** 删除组。发布所有客户端接口时，不会删除组，这是因为服务器本身保持对组的引用。发布所有接口后，客户端可能仍会调用 GetGroupByName。RemoveGroup() 会导致服务器将“最新”参考发布到组中，从而导致组被删除。

IConnectionPointContainer (服务器)

此接口可为 IOPCShutdown 提供访问连接点的权限。

- **EnumConnectionPoints**：为 OPC 组和客户端之间所支持的连接点创建枚举器。OPCServers 必须返回一个包含 IOPCShutdown 的枚举器。允许其他厂商特定回调。
- **FindConnectionPoint**：找到 OPC 服务器与客户端之间的特定连接点。OPCServers 必须支持 IID_IOPCShutdown。允许其他厂商特定回调。

IConnectionPoint (服务器)

此接口将建立客户端回调。

- **Advise**：在连接点和调用者的接收器对象之间建立顾问连接。
- **EnumConnections**：创建枚举器对象，以便通过此连接点中存在的连接进行迭代。
- **GetConnectionInterface**：返回此连接点所管理的发送接口的 IID。
- **GetConnectionPointContainer**：检索指向在概念上拥有连接点的可连接对象的 IConnectionPointContainer 接口指针。
- **Unadvise**：终止先前通过 Advise 方法建立的顾问连接。
- **ShutdownRequest** 允许服务器请求所有客户端与服务器断开连接。

IOPCBrowse

IOPCBrowse 接口提供了可用于浏览服务器地址空间和获取项属性的改进方法。

- **GetProperties**：返回一个 OPCITEMPROPERTIES 数组，一个属性对应一个项 ID。
- **“浏览”**：浏览地址空间的单个分支，并返回零个或多个 OPCBROWSEELEMENT 结构。

IOPCBrowseServerAddressSpace

此接口提供了一种客户端浏览服务器中可用数据项的方法，为用户提供了项 ID 的有效定义列表。它适用于平面或层次化地址空间，以期在网络上良好运行。它还将客户端与服务器供应商特定项 ID 的语法隔离。

- **BrowseAccessPaths**：提供一种浏览项 ID 的可用 AccessPaths 的方法。
- **BrowseOPCItemIDs**：返回由传递的属性所确定的项 ID 列表的 IENUMString。可以在 ChangeBrowsePosition 中设置进行浏览的位置。
- **ChangeBrowserPosition**：提供一种在层次化空间中上下移动的方法。
- **GetItemID**：提供一种在层次化空间中组合完全限定的项 ID 的方法。这是必需的，因为这些浏览函数仅返回构成项 ID 的组件或令牌，且不会返回用于分隔这些令牌的分隔符。此外，在各点上将仅浏览当前节点下方的名称 (例如：单元中的单位)。

- **QueryOrganization**：提供一种方法来确定底层系统本质上是属于平面还是层次化，以及确定服务器可能向客户端表示地址空间信息的方式。平面和层次化空间的表现形式稍有不同。如果结果为平面，则客户端随即了解到，无需将分支或叶标记传递至 BrowseOPCItem ID 或调用 ChangeBrowsePosition。

IOPCItemProperties

此接口可用于浏览与项 ID 相关联的可用属性，以及读取属性的当前值。

- **GetItemProperties**：返回传递的 ID 代码的当前数据值列表。
- **LookUpItemIDs**：返回每个传递的 ID 代码 (如果有的话) 的项 ID 列表。这些指示了符合以下条件的项 ID：可添加到 OPC 组并用于更有效地访问对应于项属性的数据。
- **QueryAvailableProperties**：返回此项 ID 可用属性的 ID 代码和说明列表。此列表可能因不同的项 ID 而有所不同。尽管此列表可能会因底层系统配置的变化而不时受到影响，但预计对于特定的项 ID 是相对稳定的。项 ID 会传递至此函数，因为允许服务器为不同的项 ID 返回不同的属性集。

IOPCItemIO

此接口旨在为基本应用程序获得 OPC 数据提供一种简便的方法。

- **"读取"**：为指定项读取一个或多个值、质量和时间戳。这在功能上类似于 IOPCSyncIO::Read 方法。
- **WriteVQT**：为指定项写入一个或多个值、质量和时间戳。这在功能上类似于 IOPCSyncIO2::WriteVQT，不同之处在于没有任何关联组。如果客户端尝试写入 VQ、VT 或 VQT，则服务器应将它们全部写入或全部不写入。

组

客户端调用 CoCreateInstance 来创建服务器对象和初始接口。

- **QueryInterface**：客户端可以通过调用 IConnectionPointContainer 的 QueryInterface 来询问对象是否支持一些出站接口。如果对象通过递回有效指针做出的回答为"是"，则客户端将获知此对象可以尝试建立连接。
- **AddRef**：增加对象上接口的参考计数。给定对象上接口指针的每个新副本都应该调用此参数。
- **Release**：将接口的参考计数减少 1。

IOPCGroupStateMgt

IOPCGroupStateMgt 允许客户端管理组的整体状态。这主要说明了对组的更新速度和活动状态所做的更改。

- **CloneGroup**：创建组的另一个副本 (具有唯一名称)。
- **GetState**：获取组的当前状态。通常，在调用 SetState 之前会调用此函数来获得此信息的当前值。创建组后，此信息全部由客户端提供，或全部返回给客户端。
- **SetName**：更改专用组的名称。名称必须唯一。无法更改公共组的名称。组名称对于单个客户端到服务器连接而言必须是唯一的。
- **SetState**：设置组的各种属性。这代表一个独立于原始组的新组。

IOPCGroupStateMgt2

添加此接口是为了增强现有 IOPCGroupStateMgt 接口。

- **SetKeepAlive**：导致服务器在没有要报告的新事件时在订阅中提供客户端回调。无需通过调用 GetStatus() 对服务器执行 ping 命令，即可确保客户端服务器和预订阅的健康状态。
- **GetKeepAlive**：返回订阅当前处于活动状态的保持连接时间。

IOPCItemMgt

此接口允许添加、移除和控制项行为的客户端是一个组。

- **AddItems**：向组中添加一个或多个项。允许多次向组中添加同一项，生成具有唯一 ServerHandle 的另一个项。
- **CreateEnumerator**：为组中的项创建枚举器。
- **RemoveItems**：从组中移除项。从组中移除项并不会影响服务器或物理设备的地址空间。它表示客户端是否对这些特定项感兴趣。
- **SetActiveState**：将组中的一个或多个项设置为活动或非活动状态。这将控制是否可以从那些项的读取缓存中获取有效数据，以及它们是否包含在该组的 IAdvise 订阅中。取消激活项不会导致回调，因

为根据定义，处于非活动状态的项不会发生回调。激活项通常会在下一个 UpdateRate 时间段产生 IAdvise 回调。

- **SetClientHandles**：更改组中一个或多个项的客户端句柄。通常情况下，预计客户端会在添加项后设置客户端句柄，且稍后不会对其进行更改。
- **SetDataTypes**：更改组中一个或多个项所请求的数据类型。通常情况下，预计客户端会在添加项后设置请求数据类型，且稍后不会对其进行更改。
- **ValidateItems**：确定项是否有效且可正确添加。它还会返回有关项的信息，如规范化数据类型。它不会以任何方式影响组。

IOPCItemDeadbandMgt

强制为组中的所有活动项执行回调至 IOPCDataCallback::OnDataChange，无论它们是否已更改。非活动项不包括在回调中。MaxAge 值决定可获得数据的位置。只有一个 MaxAge 值，它可确定组中全部活动项的 MaxAge。这意味着某些值可能从缓存中获得，而其他值可以从设备获得，具体取决于缓存中数据的“新鲜度”。

- **SetItemDeadband**：覆盖为每一项的组指定的死区。
- **GetItemDeadband**：获取每个请求项的死区值。
- **ClearItemDeadband**：清除单个项 PercentDeadband，有效地将其恢复为组中设置的死区值。

IOPCItemSamplingMgt

此可选接口允许客户端控制从底层设备中获取组内各项的速率。它不影响 OnDataChange 回调的组更新速率。

- **SetItemSamplingRate**：设置各项的采样速率。就底层设备中的集合而言，这将覆盖组的更新速率。与各项相关联的更新速率不会影响回调时间段。
- **GetItemSamplingRate**：获取先前使用 SetItemSamplingRate 设置的各项的采样速率。
- **ClearItemSamplngRate**：清除先前使用 SetItemSamplingRate 设置的各项的采样速率。该项将恢复为组的更新速率。
- **SetItemBufferEnable**：请求服务器开启或关闭 (具体取决于“启用”属性的值) 标识项数据的缓冲，这些数据是针对更新速率快于组更新速率的项所收集的。
- **GetItemBufferEnable**：查询缓冲所请求项的服务器的当前状态。

IOPCSyncIO

IOPCSyncIO 允许客户端对服务器执行同步读取和写入操作。操作运行至完成。

- **Read**：读取组中一个或多个项的值、质量和时间戳信息。该函数在返回前将运行至完成。数据可从缓存中读取，在此情况下，其应精确到组的 UpdateRate 和百分比死区范围内。数据可从设备中读取，在此情况下，必须执行物理设备的实际读取。缓冲和设备读取的确切实现并不是由规范定义的。
- **Write**：将值写入组中的一个或多个项。该函数将运行至完成。这些值被写入设备，这意味着该函数在验证该设备已实际接受或拒绝数据之后才可以返回。写入不受组或项的活动状态的影响。

IOPCSyncIO2

添加此接口是为了增强现有 IOPCSyncIO 接口。

- **ReadMaxAge**：为指定项读取一个或多个值、质量和时间戳。这在功能上类似于 OPCSyncIO::Read 方法，不同之处在于未指定任何源 (设备或缓存)。服务器将决定是否从设备或缓存获取信息。此决策基于 MaxAge 属性。如果缓存中的信息在 MaxAge 内，则会从缓存获得数据；否则，服务器必须访问设备以获得所请求的信息。
- **WriteVQT**：为指定项写入一个或多个值、质量和时间戳。这在功能上类似于 IOPCSyncIO::Write，不同之处在于可能会写入质量和时间戳。如果客户端尝试写入 VQ、VT 或 VQT，则服务器应写入至全部或全部不写入。

IOPCAsyncIO

IOPCAsyncIO 允许客户端对服务器执行异步读取和写入操作。这些操作已加入队列，且该函数会立即返回，以便客户端可以继续运行。每次操作均被视为一个事务，并且与事务 ID 相关联。完成操作后，将对客户端中的 IAdvise Sink (如果已建立) 进行回调。回调中的信息指示了事务 ID 和错误结果。按照约定，0 是一个无效的事务 ID。

- **Refresh**：请求服务器取消未完成的事务。
- **Read**：读取组中的一个或多个项。结果通过 IDataObject 所建立的 IAdvise Sink 连接返回。对于缓存读取，数据仅在组和项均处于活动状态时才有效。设备读取不受组或项的活动状态的影响。
- **Refresh**：强制为组中的所有活动项执行回调，无论它们是否已更改。非活动项不包括在回调中。
- **Write**：写入组中的一个或多个项。结果通过 IDataObject 所建立的 IAdviseSink 连接返回。

IDataObject

IDataObject 在 OPCGroup 上实现，而非单个项上。这允许使用 OPC 数据流格式在客户端与组之间创建 Advise 连接，以实现高效的数据传输。

- **DAdvise**：在 OPC 组与客户端之间为特定的流格式创建连接。
- **DUnadvise**：终止 OPC 组与客户端之间的连接。

IAdviseSink

客户端只需完全实现 OnDataChange 即可。

- **OnDataChange**：此方法由客户端提供，用于处理来自 OPC 组、有关基于异常的数据更改、异步读取和刷新以及异步写入完成的通知。

IAsyncIO2

此接口类似于 IOPCAsync(OPC 1.0)，旨在替换 IOPCAsyncIO。OPC 2.05 中已添加该接口。

- **Cancel2**：请求服务器取消未完成的事务。
- **GetEnable**：使用 SetEnable 检索最后一个“回调启用”值集。
- **Read**：读取组中的一个或多个项。结果通过服务器的 IConnectionPointContainer 所建立的客户端 IOPCDataCallback 连接返回。从设备读取不受组或项的活动状态的影响。
- **Refresh2**：强制为组中的所有活动项执行回调至 IOPCDataCallback::OnDataChange，无论它们是否已更改。非活动项不包括在回调中。
- **SetEnable**：控制 OnDataChange 操作。将“启用”设置为 False 将禁用任何事务 ID 为 0 的 OnDataChange 回调 (不是 Refresh 的结果)。创建组时此变量的初始值为 True；默认情况下会启用 OnDataChange 回调。
- **Write**：写入组中的一个或多个项。结果通过服务器的 IConnectionPointContainer 所建立的客户端 IOPCDataCallback 连接返回。

IAsyncIO3

添加此接口是为了增强现有 IOPCAsyncIO2 接口。

- **ReadMaxAge**：为指定项读取一个或多个值、质量和时间戳。这在功能上类似于 OPCSyncIO::Read 方法，不同之处在于其为异步且未指定任何源 (设备或缓存)。服务器将决定是否从设备或缓存获取信息。此决策基于 MaxAge 属性。如果缓存中的信息在 MaxAge 内，则会从缓存获得数据；否则，服务器必须访问设备以获得所请求的信息。
- **WriteVQT**：为指定项写入一个或多个值、质量和时间戳。结果通过服务器的 IConnectionPointContainer 所建立的客户端 IOPCDataCallback 连接返回。这在功能上类似于 IOPCAsyncIO2::Write，不同之处在于可能会写入质量和时间戳。如果客户端尝试写入 VQ、VT 或 VQT，则服务器应将它们全部写入或都不写入。
- **RefreshMaxAge**：强制为组中的所有活动项执行回调至 IOPCDataCallback::OnDataChange，无论它们是否已更改。非活动项不包括在回调中。MaxAge 值决定可获得数据的位置。只有一个 MaxAge 值，它可确定组中全部活动项的 MaxAge。这意味着某些值可能从缓存中获得，而其他值可以从设备获得，具体取决于缓存中的数据类型。

IConnectionPointContainer (组)

此接口提供了类似于 IDataObject 的功能，但更易于实现和理解。它还提供了 IDataObject 接口缺失的功能。客户端必须使用新版 IOPCAsyncIO2 接口才能通过与此接口建立的连接进行通信。旧版 IOPCAsnyc 将一如既往继续通过 IDataObject 连接进行通信。

- **EnumConnectionPoints**：为 OPC 组和客户端之间所支持的连接点创建枚举器。
- **FindConnectionPoint**：找到 OPC 组与客户端之间的特定连接点。

IConnectionPoint (组)

此接口将建立客户端回调。

- **Advise**：在连接点和调用者的接收器对象之间建立顾问连接。
- **EnumConnections**：创建枚举器对象，以便通过此连接点中存在的连接进行迭代。
- **GetConnectionInterface**：返回此连接点所管理的发送接口的 IID。
- **GetConnectionPointContainer**：检索指向在概念上拥有连接点的可连接对象的 IConnectionPointContainer 接口指针。
- **Unadvise**：终止先前通过 Advise 方法建立的顾问连接。

IOPCDataCallback

要使用连接点，客户端必须创建一个同时支持 IUnknown 和 IOPCDataCallback 接口的对象。

- **OnDataChange**：此方法由客户端提供，用于处理来自 OPC 组、有关基于异常的数据更改和刷新的通知。
- **OnReadComplete**：此方法由客户端提供，用于在完成 Async 读取后处理来自 OPC 组的通知。
- **OnWriteComplete**：此方法由客户端提供，用于在完成 AsyncIO2 写入后处理来自 OPC 组的通知。
- **OnCancelComplete**：此方法由客户端提供，用于在完成 Async 取消后处理来自 OPC 组的通知。

IEnumOPCItemAttributes

IEnumOPCItemAttributes 允许客户端找出组的内容以及这些项的属性。大部分返回的信息在客户端调用 AddItem 时由客户端提供或者返回给客户端。

- **Clone**：创建枚举器的另一个副本。新的枚举器最初处于与当前枚举器相同的状态。
- **Next**：获取组中的下一个 'celt' 项。
- **Reset**：将枚举器重置回第一项。
- **Skip**：跳过下一个 'celt' 属性。

有关连接点一般原则的详细信息，请参阅 Microsoft 文档。

OPC UA 服务

有关特定 OPC 诊断事件的详细信息，请从下表中选择一个链接。

AttributeServiceSet
DiscoveryServiceSet
MonitoredItemServiceSet
OtherServices
SecureChannelServiceSet
SessionServiceSet
SubscriptionServiceSet
ViewServiceSet

AttributeServiceSet

此服务集提供了用于访问作为节点一部分的属性的服务。

- **Read**：此服务用于读取一个或多个节点的一个或多个属性。对于元素已建立索引的构造属性值，如数组，此服务允许客户端将整个索引值集读取为复合，以读取各个元素或读取复合元素的范围。
- **Write**：此服务用于将值写入一个或多个节点的一个或多个属性。对于元素已建立索引的构造属性值，如数组，该服务允许客户端将整个索引值集写入为复合，以写入各个元素或写入复合元素的范围。

DiscoveryServiceSet

此服务集所定义的服务用于发现由服务器实现的端点，并读取这些端点的安全配置。

- **FindServers**：此服务会返回服务器或发现服务器已知的服务器。
- **GetEndpoints**：此服务会返回服务器支持的端点以及建立安全信道和会话所需的所有配置信息。

MonitoredItemServiceSet

此服务集允许客户端通过定义监控项来订阅数据和事件。每个监控项均标识了要监控的项以及用于发送通知的订阅。要监控的项可能是任何节点属性。

- **CreateMonitoredItems**：此服务用于创建一个或多个 MonitoredItem 并将其添加到“订阅”。删除“订阅”后，服务器将自动删除 MonitoredItem。
- **DeleteMonitoredItems**：此服务用于移除“订阅”的一个或多个 MonitoredItem。删除 MonitoredItem 后，其触发的项链接也将被删除。
- **ModifyMonitoredItems**：此服务用于修改“订阅”的 MonitoredItem。该服务器会立即应用 MonitoredItem 设置的更改。
- **SetMonitoringMode**：此服务用于为“订阅”的一个或多个 MonitoredItem 设置监控模式。将模式设置为禁用将导致所有已加入队列的通知被删除。
- **SetTriggering**：此服务用于创建和删除触发项的触发链接。当触发项及其链接的监控模式在默认情况下不允许这样时，这些触发项和链接会引起监控项报告采样。

OtherServices

OtherServices 表示杂项服务和通知。

- **ServiceFault**：一旦服务失败，系统即会提供此响应。
- **Unsupported**：此服务器不支持这些服务。

SecureChannelServiceSet

此服务集定义了用于打开通信信道的服务，可确保与服务器交换的所有消息的机密性和完整性。

- **CloseSecureChannel**：此服务用于终止 SecureChannel。
- **OpenSecureChannel**：此服务用于打开或续订 SecureChannel，可用于确保会话期间消息交换的机密性和完整性。此服务要求通信堆栈在发送和接收消息时为消息应用各种安全算法。

SessionServiceSet

此服务集为会话上下文中的应用程序层连接建立定义了服务。

- **ActivateSession**：客户端可使用此服务来指定与会话关联的用户的标识。
- **Cancel**：此服务用于取消任何未完成的服务请求。成功取消服务请求应以 Bad_RequestCancelledByClient ServiceFaults 做出响应。
- **CloseSession**：此服务用于终止会话。
- **CreateSession**：客户端使用此服务来创建会话，且服务器将返回唯一可标识该会话的两个值。第一个值为 sessionId，用于标识服务器 AddressSpace 中的会话。第二个值为 authenticationToken，用于将传入请求与会话相关联。

SubscriptionServiceSet

订阅用于向客户端报告来自 MonitoredItem 的通知。

- **CreateSubscription**：此服务用于创建订阅。订阅会监视通知的一组 MonitoredItem 并将其返回到客户端以响应 Publish 请求。
- **DeleteSubscriptions**：调用此服务以删除属于客户端会话的一个或多个订阅。成功完成此服务会导致所有使用订阅的 MonitoredItem 被删除。
- **ModifySubscription**：此服务用于修改订阅。
- **Publish**：此服务用于两个目的。首先，它用于确认收到一个或多个订阅的 NotificationMessages。其次，它用于请求服务器返回 NotificationMessage 或保持连接消息。由于 Publish 请求并未定向到特定的订阅，因此它们可用于任何订阅。
- **Republish**：此服务请求订阅从其重新传输队列重新发布 NotificationMessage。
- **SetPublishingMode**：此服务用于启用或禁用发送一个或多个订阅的通知。
- **TransferSubscriptions**：使用此服务在会话之间传输订阅及其 MonitoredItem。

ViewServiceSet

客户端使用此服务集的浏览服务来浏览 AddressSpace。

- **Browse**：此服务用于搜索指定节点的参考。浏览服务还支持原始筛选功能。
- **BrowseNext**：此服务用于请求下一组 Browse 或 BrowseNext 响应信息，该响应信息量太大而无法在单个响应中发送。此上下文中的“太大”意味着服务器无法返回一个更大的响应，或要返回的结果数超过了客户端在原始浏览请求中指定的最大返回结果数。
- **RegisterNodes**：客户端可使用此服务来注册它们了解将要重复访问的节点 (例如写入、读取)。它允许服务器进行所需的任何设置，以便访问操作更有效率。
- **TranslateBrowsePathsToNodeIds**：此服务用于请求服务器将一个或多个浏览路径转换为 NodeId。每个浏览路径由起始节点和 RelativePath 构成。指定的起始节点标识了 RelativePath 所基于的节点。RelativePath 包含一系列 ReferenceType 和 BrowseName。
- **UnregisterNodes**：此服务用于取消注册已通过 RegisterNodes 服务获得的 NodeId。

有关连接点一般原则的详细信息，请参阅 Microsoft 文档。

通信诊断

服务器的诊断功能可提供与通信驱动程序性能相关的实时信息。在诊断查看器中可以查看所有读取和写入操作，并可以在 OPC 客户端应用程序中通过内置诊断标记对其进行直接跟踪。诊断查看器还提供实时协议视图，可用于对关键通信参数设置 (如波特率、奇偶性或设备 ID) 进行更改。更改的效果将实时显示。设置正确的通信和设备设置后，将显示与设备之间的数据交换。

启用“通信诊断”

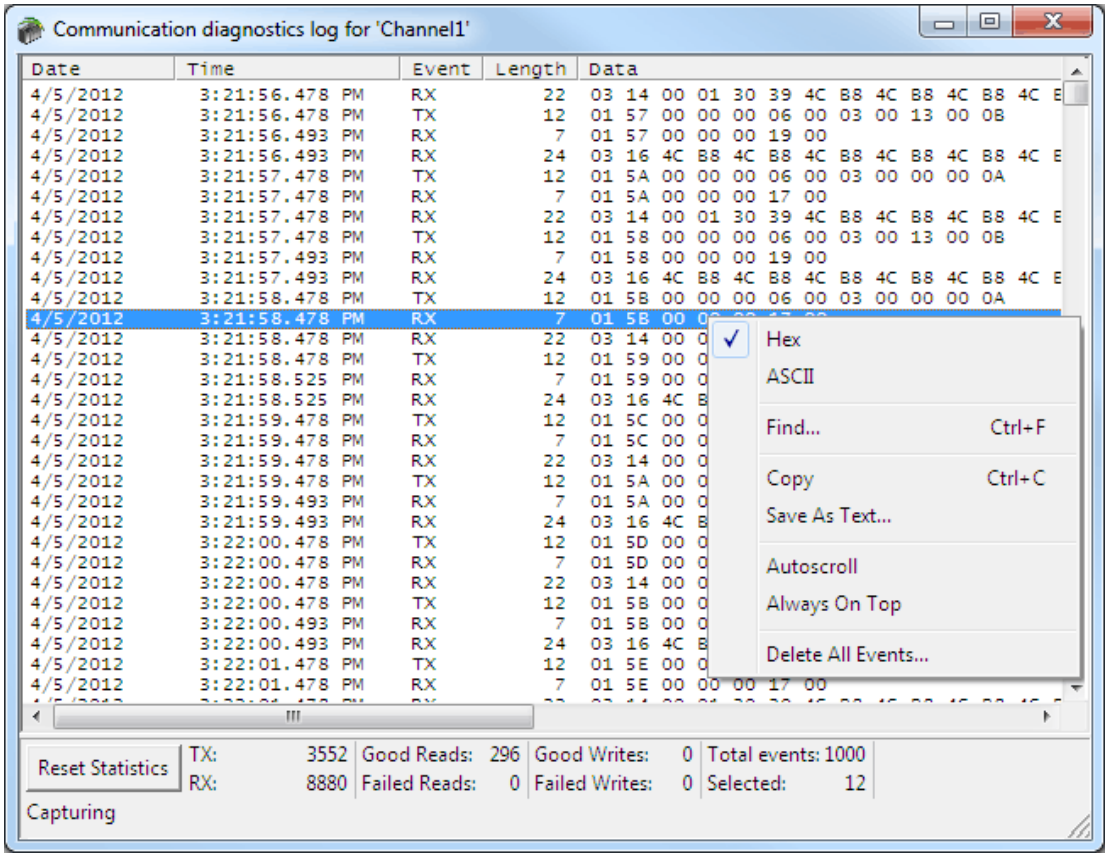
要启用“通信诊断”，请在“项目视图”中右键单击信道，然后单击**“属性”|“启用诊断”**。或者，双击信道，然后选择**“启用诊断”**。创建信道后，用户可以启用诊断。

另请参阅：信道属性 - 常规

访问“通信诊断查看器”

要访问“通信诊断查看器”，请在“项目视图”中右键单击信道或设备，然后选择**“诊断”**。或者，选择信道或设备，然后单击**“视图”|“通信诊断”**。“通信诊断查看器”以非模式形式运行，这允许它在服务器中已打开其他对话框的情况下存在。查看器打开后，便应开始捕获实时协议数据。如果正在正确进行通信，则服务器与设备之间将存在通信消息流。用户应能够查看 TX 和 RX 事件以及总事件数。

注意：当捕获被禁用时，虽然可以打开“通信诊断查看器”，但在其启用前没有任何诊断。启用后，查看器将显示“捕获”。禁用后，查看器将显示“诊断捕获已禁用”。



重置统计数据

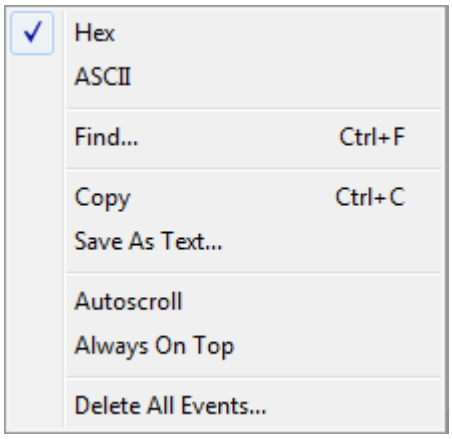
单击“重置统计数据”可将 TX、RX、读取成功次数、读取失败次数、写入成功次数和写入失败次数的计数设置为零。不会将总事件数设置为零，因为它指定了查看器中的实际事件数。

有关日志设置的信息，请参阅设置 - 事件日志。

访问上下文菜单

如果通信未正常运行，则用户可以访问信道属性并修改通信参数。即使已经显示信道属性，诊断窗口仍会显示，使用户可以更改属性并监视其效果。在访问任何对话框之前，必须先显示诊断窗口。

如果通信问题仍然存在，请右键单击诊断窗口以调用上下文菜单。然后，使用可用的选项来调整诊断窗口的操作。

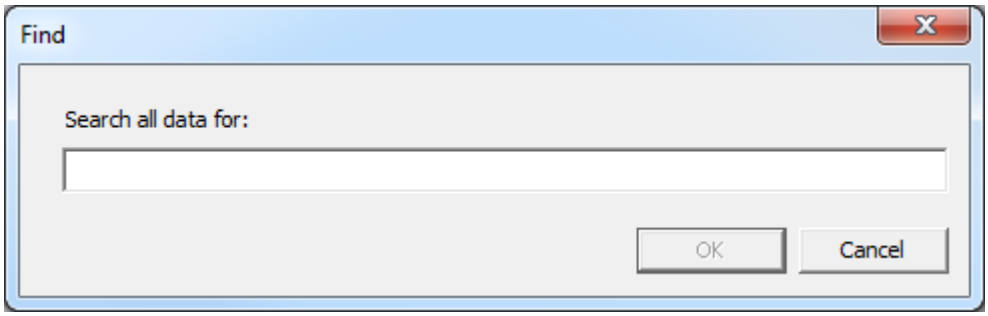


选项说明如下：

- **“十六进制”**：启用后，将使用十六进制符号设置 TX/RX 详细信息的格式。
- **ASCII**：启用后，将使用 ASCII 符号设置 TX/RX 详细信息的格式。
- **“查找”**：此选项将调用一个对话框，用于输入要应用于事件详细信息的搜索字符串。有关详细信息，请参阅查找。
- **“复制”**：此选项会将协议捕获缓冲区的内容设置为文本格式，以便于在电子邮件或传真消息中进行“剪切和粘贴”。此信息可帮助技术支持人员分析和诊断诸多通信问题。
- **“另存为文本文件”**：此选项可将视图中的所有事件保存为指定的文件名 (以文本格式)。
- **“自动滚动”**：此选项将在收到新事件时滚动显示，以确保最新事件可见。当用户手动选择事件时 (或通过“查找”/“查找下一个”进行选择时)，它将关闭。
- **“始终置顶”**：此选项将强制诊断窗口始终位于所有其他应用程序窗口的顶部。这是默认设置。
- **“删除所有事件”**：此选项将清除事件日志中保留的日志，从而删除数据。

“查找”

此对话框用于在诊断视图中搜索在客户端和服务器之间传送的密钥信息。



属性说明如下：

- **“搜索其所有数据”**：此字段可指定搜索条件。

注意：当找到带有指定文本的事件或详细信息时，包含该文本的行将突出显示。要执行“查找下一个”操作 (以及查找指定文本的下一个匹配项)，请按 "F3"。当找到最后一个匹配项后，将显示一个消息框，以指示这一情况。通过按 "Ctrl + F"，用户可以随时更改搜索条件。

事件日志消息

以下信息涉及发布到主要用户界面中“事件日志”窗格的消息。请参阅有关筛选和排序“事件日志”详细信息视图的服务器帮助。服务器帮助包含许多常见的消息，因此也应对其进行搜索。通常，其中会尽可能提供消息的类型 (信息、警告) 和故障排除信息。

服务器汇总信息

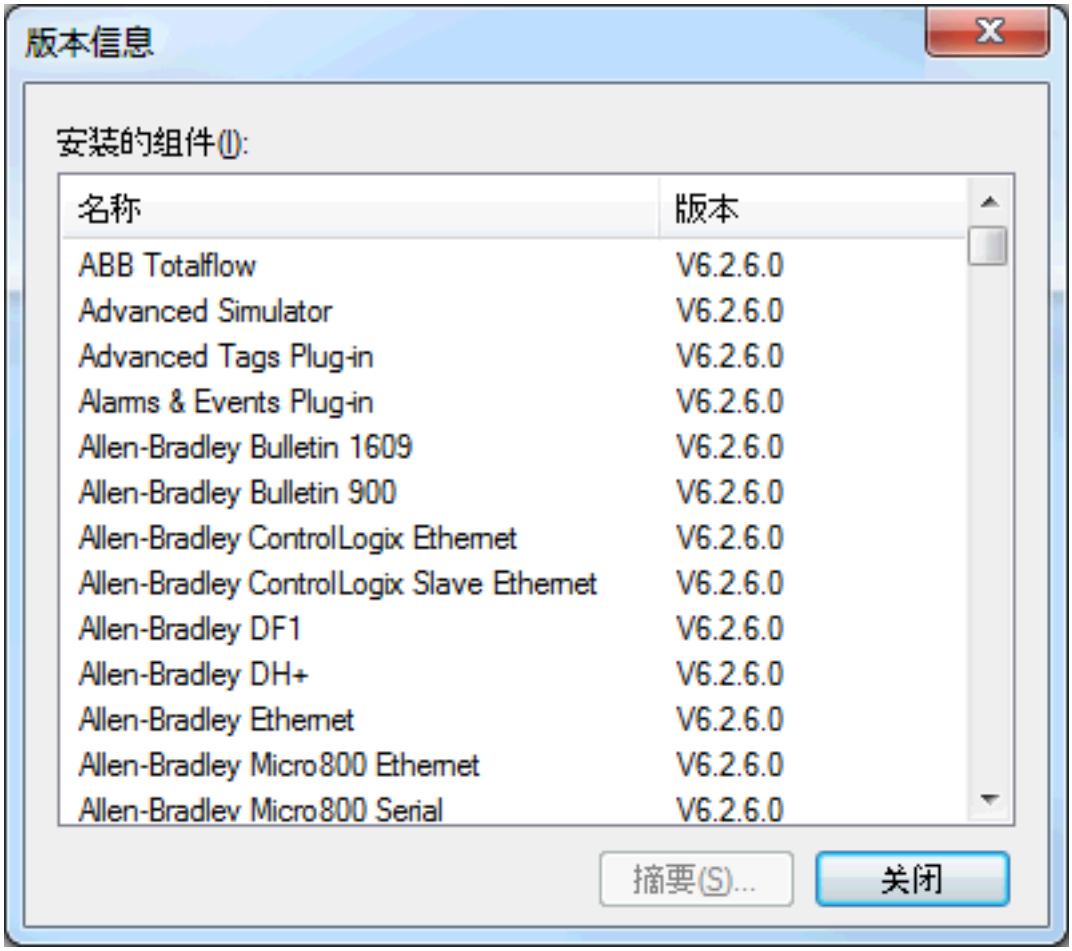
服务器提供了有关其自身以及当前所安装的驱动程序和插件的基本汇总信息。

关于服务器

此服务器版本随时可供查看，并提供了便于查找驱动程序特定信息的方式。要对其进行访问，请单击服务器配置中的**“帮助”|“支持信息”**。要显示所有已安装组件的版本信息，请单击**“版本”**。

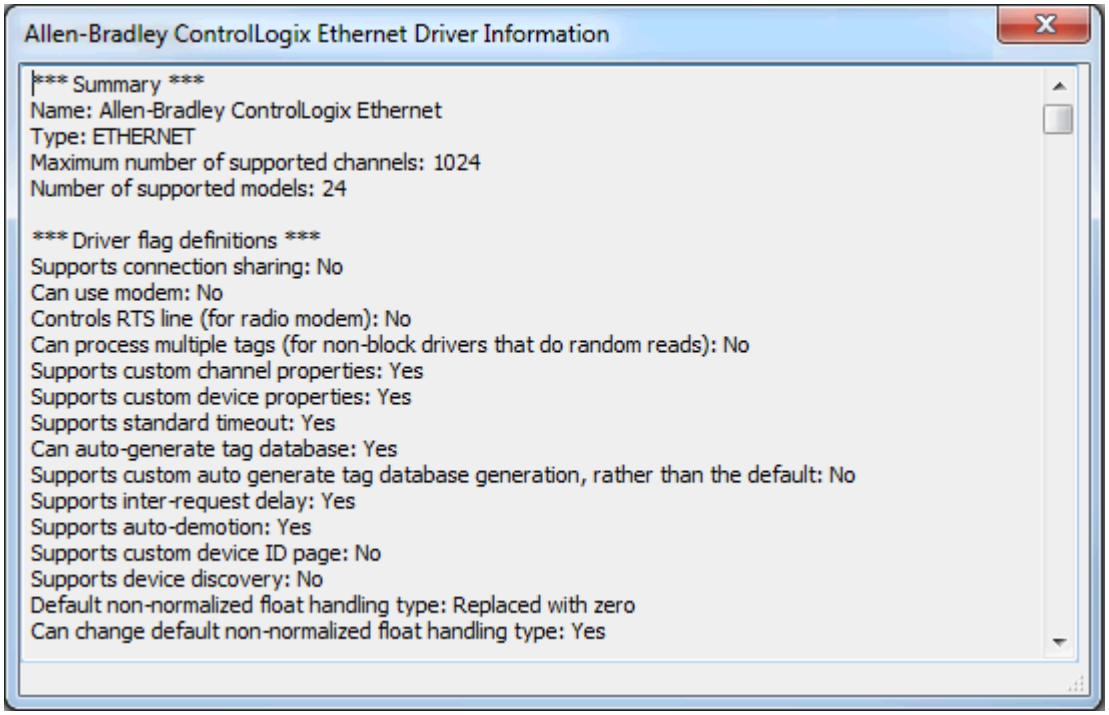
组件版本信息

版本信息窗口显示所有已安装的驱动程序和插件及其版本号。有关驱动程序特定的信息，请选择一个组件，然后单击**“汇总”**。



驱动程序信息

驱动程序信息窗口提供了驱动程序默认设置的汇总信息。例如，每个驱动程序都显示其支持的最大信道数量。



各个信息部分的说明如下：

"汇总"提供驱动程序的名称和类型、支持的最大信道数量以及驱动程序中的模型数。

"COMM 默认值"显示驱动程序的默认设置，这可能与所配置设备的设置不一致。

"驱动程序标志定义"显示驱动程序库函数，并指示它们是否已在驱动程序中启用。

"模型信息"显示设备特定的寻址和功能。除了其寻址值和其他功能之外，它还列出了每个支持模型的名称。

未找到 <名称> 设备驱动程序，或者无法加载。

错误类型：

错误

可能的原因：

1. 如果项目转移到另一台计算机，所需的驱动程序可能尚未安装。
2. 指定的驱动程序可能已从安装的服务器中删除。
3. 指定驱动程序的版本可能不适于已安装的服务器版本。

可能的解决方案：

1. 重新进行服务器安装，并添加所需的驱动程序。
2. 重新进行服务器安装，并重新安装指定的驱动程序。
3. 确保驱动程序没有置于已安装服务器目录 (与服务器版本不同步)。

无法加载 '<名称>' 驱动程序，因为存在多个副本 ('<名称>' 和 '<name>')。请移除冲突的驱动程序并重新启动应用程序。

错误类型：

错误

可能的原因：
服务器的驱动程序文件夹中存在多个版本的驱动程序 DLL。

可能的解决方案：
1. 重新进行服务器安装，并重新安装指定的驱动程序。
2. 请联系技术支持并验证正确的版本。移除无效的驱动程序，然后重新启动服务器并加载项目。

项目文件无效。
错误类型：
错误

未能打开调制解调器线路 '<line>' [TAPI 错误 = <code>]。
错误类型：
错误

可能的原因：
TAPI 尝试打开服务器的调制解调器线路，遇到错误。

可能的解决方案：
请更正指定错误的条件，然后重新尝试打开调制解调器线路。

由于驱动程序级别故障，无法添加通道。
错误类型：
错误

可能的原因：
由于驱动程序出现问题，尝试失败。

可能的解决方案：
请查阅有关驱动器错误的详细消息，并更正相关的问题。

由于驱动程序级别故障，无法添加设备。
错误类型：
错误

可能的原因：
由于驱动程序出现问题，尝试失败。

可能的解决方案：
请查阅有关驱动器错误的详细消息，并更正相关的问题。

版本不匹配。
错误类型：
错误

无效的 XML 文档:
错误类型：
错误

可能的原因：
服务器无法解析指定的 XML 文件。

可能的解决方案：
如果使用第三方 XML 编辑器编辑了服务器项目，请通过服务器和驱动程序架构来验证格式正确。

无法加载项目 <名称>:

错误类型：
错误

可能的原因：
创建项目的服务器版本与要加载项目的版本不兼容。

可能的解决方案：
这通常是因为项目在较新的服务器版本中创建，而在较旧的版本中打开。

注意：
我们尽可能确保服务器具有向后兼容性，使得较旧版本中创建的项目能够在较新的版本中加载。但是，由于服务器和驱动程序的新版本所具有的一些属性和配置不存在于旧版本，因此可能无法在较新的版本中打开或加载旧版本的项目文件。

无法将项目备份到 '<路径>' [<原因>]。保存操作已中止。请验证目标文件未锁定并且具有读/写权限。要继续保存此项目而不进行备份，请在“工具”|“选项”|“常规”下取消选择备份选项，然后重新保存项目。

错误类型：
错误

可能的原因：

1. 目标文件可能未被其他应用程序锁定。
2. 目标文件或其所在的文件夹不允许读/写访问。

可能的解决方案：

1. 请确保目标文件未被其他应用程序锁定，解锁文件或关闭应用程序。
2. 请确保目标文件及其所在的文件夹允许读写访问。

未找到 <feature name>，或者无法加载。

错误类型：
错误

可能的原因：
功能未安装，或者不在预期位置。

可能的解决方案：
重新运行服务器安装，并选择指定的功能进行安装。

无法保存项目文件 <名称>:

错误类型：
错误

设备发现超出最大允许设备数 <count>。请限制发现范围并重试。

错误类型：
错误

加载此项目需要 <feature name>。

错误类型：
错误

当前语言不支持加载 XML 项目。要加载 XML 项目，请在“服务器管理”中将产品语言选择更改为英语。

错误类型：
错误

可能的原因：
只允许在英语环境中加载 XML 项目。

可能的解决方案：
在“服务器管理”中将产品语言选择更改为英语，然后重试。

由于缺少对象，无法加载项目。| 对象 = '<object>'。

错误类型：
错误

可能的原因：
编辑 JSON 项目文件可能将其置于无效状态。

可能的解决方案：
还原对 JSON 项目文件的所有更改。

加载项目时遇到无效的型号。| 设备 = '<设备>'。

错误类型：
错误

可能的原因：
此版本的服务器不支持指定设备的一个型号。

可能的解决方案：
用服务器的新版本打开此项目。

无法添加设备。此通道中可能存在重复设备。

错误类型：
错误

自动生成的标记 '<tag>' 已存在，无法覆盖。

错误类型：
警告

可能的原因：
虽然服务器正在为标记数据库重新生成标记，但已将其设置为不覆盖已存在的标记。

可能的解决方案：
如果这不是所需的操作，请更改设备的“对于重复标记”属性的设置。

无法为设备 '<device>' 生成标记数据库。设备未响应。

错误类型：
警告

可能的原因：

1. 设备未响应通信请求。
2. 指定的设备未开启、未连接或出现错误。

可能的解决方案：

1. 请验证设备已通电并且计算机已开启 (使服务器可以与其连接)。
2. 请验证所有布线正确无误。
3. 请验证设备 ID 正确无误。
4. 请更正设备故障并重试标记生成。

无法为设备 '<device>' 生成标记数据库:

错误类型：
警告

可能的原因：
指定的设备未开启、未连接或出现错误。

可能的解决方案：
请更正设备故障并重试标记生成。

自动生成造成的覆盖太多，已停止发布错误消息。

错误类型：
警告

可能的原因：

1. 为避免填满错误日志，对于自动标记生成期间无法覆盖的标记，服务器已停止发布其相关错误消息。
2. 请缩小自动标记生成的范围，或者消除有问题的标记。

未能添加标记 '<tag>'，因为地址太长。最大地址长度为 <number>。

错误类型：
警告

线路 '<line>' 已使用。

错误类型：
警告

可能的原因：
目标调制解调器线路已开通，可能正由其他应用程序使用。

可能的解决方案：
请查找占用调制解调器的应用程序，关闭该应用程序或释放占用线路。

线路 '<line>' 存在硬件错误。

错误类型：
警告

可能的原因：
对连接到调制解调器的设备中的标记发出请求后，返回了硬件错误。

可能的解决方案：
请禁用设备的数据收集。在调制解调器连接到目标调制解调器之后，再启用。

注意：
错误在首次扫描时出现，未重现。

未对线路 '<line>' 的连接提供通信处理。

错误类型：
警告

可能的原因：
曾尝试使用非指定的通信处理来连接到调制解调器线路。

可能的解决方案：
请验证调制解调器已正确安装并初始化。

无法在线路 '<line>' 拨号。

错误类型：
警告

可能的原因：
调制解调器不处于允许拨号状态。

可能的解决方案：
要拨号，线路必须空闲。请监视 _Mode Modem 标记并在其指示空闲状态时拨号。

无法使用通道 '<name>' 上的网络适配器 '<adapter>'。正在使用默认网络适配器。

错误类型：
警告

可能的原因：
此计算机不存在项目指定的网络适配器。服务器使用默认网络适配器。

可能的解决方案：
请选择计算机使用的网络适配器并保存项目。

也可以看看：
通道属性 - 网络接口

系统正在拒绝对引用设备 '<channel device>' 上的型号类型的更改。

错误类型：
警告

TAPI 线路初始化失败: <code>。

错误类型：
警告

可能的原因：
启动运行时不需要电话服务正在运行。如果禁用此服务并将串行驱动程序添加到项目，则会报告此错误消息。

可能的解决方案：

1. 如果未使用调制解调器通信，则不需要任何操作。
2. 如果需要调制解调器通信，则必须在计算机上启动电话服务。

验证错误位于 '<tag>': <error>。

错误类型：
警告

可能的原因：
曾尝试对指定标记设置无效的参数。

无法加载驱动程序 DLL '<名称>'。

错误类型：
警告

可能的原因：
启动项目时无法加载指定驱动程序。

可能的解决方案：

1. 请验证已安装驱动程序的版本。在网站查验驱动程序版本是否适于已安装的服务器版本。
2. 如果驱动程序损坏，请将其删除并重新运行服务器安装。

注意：
造成这一问题通常是由于驱动程序 DLL 损坏或驱动程序与服务器版本不兼容。

验证错误位于 '<tag>': 缩放参数无效。

错误类型：
警告

可能的原因：
曾尝试对指定的标记设置无效的缩放参数。

也可以看看：
标记属性 - 缩放

无法对线路 '<line>' 应用调制解调器配置。

错误类型：
警告

可能的原因：
TAPI Manager 无法对服务器应用配置更改。

可能的解决方案：

1. 请验证调制解调器的布线。
2. 请验证调制解调器已设为接受配置更改。
3. 请验证调制解调器未被其他应用程序使用。

设备 '<device>' 已自动降级。

错误类型：
警告

可能的原因：
与指定设备的通信失败。该设备已从轮询周期降级。

可能的解决方案：

1. 如果设备无法重新连接，请调查通信中断的实际原因并进行纠正。
2. 要防止设备降级，请禁用“自动降级”。

也可以看看：
自动降级

<Source>: 无效的以太网封装 IP '<address>'。

错误类型：
警告

可能的原因：
对以太网封装通道上的设备指定的 IP 地址不是有效的 IP 地址。

可能的解决方案：
请更正 XML 文件中的 IP，然后重新加载项目。

注意：
如果加载的 XML 格式文件由第三方 XML 软件创建或编辑，则会出现此错误。

'<product>' 驱动程序当前不支持 XML 持久存储。请使用默认文件格式进行保存。

错误类型：
警告

可能的原因：
指定的驱动程序不支持 XML 格式。

可能的解决方案：
以 .opf 格式保存项目。

无法加载插件 DLL '<名称>'。

错误类型：
警告

可能的原因：
启动项目时无法加载指定的插件。

可能的解决方案：

1. 请验证已安装插件的版本。在网站查验插件版本是否与已安装的服务器兼容。如果不兼容，请更改服务器或重新运行服务器安装。
2. 如果插件损坏，请将其删除并重新运行服务器安装。

注意：
造成这一问题通常是由于插件 DLL 损坏或插件与服务器版本不兼容。

为 '<device>' 设备设置的时区是 '<zone>'。对于系统而言，这是无效的时区。正在将时区更改为默认时区 '<zone>'。

错误类型：
警告

无法加载驱动程序 DLL '<名称>'。原因:

错误类型：
警告

可能的原因：
启动项目时无法加载指定的插件。

可能的解决方案：

1. 请验证已安装插件的版本。在网站查验插件版本是否与已安装的服务器兼容。如果不兼容，请更改服务器或重新运行服务器安装。
2. 如果插件损坏，请将其删除并重新运行服务器安装。

无法加载插件 DLL '<名称>'。原因:

错误类型：
警告

可能的原因：
启动项目时无法加载指定的插件。

可能的解决方案：

1. 请验证已安装插件的版本。在网站查验插件版本是否与已安装的服务器兼容。如果不兼容，请更改服务器或重新运行服务器安装。
2. 如果插件损坏，请将其删除并重新运行服务器安装。

电话簿必须至少有一个号码，通道才能自动拨号。| 通道 = '<channel>'。

错误类型：
警告

可能的原因：
“自动拨号”属性设置为“启用”，电话簿中没有条目。

可能的解决方案：
如果需要自动拨号，请将电话号码条目添加到电话簿。如果不需要自动拨号，请禁用“自动拨号”。

通道需要启用自动拨号和其电话簿中的至少一个号码，以使用共享调制解调器连接。| 通道 = '< 通道>'。

错误类型：
警告

可能的原因：
通道与一个和多个现有通道共享调制解调器，未启用“自动拨号”也没有自动拨号的电话号码。

可能的解决方案：

1. 对报告通道启用自动拨号。
2. 向报告通道的电话簿添加电话号码。

指定的网络适配器在通道 '%1' 上无效。| 适配器 = '%2'。

错误类型：
警告

可能的原因：
项目中指定的网络适配器在此计算机中不存在。

可能的解决方案：
请选择计算机使用的网络适配器并保存项目。

也可以看看：
通道属性 - 网络接口

标记生成请求未创建标记。有关详细信息，请参阅事件日志。

错误类型：
警告

可能的原因：
驱动程序未生成标记信息，且未提供原因。

可能的解决方案：
事件日志中的信息可能有助于诊断排除问题。

TAPI 配置已更改，正在重新初始化...

错误类型：
信息化

<Product> 设备驱动程序加载成功。

错误类型：
信息化

正在启动 <名称> 设备驱动程序。

错误类型：
信息化

正在停止 <名称> 设备驱动程序。

错误类型：
信息化

正在线路 '<modem>' 拨号 '<number>'。

错误类型：
信息化

线路 '<modem>' 已断开连接。

错误类型：
信息化

线路 '<modem>' 的拨号已被用户取消。

错误类型：
信息化

线路 '<modem>' 已连接，速率 <rate> 波特。

错误类型：
信息化

远端线路在 '<modem>' 占线。

错误类型：
信息化

远端线路在 '<modem>' 未应答。

错误类型：
信息化

'<modem>' 无拨号音。

错误类型：
信息化

电话号码无效 (<number>)。

错误类型：
信息化

'<modem>' 的拨号已中止。

错误类型：
信息化

线路在远程站点的 '<modem>' 断开。

错误类型：
信息化

线路 '<modem>' 检测到呼入。

错误类型：
信息化

调制解调器线路开通: '<modem>'。

错误类型：
信息化

调制解调器线路关闭: '<modem>'。

错误类型：
信息化

<Product> 设备驱动程序已从内存卸载。

错误类型：
信息化

线路 '<modem>' 已连接。

错误类型：
信息化

设备 '<device>' 已启用模拟模式。

错误类型：
信息化

设备 '<device>' 已禁用模拟模式。

错误类型：
信息化

正在尝试对设备 '<device>' 自动生成标记。

错误类型：
信息化

已完成设备 '<device>' 的自动标记生成。

错误类型：
信息化

正在开始对调制解调器线路 '<modem>' 断开连接。

错误类型：
信息化

客户端应用程序已对设备 '<device>' 启用自动降级。

错误类型：
信息化

可能的原因：
连接到服务器的客户端应用程序已对指定设备启用或禁用“自动降级”。

可能的解决方案：
要限制客户端应用程序执行此操作，请通过“用户管理器”禁用其写入系统级别标记的能力。

也可以看看：
用户管理器

设备 '<device>' 已启用数据收集。

错误类型：
信息化

设备 '<device>' 已禁用数据收集。

错误类型：
信息化

项目不允许对象类型 '<name>'。

错误类型：
信息化

已将项目 '<名称>' 的备份创建到 '<路径>'。

错误类型：

信息化

已自动升级设备 '<device>' 以确定是否可以重新建立通信。

错误类型：
信息化

未能加载库: <名称>。

错误类型：
信息化

未能读取构建清单资源: <名称>。

错误类型：
信息化

项目文件由此软件的最新版本创建。

错误类型：
信息化

客户端应用程序已对设备 '<device>' 禁用自动降级。

错误类型：
信息化

电话号码优先级已更改。| 电话号码名称 = '<名称>', 更新的优先级 = '<priority>'。

错误类型：
信息化

设备 '<设备>' 的标记生成结果。| 创建的标记 = <计数>。

错误类型：
信息化

设备 '<设备>' 的标记生成结果。| 创建的标记 = <计数>, 覆盖的标记 = <计数>。

错误类型：
信息化

设备 '<设备>' 的标记生成结果。| 创建的标记 = <计数>, 未覆盖的标记 = <计数>。

错误类型：
信息化

访问对象被拒绝。| 用户 = '<account>', 对象 = '<object path>', 权限 =

错误类型：
安全

正在更改运行时操作模式。

错误类型：
信息化

更改运行时操作模式完成。

错误类型：
信息化

关闭以执行安装。

错误类型：
信息化

OPC ProgID 已添加到 ProgID 重定向列表。| ProgID = '<ID>'。

错误类型：
信息化

OPC ProgID 已从 ProgID 重定向列表移除。| ProgID = '<ID>'。

错误类型：
信息化

无效的 ProgID 条目已从 ProgID 重定向列表删除。| ProgID = '<ID>'。

错误类型：
信息化

管理员密码已由当前用户重置。| 管理员名称 = '<名称>', 当前用户 = '<名称>'。

错误类型：
安全

用户已从用户组转移。| 用户 = '<名称>', 旧组 = '<名称>', 新组 '<name>'。

错误类型：
安全

用户组已创建。| 组 = '<名称>'。

错误类型：
安全

用户已添加到用户组。| 用户 = '<名称>', 组 = '<名称>'。

错误类型：
安全

用户信息已替换为导入内容。| 导入的文件 = '<absolute file path>'。

错误类型：
安全

用户组已重命名。| 旧名称 = '<名称>', 新名称 = '<名称>'。

错误类型：
安全

用户组的权限定义已更改。| 组 = '<名称>'。

错误类型：
安全

用户已重命名。| 旧名称 = '<名称>', 新名称 = '<名称>'。

错误类型：
安全

用户已禁用。| 用户 = '<名称>'。

错误类型：
安全

用户组已禁用。| 组 = '<名称>'。

错误类型：
安全

用户已启用。| 用户 = '<名称>'。

错误类型：
安全

用户组已启用。| 组 = '<名称>'。

错误类型：
安全

未能重置管理员密码。| 管理员名称 = '<名称>'。

错误类型：
安全

重置管理员密码失败。当前用户不是 Windows 管理员。| 管理员名称 = '<名称>', 当前用户 = '<名称>'。

错误类型：
安全

用户密码已更改。| 用户 = '<名称>'。

错误类型：
安全

导入 CSV 标记时出现一般故障。

错误类型：
错误

尝试连接到运行时失败。| 运行时主机地址 = '<host address>', 用户 = '<name>', 原因 = '<reason>'。

错误类型：
错误

用户信息无效或缺失。

错误类型：
错误

用户权限不足以替换运行时项目。

错误类型：
错误

运行时项目更新失败。

错误类型：
错误

未能检索运行时项目。

错误类型：
错误

无法替换通道上的设备，因为该设备有活动的引用计数。| 通道 = '<名称>'。

错误类型：
错误

未能替换通道上的现有自动生成设备，删除失败。| 通道 = '<名称>'。

错误类型：
错误

通道不再有效，可能在等待用户输入时被移除。| 通道 = '<名称>'。

错误类型：
错误

未加载设备驱动程序 DLL。

错误类型：
错误

未找到或无法加载设备驱动程序。| 驱动程序 = '<名称>'。

错误类型：
错误

导入 CSV 数据时出错。\n\n读取标识记录时字段缓冲区溢出。

错误类型：
错误

导入 CSV 数据时出错。\n\n无法识别字段名称。| 字段 = '<名称>'。

错误类型：
错误

导入 CSV 数据时出错。\n\n字段名称重复。| 字段 = '<名称>'。

错误类型：
错误

导入 CSV 数据时出错。\n\n字段标识记录缺失。

错误类型：
错误

导入 CSV 记录时出错。\n\n字段缓冲区溢出。| 记录索引 = '<number>'。

错误类型：
错误

导入 CSV 记录时出错。\n\n插入失败。| 记录索引 = '<number>', 记录名称 = '<name>'。

错误类型：
错误

无法启动应用程序。| 应用程序 = '<路径>', 操作系统错误 = '<code>'。

错误类型：
错误

导入 CSV 记录时出错。\n\n“映射到”标记地址对于此项目无效。| 记录索引 = '<number>', 标记地址 = '<address>'。

错误类型：
错误

导入 CSV 记录时出错。\n\n别名无效。名称不能包含双引号或以下划线开头。| 记录索引 = '<number>'。

错误类型：
错误

无效的 XML 文档:

错误类型：
错误

重命名失败。已存在具有该名称的对象。| 建议名称 = '<名称>'。

错误类型：
错误

未能启动通道诊断。

错误类型：
错误

重命名失败。名称不能包含句点、双引号或以下划线开头。| 建议名称 = '<名称>'。

错误类型：
错误

与远程运行时同步失败。

错误类型：
错误

导入 CSV 记录时出错。标记名称无效。| 记录索引 = '<number>', 标记名称 = '<name>'。

错误类型：
警告

导出 CSV 记录时出错。标记或组名称超出最大名称长度。| 记录索引 = '<number>', 最大名称长度 (字符数) = '<number>'。

错误类型：
警告

导入 CSV 记录时出错。地址缺失。| 记录索引 = '<number>'。

错误类型：
警告

导入 CSV 记录时出错。标记组名称无效。| 记录索引 = '<index>', 组名称 = '<name>'。

错误类型：
警告

由于存在启用的连接，已忽略关闭请求。| 启用的连接 = '<count>'。

错误类型：
警告

未能保存嵌入的相关文件。| 文件 = '<路径>'。

错误类型：
警告

配置实用程序不能与第三方配置应用程序同时运行。请关闭这两个程序，然后只打开要使用的程序。| 产品 = '<名称>'。

错误类型：
警告

正在打开项目。| 项目 = '<名称>'。

错误类型：
信息化

正在关闭项目。| 项目 = '<名称>'。

错误类型：
信息化

虚拟网络模式已更改。这会影响所有通道和虚拟网络。有关虚拟网络模式的详细信息，请参阅帮助。| 新模式 = '<mode>'。

错误类型：
信息化

正在开始对通道进行设备发现。| 通道 = '<名称>'。

错误类型：
信息化

通道的设备发现已完成。| 通道 = '<名称>', 发现的设备 = '<计数>'。

错误类型：
信息化

通道的设备发现已取消。| 通道 = '<名称>'。

错误类型：
信息化

通道的设备发现已取消。| 通道 = '<名称>', 发现的设备 = '<计数>'。

错误类型：
信息化

无法开始对通道进行设备发现。| 通道 = '<名称>'。

错误类型：
信息化

正在关闭以执行安装。

错误类型：
信息化

运行时项目已重置。

错误类型：
信息化

运行时项目已替换。| 新项目 = '<路径>'。

错误类型：
信息化

未连接到事件记录器服务。

错误类型：
安全

尝试添加项 '<名称>' 失败。

错误类型：
错误

未加载设备驱动程序 DLL。

错误类型：
错误

无效的项目文件: '<名称>'。

错误类型：
错误

无法打开项目文件: '<名称>'。

错误类型：
错误

替换项目的请求被拒绝，因为它与正在使用的项目相同: '<名称>'。

错误类型：
错误

文件名不能覆盖现有文件: '<name>'。

错误类型：
错误

文件名不能为空白。

错误类型：
错误

文件名的格式必须是 <subdir>/<name>.{json,opf,sopf}

错误类型：
错误

文件名包含一个或多个无效字符。

错误类型：
错误

在启用项目文件加密的情况下保存项目文件，因为不支持 .OPF 文件类型。支持的文件类型为 .SOPF 和 .JSON。

错误类型：
错误

在禁用项目文件加密的情况下保存项目文件，因为不支持 .SOPF 文件类型。支持的文件类型为 .OPF 和 .JSON。

错误类型：
错误

添加对象到 '<名称>' 失败: <原因>。

错误类型：
警告

移动对象 '<名称>' 失败: <原因>。

错误类型：
警告

更新对象 '<名称>' 失败: <原因>。

错误类型：
警告

删除对象 '<名称>' 失败: <原因>。

错误类型：
警告

无法加载启动项目 '<名称>': <原因>。

错误类型：
警告

未能更新启动项目 '<名称>': <原因>。

错误类型：
警告

运行时项目已替换为定义的启动项目。运行时项目将在下次重新启动时从 '<名称>' 恢复。

错误类型：

警告

正在忽略用户定义的启动项目，因为配置会话处于活动状态。

错误类型：
警告

读取只读项引用 '<名称>' 请求被拒绝。

错误类型：
警告

无法写入项 '<名称>'。

错误类型：
警告

对项 '<名称>' 的写入请求失败。写入数据类型 '<type>' 无法转换为标记数据类型 '<type>'。

错误类型：
警告

对项 '<名称>' 的写入请求失败。缩放写入数据时出错。

错误类型：
警告

对项引用 '<名称>' 的写入请求被拒绝，因为其所属的设备被禁用。

错误类型：
警告

已成功配置 <名称> 作为系统服务运行。

错误类型：
信息化

已成功从服务控制管理器数据库移除 <名称>。

错误类型：
信息化

运行时重新初始化已启动。

错误类型：
信息化

运行时重新初始化已完成。

错误类型：
信息化

已更新启动项目 '<名称>'。

错误类型：
信息化

运行时服务已启动。

错误类型：
信息化

运行时进程已启动。

错误类型：
信息化

运行时正在退出。

错误类型：
信息化

运行时关闭完成。

错误类型：
信息化

关闭以执行安装。

错误类型：
信息化

运行时项目从 '<名称>' 替换。

错误类型：
信息化

应用程序数据目录缺失。

错误类型：
信息化

运行时项目已另存为 '<名称>'。

错误类型：
信息化

运行时项目已替换。

错误类型：
信息化

配置会话由 <名称> (<名称>) 启动。

错误类型：
安全

分配到 <名称> 的配置会话已结束。

错误类型：
安全

分配到 <名称> 的配置会话已升级为写入权限。

错误类型：
安全

分配到 <名称> 的配置会话已降级为只读。

错误类型：
安全

应用于配置会话的权限更改已分配到 <名称>。

错误类型：
安全

Failed to start Script Engine server. Socket error occurred binding to local port. | Error = <error>, Details = '<information>'.

错误类型：
错误

可能的原因：
端口与另一个应用程序冲突。

可能的解决方案：
Use the server administration settings to update the Script Engine port.

An unhandled exception was thrown from the script. | Function = '<function>', error = '<error>'.

错误类型：
错误

可能的原因：
An exception was thrown from the script.

可能的解决方案：
Correct the condition that lead to the exception, or update the script logic.

Script Engine service stopping.

错误类型：
信息化

Script Engine service starting.

错误类型：
信息化

Config API SSL 证书包含错误签名。

错误类型：
错误

Config API 无法加载 SSL 证书。

错误类型：
错误

无法启动 Config API 服务。绑定到端口时可能出现了问题。

错误类型：
错误

可能的原因：
The HTTP or HTTPS port specified in the Config API settings is already bound by another application.

可能的解决方案：
Change the configuration of the Config API or blocking application to use a different port, or stop the application blocking the port.

Config API SSL 证书已过期。

错误类型：
警告

Config API SSL 证书是自签名证书。

错误类型：
警告

Configuration API started without SSL on port <port number>.

错误类型：
信息化

Configuration API started with SSL on port <port number>.

错误类型：
信息化

OPC .NET 服务器无法启动。有关详细信息，请查看 Windows 应用程序事件日志。另请确保 .NET 3.5 Framework 已安装。| OS 错误 = '<错误原因>'。

错误类型：
错误

OPC .NET 服务器无法启动，因为尚未安装。请重新运行安装程序。

错误类型：
错误

尝试启动 OPC .NET 服务器超时。请验证服务器是否正在通过使用 OPC .NET Configuration Manager 运行。

错误类型：
警告

缺少服务器实例证书 '<证书位置>'。请使用 OPC UA Configuration Manager 重新颁发证书。

错误类型：
错误

无法导入服务器实例证书: '<证书位置>'。请使用 OPC UA Configuration Manager 重新颁发证书。

错误类型：
错误

UA 服务器证书过期。请使用 OPC UA Configuration Manager 重新颁发证书。

错误类型：
错误

侦听客户端连接时发生套接字错误。| 端点 URL = '<端点 URL>'，错误 = <错误代码>，详细信息 = '<说明>'。

错误类型：
错误

UA 服务器无法注册到 UA Discovery Server。| 端点 URL: '<端点 url>'。

错误类型：
错误

UA 服务器无法从 UA Discovery Server 取消注册。| 端点 URL: '<端点 url>'。

错误类型：
警告

UA 服务器成功注册到 UA Discovery Server。| 端点 URL: '<端点 url>'。

错误类型：
信息化

UA 服务器成功从 UA Discovery Server 取消注册。| 端点 URL: '<端点 url>'。

错误类型：
信息化

无法启用对此服务器的 iFIX PDB 支持。| OS 错误 = '<错误>'。

错误类型：
信息化

无法启用对此服务器的 iFIX PDB 支持。| OS 错误 = '<错误>'。

错误类型：
信息化

ReadProcessed 请求超时。| 运行时间 = <秒> (秒)。

错误类型：
错误

ReadAtTime 请求超时。| 运行时间 = <秒> (秒)。

错误类型：
错误

尝试添加 DDE 项失败。| 项 = '<项名称>'。

错误类型：
错误

DDE 客户端尝试添加主题失败。| 主题 = '<主题>'。

错误类型：
错误

可能的原因：
主题名称无效。

可能的解决方案：
查看别名映射以将引用指向有效主题。

也可以看看：
别名映射

无法写入项。| 项 = '<项名称>'。

错误类型：
警告

指定的区域无效。无法设置订阅筛选器。| 区域 = '<区域名称>'。

错误类型：
错误

指定的源无效。无法设置订阅筛选器。| 源 = '<源名称>'。

错误类型：
错误

Config API SSL 证书包含错误签名。

错误类型：
错误

Config API 无法加载 SSL 证书。

错误类型：
错误

无法启动 Config API 服务。绑定到端口时可能出现了问题。

错误类型：
错误

可能的原因：
The HTTP or HTTPS port specified in the Config API settings is already bound by another application.

可能的解决方案：
Change the configuration of the Config API or blocking application to use a different port, or stop the application blocking the port.

Config API SSL 证书已过期。

错误类型：
警告

Config API SSL 证书是自签名证书。

错误类型：
警告

Configuration API started without SSL on port <port number>.

错误类型：
信息化

Configuration API started with SSL on port <port number>.

错误类型：
信息化

与 ThingWorx 连接失败。| 平台 <主机:端口资源>，错误: <原因>。

错误类型：
错误

可能的原因：
无法建立与 ThingWorx Platform 的连接。

可能的解决方案：

1. 验证主机、端口、资源和应用程序密钥是否均有效且正确。
2. 验证主机是否可在 ThingWorx Platform 上访问 Composer。
3. 验证使用自签名证书或未加密时，是否启用了正确的证书设置。

添加项时出错。| 项名称 = '<项名称>'。

错误类型：
错误

可能的原因：
无法向服务器添加项 <标记名称> 以用于扫描。

可能的解决方案：

1. 验证标记是否在有效的通道和设备上。
2. 验证标记是否可使用另一客户端 (如 QuickClient) 读取。

无法在平台上触发自动绑定完成事件。

错误类型：
错误

可能的原因：
ThingWorx 连接在自动绑定进程完成前终止。

可能的解决方案：
等到所有自动绑定均已完成后，重新初始化或更改 ThingWorx 项目属性。

与 ThingWorx 连接失败，原因未知。| 平台 = <主机:端口资源>，错误= <错误>。

错误类型：
错误

可能的原因：
与 ThingWorx Platform 连接失败。

可能的解决方案：

1. 验证主机、端口、资源和应用程序密钥是否均有效且正确。
2. 验证主机是否可在 ThingWorx Platform 上访问 Composer。
3. 验证使用自签名证书或未加密时，是否启用了正确的证书设置。
4. 请联系技术支持，并提供错误代码和应用程序报告。

因连接缓冲区中空间不足，一或多个值更改更新丢失。| 丢失更新数 = <计数>。

错误类型：
错误

可能的原因：

数据正在丢失，因为 ThingWorx Platform 不可用或实例收集的数据过多。

可能的解决方案：

1. 验证 ThingWorx Platform 上的部分数据是否正在更新以及该平台是否可访问。
2. 降低标记扫描速率以减少移入 ThingWorx Platform 的数据。

发布项失败。多维数组不受支持。| 项名称 = '%s'。

错误类型：

错误

可能的原因：

项 <ItemName> 引用的标记的数据是多维数组。

可能的解决方案：

修改项以引用具有受支持数据类型的标记。

由于磁盘已满，“存储并转发”数据存储无法存储数据。

错误类型：

错误

可能的原因：

用于存储更新的磁盘已填满至 500 MiB 上限。

可能的解决方案：

1. 释放磁盘上用于存储更新的空间。
2. 使用 the _DeleteStoredData 系统标记删除数据存储中的数据。
3. 以更大的磁盘替换用于存储数据的磁盘。

已达到“存储并转发”数据存储大小限制。

错误类型：

错误

可能的原因：

ThingWorx Interface 无法在更新生成后立即将更新发送至平台。

可能的解决方案：

1. 验证 ThingWorx Interface 可以连接到 ThingWorx Platform。
2. 降低 ThingWorx Interface 收集更新的频率。

与 ThingWorx 的连接已关闭。| 平台 = <主机:端口资源>。

错误类型：

警告

可能的原因：

连接已关闭。服务已停止或接口无法再访问平台。

可能的解决方案：

1. 验证在项目属性中本机接口是否为启用状态。
2. 验证主机是否可在 ThingWorx Platform 上访问 Composer。

无法自动绑定属性。| 名称 = '<属性名称>'。

错误类型：
警告

可能的原因：
在此“事物”下具有此名称的属性已存在。

可能的解决方案：

1. 检查属性以查看数据是否为最新。
2. 如果数据不是最新的，请删除“事物”下的属性，然后再次运行 AddItem 服务。

无法重启“事物”。| 名称 = '<事物名称>'。

错误类型：
警告

可能的原因：
AddItem 服务完成时，对“事物”调用重启服务。以使 Composer 可以显示更改。即使已显示此错误，但仍会将数据更改发送至平台。

可能的解决方案：
重新启动 Composer 以重启“事物”。

写入属性失败。| 属性名称 = '<名称>'，原因 = <原因>。

错误类型：
警告

可能的原因：
因转换问题而无法写入标记。

可能的解决方案：

1. 验证 KEPServerEX 以及 ThingWorx Platform 中标记的数据类型是否正确且一致。
2. 验证要写入的值是否在数据类型的正确范围内。

ThingWorx 请求添加项失败。该项已添加。| 项名称 = '<名称>'。

错误类型：
警告

可能的原因：
已将标记添加至此“事物”。

可能的解决方案：

1. 检查属性以查看数据是否为最新。
2. 如果数据不是最新的，请删除“事物”下的属性，然后再次运行 AddItem 服务。

ThingWorx 请求移除项失败。该项不存在。| 项名称 = '<名称>'。

错误类型：
警告

可能的原因：
已从“事物”中移除标记或不存在此类标记。

可能的解决方案：
如果标记仍在“事物”的属性下显示，请在 ThingWorx Composer 中删除该属性。

已将服务器配置为针对每次扫描发送更新，但将一个或多个属性的推送类型设置为仅在值更改时推送。| 计数 = <计数>。

错误类型：
警告

可能的原因：
针对某些项，将 ThingWorx Platform 中的推送类型设置为仅更改。此推送类型仅会在数据值更改时更新平台上的数据。

可能的解决方案：
要使用“每次扫描时发送”选项，请将此值设为“始终”。

将一个或多个属性的推送类型设置为从不向平台推送更新。| 计数 = <计数>。

错误类型：
警告

可能的原因：
针对某些项，将 ThingWorx Platform 中的推送类型设置为“从不”，这可防止在平台上自动更新任何数据更改。

可能的解决方案：
如果这不是所需行为，请在 ThingWorx Platform 中更改推送类型。

ThingWorx 请求移除项失败。该项已绑定且强制标志为 false。| 项名称 = '<名称>'。

错误类型：
警告

可能的原因：
RemoveItems 服务无法移除项，因为该项与属性绑定且“强制标志”未设为 True。

可能的解决方案：
重新运行服务，明确调用 ForceRemove 标志为 True。

写入属性失败。| 事物名称 = '<名称>'，属性名称 = '<名称>'，原因 = <原因>。

错误类型：
警告

可能的原因：
因转换问题而无法写入标记。

可能的解决方案：

1. 验证 KEPServerEX 以及 ThingWorx Platform 中标记的数据类型是否正确且一致。
2. 验证要写入的值是否在数据类型的正确范围内。

将属性更新推送到事物时出错。| 事物名称 = '<名称>'。

错误类型：
警告

可能的原因：
命名事物的属性更新没有成功发布到平台。

可能的解决方案：
请查阅平台日志，找出属性更新失败的原因，例如: 权限问题。

无法连接或附加到“存储并转发”数据存储。正在使用内存存储。| 内存存储大小 (更新数) = <计数>。

错误类型：
警告

可能的原因：

1. “存储并转发”服务未运行。
2. 服务不能访问指定的存储目录。
3. 存在端口冲突，导致“存储并转发”服务无法接受连接。

可能的解决方案：

1. 重新启动服务器运行时。
2. 验证“存储并转发”服务可以访问指定的存储位置。
3. 通过在“服务器管理”中为“存储并转发”配置新端口来解决端口冲突。

由于文件 IO 错误或数据存储损坏，“存储并转发”数据存储已重置。

错误类型：
警告

可能的原因：

1. 数据存储已被用户或其他程序损坏。
2. 数据存储由于硬件错误而损坏。
3. 尝试从磁盘读取数据时出错，可能因为出现硬件问题。

可能的解决方案：

1. 通过用户访问控制来限制可以访问数据存储位置的用户。
2. 将数据存储转移到另一个磁盘。

无法应用由 ThingWorx Platform 引起的设置更改。权限被拒绝。| 用户 = '<用户名>'。

错误类型：

警告

可能的原因：
用户组 "ThingWorx Interface Users" 的权限 "Project Modification:Servermain.Project" 被设置为 "拒绝"。

可能的解决方案：
将用户组 "ThingWorx Interface Users" 的权限 "Project Modification:Servermain.Project" 设置为 "允许"。

面向 ThingWorx Platform 的配置传输失败。

错误类型：
警告

面向 ThingWorx Platform 的配置传输失败。| 原因 = '<原因>'

错误类型：
警告

可能的原因：

1. 有关详细信息，请查看原因说明。
2. 运行时项目已锁定，因为有用户正在编辑它。
3. 此 ThingWorx Interface 用户帐户没有足够的权限执行该操作。

无法删除“存储并转发”数据存储中已存储的更新。

错误类型：
警告

可能的原因：
硬件或操作系统错误导致操作无法完成。

可能的解决方案：
重新启动机器并重试。

来自 ThingWorx Platform 的配置传输失败。

错误类型：
警告

来自 ThingWorx Platform 的配置传输失败。| 原因 = '<原因>'

错误类型：
警告

可能的原因：

1. 有关详细信息，请查看原因说明。
2. 运行时项目已锁定，因为有用户正在编辑它。
3. 此 ThingWorx Interface 用户帐户没有足够的权限执行该操作。

已连接至 ThingWorx。| 平台 = <主机:端口资源>，事物名称 = '<名称>'。

错误类型：
信息化

可能的原因：
建立了与 ThingWorx Platform 的连接。

因平台引起了项目设置更改而正在重新初始化 ThingWorx 连接。

错误类型：
信息化

可能的原因：
使用 SetConfiguration 服务时，此消息会通知操作员查看 KEPServerEX 事件日志了解发生了更改。

因接口关闭或重新初始化而正在放弃待处理的自动绑定。| 计数 = <计数>。

错误类型：
信息化

可能的原因：
在调用 AddItems 服务而正在处理自动绑定期间，请求了服务器关机或启动。

可能的解决方案：
未自动绑定的任何项都需要在 ThingWorx Composer 中手动创建和绑定。

已处理一个或多个自动绑定请求。| 计数 = <计数>。

错误类型：
信息化

可能的原因：
部分 AddItems 服务是自动绑定操作。此操作可能会比实际添加项花费更多的时间。此消息会提醒操作员已自动绑定多少项。

因 Configuration API 引起了项目设置更改而正在重新初始化 ThingWorx 连接。

错误类型：
信息化

可能的原因：
使用 Configuration API 时，此消息会通知操作员查看 KEPServerEX 事件日志了解发生了更改。

已恢复将属性更新推送到事物: 错误条件已解决。| 事物名称 = '<名称>'。

错误类型：
信息化

来自 ThingWorx 的配置传输已启动。

错误类型：
信息化

来自 ThingWorx 的配置传输已中止。

错误类型：
信息化

已初始化“存储并转发”数据存储。| 数据存储位置: '<位置>'。

错误类型：
信息化

可能的原因：
ThingWorx Native Interface 配置为使用“存储并转发”。

已成功从“存储并转发”数据存储中删除已存储的数据。

错误类型：
信息化

可能的原因：
客户端已使用 _DeleteStoredData 系统标记来删除“存储并转发”数据存储中缓存的 ThingWorx Interface 数据。

“存储并转发”模式已更改。| 转发模式 = '<模式>'。

错误类型：
信息化

可能的原因：
_ForwardMode 系统标记由连接的客户端写入，并且写入值导致设置更改。

已初始化“存储并转发”数据存储。| 转发模式 = '<模式>' | 数据存储位置 = '<位置>'。

错误类型：
信息化

可能的原因：
ThingWorx Native Interface 配置为使用“存储并转发”。

由于数据存储路径无效，连接到数据存储时出错。| 路径 = '<路径>'。

错误类型：
错误

可能的原因：
使用“存储并转发”的组件所指定的路径无效。有关详细信息，请查看消息正文中的组件文档和验证错误。

可能的解决方案：
更正消息中指明的错误。

无法启动“存储并转发”服务器。绑定到本地端口时出现 socket 错误。| 错误 = <错误>, 详细信息 = '<信息>'。

错误类型：
错误

可能的原因：
端口与另一个应用程序冲突。

可能的解决方案：
使用服务器管理设置来更新“存储并转发”端口。

“存储并转发”服务正在停止。

错误类型：
信息化

“存储并转发”服务正在启动。

错误类型：
信息化

连接数据存储时文件损坏，已重新创建数据存储。| 数据存储路径 = '<路径>'。

错误类型：
信息化

可能的原因：
数据存储所使用的文件已被系统、其他应用程序或用户损坏。

可能的解决方案：

1. 已自动替换旧的数据存储，无需用户操作。
2. 如果此问题重复出现，请考虑将数据存储目录更改为其他应用程序或用户无法访问的位置。

配置更改导致数据存储被覆盖。| 数据存储路径 = '<路径>'。

错误类型：
信息化

可能的原因：
数据存储大小参数已更改。

注意：
更改数据存储大小会导致所有数据存储的文件被重新创建。除非由于与 ThingWorx Platform 断开连接而将数据存储在数据存储，否则数据通常不会丢失。

无法连接到现有的数据存储，因为该数据存储由较早版本的服务器创建。已重新创建数据存储。| 数据存储路径 = '<路径>'。

错误类型：
信息化

可能的原因：
服务器已升级到使用较新数据存储格式的版本。

可能的解决方案：
旧的数据存储已替换为新版本的数据存储，无需用户操作。

由于数据存储路径无效，连接到数据存储时出错。| 路径 = '<路径>'。

错误类型：
错误

可能的原因：
使用“存储并转发”的组件所指定的路径无效。有关详细信息，请查看消息正文中的组件文档和验证错误。

可能的解决方案：
更正消息中指明的错误。

无法启动“存储并转发”服务器。绑定到本地端口时出现 socket 错误。| 错误 = <错误>, 详细信息 = '<信息>'。

错误类型：
错误

可能的原因：
端口与另一个应用程序冲突。

可能的解决方案：
使用服务器管理设置来更新“存储并转发”端口。

“存储并转发”服务正在停止。

错误类型：
信息化

“存储并转发”服务正在启动。

错误类型：
信息化

连接数据存储时文件损坏，已重新创建数据存储。| 数据存储路径 = '<路径>'。

错误类型：
信息化

可能的原因：
数据存储所使用的文件已被系统、其他应用程序或用户损坏。

可能的解决方案：

1. 已自动替换旧的数据存储，无需用户操作。
2. 如果此问题重复出现，请考虑将数据存储目录更改为其他应用程序或用户无法访问的位置。

配置更改导致数据存储被覆盖。| 数据存储路径 = '<路径>'。

错误类型：
信息化

可能的原因：
数据存储大小参数已更改。

注意：
更改数据存储大小会导致所有数据存储的文件被重新创建。除非由于与 ThingWorx Platform 断开连接而将数据存储在数据存储，否则数据通常不会丢失。

无法连接到现有的数据存储，因为该数据存储由较早版本的服务器创建。已重新创建数据存储。| 数据存储路径 = '<路径>'。

错误类型：
信息化

可能的原因：
服务器已升级到使用较新数据存储格式的版本。

可能的解决方案：
旧的数据存储已替换为新版本的数据存储，无需用户操作。

另一应用程序正在使用 Com 端口。| 端口 = '<端口>'。

错误类型：
错误

可能的原因：
另一应用程序正在使用分配给设备的串行端口。

可能的解决方案：

1. 验证是否已将正确端口分配给通道。
2. 验证是否仅当前项目的一个副本正在运行。

无法使用指定参数配置 com 端口。| 端口 = COM<数字>，OS 错误 = <错误>。

错误类型：
错误

可能的原因：
指定 COM 端口的串行参数无效。

可能的解决方案：
验证串行参数，并进行任何必要的更改。

驱动程序初始化失败。

错误类型：
错误

无法创建串行 I/O 线程。

错误类型：
错误

可能的原因：
服务器进程没有可以创建新线程的资源。

可能的解决方案：
每个标记组消耗一个线程。一个进程的一般限制约为 2000 个线程。请减少项目中的标记组数量。

Com 端口不存在。| 端口 = '<端口>'。

错误类型：
错误

可能的原因：
目标计算机上不存在指定的 COM 端口。

可能的解决方案：
验证是否选择了正确的 COM 端口。

打开 com 端口时出错。| 端口 = '<端口>'，OS 错误 = <错误>。

错误类型：
错误

可能的原因：
由于目标计算机上存在内部硬件或软件的问题，无法打开指定的 COM 端口。

可能的解决方案：
验证 COM 端口是否正常工作，以及是否由其他应用程序访问。

连接失败。无法绑定到适配器。| 适配器 = '<名称>'。

错误类型：
错误

可能的原因：

由于在系统设备列表中无法找到指定的网络适配器，因此无法绑定至其中以进行通信。当将项目从一台 PC 移动到另一台 PC (以及当项目指定了某个网络适配器，而不是使用默认适配器) 时，即会发生此问题。服务器会恢复到默认适配器。

可能的解决方案：

将网络适配器属性更改为“默认”(或选择新的适配器)，保存项目，然后重试。

Winsock 关闭失败。| OS 错误 = <错误>。

错误类型：

错误

Winsock 初始化失败。| OS 错误 = <错误>。

错误类型：

错误

可能的解决方案：

1. 底层网络子系统尚未准备好进行网络通信。请等待几秒钟，然后重新启动驱动程序。
2. 已经达到 Windows 套接字实施所支持的任务数限制。请关闭一个或多个可能使用 Winsock 的应用程序，然后重新启动驱动程序。

要使用此驱动程序，必须安装 Winsock V1.1 或更高版本。

错误类型：

错误

可能的原因：

在系统上发现的 Winsock DLL 的版本号早于 1.1。

可能的解决方案：

将 Winsock 升级到 1.1 或更高版本。

绑定到本地端口时发生套接字错误。| 错误 = <错误>，详细信息 = '<信息>'。

错误类型：

错误

设备未响应。

错误类型：

警告

可能的原因：

1. 设备与主机 PC 之间的连接断开。
2. 连接的通信参数错误。
3. 可能为指定设备分配了不正确的设备 ID。
4. 接受设备响应所消耗的时间超出了“请求超时”设备设置允许的时间。

可能的解决方案：

1. 验证 PC 与 PLC 设备之间的电缆连接。
2. 验证指定通信参数是否与设备通信参数匹配。

3. 验证指定设备的设备 ID 是否与实际设备的设备 ID 相匹配。
4. 请增加“请求超时”设置以便处理完整的响应。

设备未响应。| ID = '<设备>'。

错误类型：
警告

可能的原因：

1. 设备与主机 PC 之间的网络连接断开。
2. 为设备和驱动程序配置的通信参数不匹配。
3. 接受设备响应所消耗的时间超出了“请求超时”设备设置允许的时间。

可能的解决方案：

1. 验证 PC 与 PLC 设备之间的电缆连接。
2. 验证指定通信参数是否与设备通信参数匹配。
3. 请增加“请求超时”设置以便处理完整的响应。

通道上出现串行通信错误。| 错误掩码 = <掩码>。

错误类型：
警告

可能的原因：

1. 设备与主机 PC 之间的串行连接断开。
2. 串行连接通信参数错误。

可能的解决方案：

1. 调查错误掩码和相关信息。
2. 验证 PC 与 PLC 设备之间的电缆连接。
3. 验证指定的通信参数是否与设备的通信参数相匹配。

也可以看看：
错误掩码

写入标记 <设备名称>.<地址> 时检测到无效的数组大小。

错误类型：
警告

可能的原因：
客户端在更新之前尝试进行写操作。

可能的解决方案：
在尝试写操作之前读取数组。

无法写入设备上的地址。| 地址 = '<地址>'。

错误类型：

警告

可能的原因：

1. 设备与主机 PC 之间的连接断开。
2. 用于连接的通信参数不正确。
3. 可能为指定设备分配了不正确的设备 ID。

可能的解决方案：

1. 验证 PC 与 PLC 设备之间的电缆连接。
2. 验证指定的通信参数是否与设备的通信参数相匹配。
3. 验证分配给指定设备的设备 ID 是否与实际设备的设备 ID 相匹配。

驱动程序在处理标记期间可能不会更改此页上的项。

错误类型：

警告

可能的原因：

当数据客户端连接到服务器并从通道/设备接收数据时，尝试更改了通道或设备配置。

可能的解决方案：

在进行更改前，请从服务器断开所有数据客户端。

设备上的指定地址无效。| 无效地址 = '<地址>'。

错误类型：

警告

可能的原因：

为标记地址分配了无效地址。

可能的解决方案：

修改客户端应用程序中请求的地址。

地址 '<地址>' 在设备 '<名称>' 上无效。

错误类型：

警告

驱动程序在处理标记期间可能不会更改此属性。

错误类型：

警告

无法写入设备 '<名称>' 上的地址 '<地址>'。

错误类型：

警告

可能的原因：

1. 设备与主机 PC 之间的连接断开。
2. 用于连接的通信参数不正确。
3. 可能为指定设备分配了不正确的设备 ID。

可能的解决方案：

1. 验证 PC 与 PLC 设备之间的电缆连接。
2. 验证指定的通信参数是否与设备的通信参数相匹配。
3. 验证分配给指定设备的设备 ID 是否与实际设备的设备 ID 相匹配。

连接时发生套接字错误。| 错误 = <错误>，详细信息 = '<信息>'。

错误类型：
警告

可能的原因：
在指定的套接字运行时，与设备的通信失败。

可能的解决方案：
遵照错误和详细信息的指导，该指导解释了错误发生的原因并在适用时提出了修复方法。

接收数据时发生套接字错误。| 错误 = <错误>，详细信息 = '<信息>'。

错误类型：
警告

可能的原因：
在指定的套接字运行时，与设备的通信失败。

可能的解决方案：
遵照错误和详细信息的指导，该指导解释了错误发生的原因并在适用时提出了修复方法。

发送数据时发生套接字错误。| 错误 = <错误>，详细信息 = '<信息>'。

错误类型：
警告

可能的原因：
在指定的套接字运行时，与设备的通信失败。

可能的解决方案：
遵照错误和详细信息的指导，该指导解释了错误发生的原因并在适用时提出了修复方法。

检查可读性时发生套接字错误。| 错误 = <错误>，详细信息 = '<信息>'。

错误类型：
警告

可能的原因：
在指定的套接字运行时，与设备的通信失败。

可能的解决方案：
遵照错误和详细信息的指导，该指导解释了错误发生的原因并在适用时提出了修复方法。

检查可写性时发生套接字错误。| 错误 = <错误>，详细信息 = '<信息>'。

错误类型：
警告

可能的原因：
在指定的套接字运行时，与设备的通信失败。

可能的解决方案：
遵照错误和详细信息的指导，该指导解释了错误发生的原因并在适用时提出了修复方法。

%s |

错误类型：
信息化

<名称> 设备驱动程序 '<名称>'

错误类型：
信息化

功能 '<名称>' 未获许可，因而无法使用。

错误类型：
错误

可能的原因：

1. 尚未购买及许可产品的指定功能。
2. 产品许可证已移除或受信任的存储已损坏。

可能的解决方案：

1. 下载或安装软件功能并购买许可证。
2. 有关激活紧急许可证的说明，请参阅许可用户手册。
3. 如需帮助，请联系销售或支持代表。

也可以看看：
许可证实用工具帮助

无法加载许可证界面，可能是因为缺失第三方依赖项。仅在“限时”模式下运行。

错误类型：
错误

可能的原因：
系统中所需的一个或多个 OEM 许可组件缺失。

可能的解决方案：
如需帮助，请联系销售或支持代表。

也可以看看：
许可证实用工具帮助

“限时”模式已过期。

错误类型：
警告

可能的原因：

1. “限时”模式期间，尚未购买及许可产品。
2. 服务器在“限时”模式下启动，指定时间仍处于“限时”模式内。

可能的解决方案：

1. 评估服务器时，无需进行任何操作。
2. 如果这是一台生产计算器，则请在“限时”模式过期前激活已安装组件的产品许可证。
3. 为将要使用的所有产品功能购买许可证。
4. 如需帮助，请联系销售或支持代表。

也可以看看：

许可证实用工具帮助

对于精简版 '<数字>' 许可证，已超出最大设备计数。编辑项目并重新启动服务器。

错误类型：

警告

可能的原因：

指定驱动程序已使用精简版许可证激活，这限制了可配置的设备数。

可能的解决方案：

1. 验证许可证授权的设备数并更正项目设计以减少设备计数。
2. 如果需要更多设备或精简版激活不正确，请联系销售代表咨询升级许可证的相关事宜以支持更多设备。

也可以看看：

许可证实用工具帮助

对于精简版 '<数字>' 许可证，已超出最大运行时标记计数。编辑客户端项目并重新启动服务器。

错误类型：

警告

可能的原因：

指定驱动程序已使用精简版许可证激活，这限制了可配置的标记数。

可能的解决方案：

1. 验证许可证授权的标记数并更正项目设计以减少标记计数。
2. 如果需要更多标记或精简版激活不正确，请联系销售代表咨询升级许可证的相关事宜以支持更多标记。

也可以看看：

许可证实用工具帮助

在功能 '<名称>' 上，已超出 <最大计数> 的类型 <数字类型 ID> 限制。

错误类型：

警告

可能的原因：

已安装的功能许可证限制了可配置的指定类型的项目数。

可能的解决方案：

1. 请联系客户解决方案以确定应减少哪种对象类型数以保持处于许可证的限制范围内。
2. 如果需要更多项目，请联系销售代表咨询升级许可证的相关事宜。

也可以看看：
许可证实用工具帮助

在功能 '<名称>' 上，已超出 <最大计数> 的 <对象类型名称> 限制。

错误类型：
警告

可能的原因：
已安装的功能许可证限制了可配置的指定类型的项目数。

可能的解决方案：

1. 验证许可证授权的数量并更正项目设计以仅使用所授权的项目数。
2. 如果需要更多项目，请联系销售代表咨询升级许可证的相关事宜。

也可以看看：
许可证实用工具帮助

必须启用 FlexNet 许可服务才能处理许可证。启用服务失败将导致“限时”模式启动。

错误类型：
警告

可能的原因：
已尝试验证许可证一次，但 FlexNet 许可服务处于禁用状态。

可能的解决方案：
使用 Windows 服务控制管理器以启用 FlexNet 许可服务，这需要重新启动运行时。

也可以看看：
许可证实用工具帮助

<名称> 功能许可证已移除。服务器将进入“限时”模式，除非在宽限期过期之前已恢复许可证。

错误类型：
警告

可能的原因：
功能许可证已删除，移动至另一台计算机，硬件密钥已移除，或受信任的存储已损坏。

可能的解决方案：

1. 有关激活紧急许可证的说明，请参阅许可用户手册。
2. 如需帮助，请联系销售或支持代表。

也可以看看：
许可证实用工具帮助

无法访问功能 <名称> 的许可证 [错误=<代码>]，必须重新将其激活。

错误类型：
警告

可能的原因：
受信任的存储已损坏，可能是因为系统升级。

可能的解决方案：

1. 有关激活紧急许可证的说明，请参阅许可用户手册。
2. 如需帮助，请联系销售或支持代表。

也可以看看：
许可证实用工具帮助

功能 <名称> 存在时间限制，且将于 <日期/时间> 过期。

错误类型：
警告

功能 <名称> 存在时间限制，且将于 <日期/时间> 过期。

错误类型：
警告

在功能 <名称> 上，已超出对象计数限制。受时间限制的使用将于 <日期/时间> 过期。

错误类型：
警告

在 <名称> 上，已超出特征计数限制。受时间限制的使用将于 <日期/时间> 过期。

错误类型：
警告

在功能 <名称> 上，受时间限制的使用已过期。

错误类型：
警告

对于精简版 '<名称>' 驱动程序许可证，已超出最大驱动程序计数。编辑项目并重新启动服务器。

错误类型：
信息化

可能的原因：
指定驱动程序已使用精简版许可证激活，这限制了可配置的驱动程序数。

可能的解决方案：

1. 验证许可证授权的驱动程序数。更正项目以仅使用所授权的驱动程序数。
2. 如果需要更多驱动程序或精简版激活不正确，请联系销售代表咨询升级许可证的相关事宜以支持更多驱动程序。

也可以看看：

1. 事件日志 (位于服务器帮助中)
2. 许可证实用工具帮助

无法添加项目。请求的 <数字> 计数会超过 <最大计数> 的许可证限制。

错误类型：
信息化

可能的原因：
产品许可证限制了可配置的项目数。

可能的解决方案：

1. 验证许可证授权的数量并更正项目以仅使用所授权的项目数。
2. 如果需要更多项目，请联系销售代表咨询升级许可证的相关事宜。

也可以看看：
许可证实用工具帮助

要求组件 <名称> (<版本>) 版本与组件 <名称> (<版本>) 版本相匹配。

错误类型：
信息化

可能的原因：
两个已安装组件之间形成相互依赖关系的前提要求是两个版本相互匹配。

可能的解决方案：
验证组件版本，下载或安装匹配版本的组件。

也可以看看：
许可证实用工具帮助

对于精简版 '<名称>' 驱动程序许可证，已超出最大通道计数。编辑项目并重新启动服务器。

错误类型：
信息化

可能的原因：
指定驱动程序已使用精简版许可证激活，这限制了可配置的通道数。

可能的解决方案：

1. 验证许可证授权的通道数并更正项目以仅使用所授权的通道数。
2. 如果需要更多通道或精简版激活不正确，请联系销售代表咨询升级许可证的相关事宜以支持更多通道。

也可以看看：

1. 事件日志 (位于服务器帮助中)
2. 许可证实用工具帮助

%s 此刻已获许可。

错误类型：

信息化

资源

除了本用户手册之外，我们还提供各种资源来帮助客户、回答问题、提供有关具体实施的更多细节，或者帮助解决特定问题。

知识库
白皮书
连接指南
技术说明
培训课程
培训视频
Kepware 技术支持
PTC 技术支持

索引

'

'<modem>' 的拨号已中止。 182
'<modem>' 无拨号音。 182
'<product>' 驱动程序当前不支持 XML 持久存储。请使用默认文件格式进行保存。 179

%

%s | 213
%s 此刻已获许可。 218

“

“存储并转发”服务正在启动。 205, 207
“存储并转发”服务正在停止。 205, 207
“存储并转发”模式已更改。| 转发模式 = '<模式>'。 205
“导出” 88
“限时”模式已过期。 213

<

<Product> 设备驱动程序加载成功。 181
<Product> 设备驱动程序已从内存卸载。 182
<Source>
无效的以太网封装 IP '<address>'。 179
<名称> 功能许可证已移除。服务器将进入“限时”模式，除非在宽限期过期之前已恢复许可证。 215
<名称> 设备驱动程序 '<名称>' 213

A

Absolute 74
An unhandled exception was thrown from the script. | Function = '<function>', error = '<error>'. 194
AttributeServiceSet 166

C

Com 端口不存在。| 端口 = '<端口>'。 208
Config API SSL 证书包含错误签名。 194, 197
Config API SSL 证书是自签名证书。 195, 197
Config API SSL 证书已过期。 195, 197

Config API 无法加载 SSL 证书。 194, 197
Configuration API started with SSL on port <port number>. 195, 197
Configuration API started without SSL on port <port number>. 195, 197
CORS 36
CSV 88

D

DCOM 26
DDE 20
DDE 客户端尝试添加主题失败。| 主题 = '<主题>'。 196
DiscoveryServiceSet 166

F

Failed to start Script Engine server. Socket error occurred binding to local port. | Error = <error>, Details = '<information>'. 194
FastDDE/SuiteLink 21

G

GET 请求 URI 140

H

HTTP 端口 36
HTTPS 端口 36

I

ID 70
iFIX 本机接口 21
iFIX 信号调节选项 151
iFIX 应用程序的项目启动 156
IP 地址 72

J

JSON 122
JSON 响应结构 140

M

MonitoredItemServiceSet 166

O

OPC .NET 20
OPC .NET 服务器无法启动，因为尚未安装。请重新运行安装程序。 195
OPC .NET 服务器无法启动。有关详细信息，请查看 Windows 应用程序事件日志。另请确保 .NET 3.5 Framework 已安装。| OS 错误 = '<错误原因>'。 195
OPC AE 18
OPC DA 18
OPC DA 服务 166
OPC ProgID 已从 ProgID 重定向列表移除。| ProgID = '<ID>'。 185
OPC ProgID 已添加到 ProgID 重定向列表。| ProgID = '<ID>'。 185
OPC UA 19
OPC 诊断查看器 158
OPC 诊断事件 161
OPF 122
OtherServices 167

P

Password 143, 149

R

ReadAtTime 请求超时。| 运行时间 = <秒> (秒)。 196
ReadProcessed 请求超时。| 运行时间 = <秒> (秒)。 196

S

Script Engine service starting. 194
Script Engine service stopping. 194
SecureChannelServiceSet 167
SessionServiceSet 167
SOPF 122
SSL 37
SubscriptionServiceSet 167

T

TAPI 配置已更改，正在重新初始化... 181
TAPI 线路初始化失败
<code>。 177
ThingWorx 51
ThingWorx - 传统模式示例 55
ThingWorx 本机接口 21
ThingWorx 请求添加项失败。该项已添加。| 项名称 = '<名称>'。 200
ThingWorx 请求移除项失败。该项不存在。| 项名称 = '<名称>'。 201
ThingWorx 请求移除项失败。该项已绑定且强制标志为 false。| 项名称 = '<名称>'。 201

U

UA 服务器成功从 UA Discovery Server 取消注册。| 端点 URL
'<端点 url>'。 196
UA 服务器成功注册到 UA Discovery Server。| 端点 URL
'<端点 url>'。 196
UA 服务器无法从 UA Discovery Server 取消注册。| 端点 URL
'<端点 url>'。 196
UA 服务器无法注册到 UA Discovery Server。| 端点 URL
'<端点 url>'。 196
UA 服务器证书过期。请使用 OPC UA Configuration Manager 重新颁发证书。 195

V

ViewServiceSet 167

W

Winsock 初始化失败。| OS 错误 = <错误>。 209
Winsock 关闭失败。| OS 错误 = <错误>。 209

嬾

安全 122
安全性 26, 31, 36-37, 43, 46, 51, 61, 111, 122, 140, 148

拂

按钮栏 40

燨

版本不匹配。 173

繶

绑定到本地端口时发生套接字错误。| 错误 = <错误>，详细信息 = '<信息>'。 209

佽

保存项目 122

墝

备份 26

彥

必须启用 FlexNet 许可服务才能处理许可证。启用服务失败将导致“限时”模式启动。 215

枧

标记管理 88
标记生成 73
标记生成请求未创建标记。有关详细信息，请参阅事件日志。 181
标记属性 - 常规 76
标记属性 - 缩放 82
标记组属性 84

凋

别名 85
别名属性 85

帖

并发客户端 140

尪

不超过扫描速率请求数据 71
不扫描，仅按需求轮询 71

荼

菜单栏 40

泫

测试项目 124

宽

尝试连接到运行时失败。| 运行时主机地址 = '<host address>', 用户 = '<name>', 原因 = '<reason>'。 186
尝试启动 OPC .NET 服务器超时。请验证服务器是否正在通过使用 OPC .NET Configuration Manager 运行。 195
尝试添加 DDE 项失败。| 项 = '<项名称>'。 196
尝试添加项 '<名称>' 失败。 190

败

超时前的尝试次数 75

憰

成员 141

抡

持久模式 28
持续数据存储 29

墳

处理阵列数据 133

冻

创建 74
创建和使用别名 131

嬉

存储并转发 - 完成率示例 57
存储并转发 - 系统标记 57
存储并转发服务 157

憩

打开 com 端口时出错。| 端口 = '<端口>', OS 错误 = <错误>。 208
打开加密的项目 124

勵

单个文件 28

廳

当前语言不支持加载 XML 项目。要加载 XML 项目，请在“服务器管理”中将产品语言选择更改为英语。 175

宜

导出 CSV 记录时出错。标记或组名称超出最大名称长度。| 记录索引 = '<number>', 最大名称长度 (字符数) = '<number>'。 188
导入 88
导入 CSV 标记时出现一般故障。 186
导入 CSV 记录时出错。\n\n“映射到”标记地址对于此项目无效。| 记录索引 = '<number>', 标记地址 = '<address>'。 188
导入 CSV 记录时出错。\n\n别名无效。名称不能包含双引号或以下划线开头。| 记录索引 = '<number>'。 188
导入 CSV 记录时出错。\n\n插入失败。| 记录索引 = '<number>', 记录名称 = '<name>'。 188
导入 CSV 记录时出错。\n\n字段缓冲区溢出。| 记录索引 = '<number>'。 187
导入 CSV 记录时出错。标记名称无效。| 记录索引 = '<number>', 标记名称 = '<name>'。 188
导入 CSV 记录时出错。标记组名称无效。| 记录索引 = '<index>', 组名称 = '<name>'。 189
导入 CSV 记录时出错。地址缺失。| 记录索引 = '<number>'。 189
导入 CSV 数据时出错。\n\n读取标识记录时字段缓冲区溢出。 187
导入 CSV 数据时出错。\n\n无法识别字段名称。| 字段 = '<名称>'。 187
导入 CSV 数据时出错。\n\n字段标识记录缺失。 187

导入 CSV 数据时出错。\n\n字段名称重复。| 字段 = '<名称>'。 187

癒

登录凭据 46, 123

嚥

地址 '<地址>' 在设备 '<名称>' 上无效。 211

璽

电话簿 108
电话簿必须至少有一个号码，通道才能自动拨号。| 通道 = '<channel>'。 180
电话号码无效 (<number>)。 182
电话号码优先级已更改。| 电话号码名称 = '<名称>', 更新的优先级 = '<priority>'。 184

削

动态标记 83

辶

逗号分隔变量 88

讌

读取只读项引用 '<名称>' 请求被拒绝。 192

窏

端口 28, 72

宙

对项 '<名称>' 的写入请求失败。缩放写入数据时出错。 192
对项 '<名称>' 的写入请求失败。写入数据类型 '<type>' 无法转换为标记数据类型 '<type>'。 192
对项引用 '<名称>' 的写入请求被拒绝，因为其所属的设备被禁用。 192
对于精简版 '<名称>' 驱动程序许可证，已超出最大驱动程序计数。编辑项目并重新启动服务器。 216
对于精简版 '<名称>' 驱动程序许可证，已超出最大通道计数。编辑项目并重新启动服务器。 217

对于精简版 '<数字>' 许可证，已超出最大设备计数。编辑项目并重新启动服务器。 214
对于精简版 '<数字>' 许可证，已超出最大运行时标记计数。编辑客户端项目并重新启动服务器。 214
对于重复标记 73

墺

多标记生成 79

危

发布项失败。多维数组不受支持。| 项名称 = '%s'。 199
发送数据时发生套接字错误。| 错误 = <错误>，详细信息 = '<信息>'。 212

教

方法 74

譟

访问对象被拒绝。| 用户 = '<account>', 对象 = '<object path>', 权限 = 184
访问管理菜单 23

雾

非持久 28

冦

分配到 <名称> 的配置会话已降级为只读。 193
分配到 <名称> 的配置会话已结束。 193
分配到 <名称> 的配置会话已升级为写入权限。 193

暝

服务器汇总信息 171
服务器选项 60

熋

父组 74

褦

覆盖 73

暔

更改运行时操作模式完成。 185
更新对象 '<名称>' 失败
<原因>。 191

刿

功能 '<名称>' 未获许可，因而无法使用。 213
功能 <名称> 存在时间限制，且将于 <日期/时间> 过期。 216

擰

故障时降级 71

儓

关闭以执行安装。 185, 193

笛

管理员密码已由当前用户重置。| 管理员名称 = '<名称>', 当前用户 = '<名称>'。 185

垚

基本服务器组件 62

剀

加密 36, 51, 111, 122, 124
加载此项目需要 <feature name>。 175
加载项目时遇到无效的型号。| 设备 = '<设备>'。 175

壙

夹具 82

閔

间隔 74

桠

检查可读性时发生套接字错误。| 错误 = <错误>，详细信息 = '<信息>'。 212
检查可写性时发生套接字错误。| 错误 = <错误>，详细信息 = '<信息>'。 212

笠

简介 15

宦

将属性更新推送到事物时出错。| 事物名称 = '<名称>'。 202
将一个或多个属性的推送类型设置为从不向平台推送更新。| 计数 = <计数>。 201

闭

降级超时 71
降级期间 71
降级时放弃请求 72

捅

接口和连接 18
接收数据时发生套接字错误。| 错误 = <错误>，详细信息 = '<信息>'。 212

覃

解密 124

轻

进程模式 17

雹

静态标记 (用户定义) 84

庠

开始新建项目 111

嬗

客户端应用程序已对设备 '<device>' 禁用自动降级。 184
客户端应用程序已对设备 '<device>' 启用自动降级。 183

朅

来自 ThingWorx Platform 的配置传输失败。 203
来自 ThingWorx Platform 的配置传输失败。| 原因 = '<原因>' 203
来自 ThingWorx 的配置传输已启动。 204
来自 ThingWorx 的配置传输已中止。 204
来自缓存的初始更新 71

簖

类型定义 141

轾

连接 28
连接超时 75
连接到服务器的 DNS/DHCP 设备重新上电时，解决通信问题 134
连接失败。无法绑定到适配器。| 适配器 = '<名称>'。 208
连接时发生套接字错误。| 错误 = <错误>，详细信息 = '<信息>'。 212
连接数据存储时文件损坏，已重新创建数据存储。| 数据存储路径 = '<路径>'。 206-207

历

另一应用程序正在使用 Com 端口。| 端口 = '<端口>'。 207

泯

浏览标记 118

輶

轮询 74

学

密码 31, 33, 46, 61, 111, 123-124

霖

面向 ThingWorx Platform 的配置传输失败。 203
面向 ThingWorx Platform 的配置传输失败。| 原因 = '<原因>' 203

厭

名称 70

槁

模板 88
模拟 70

鹼

默认值 42

儥

内存 28
内容检索 140
内置诊断 158

募

匿名 31, 46

郭

配置 API 服务 139
配置更改导致数据存储被覆盖。| 数据存储路径 = '<路径>'。 206-207
配置会话由 <名称> (<名称>) 启动。 193
配置实用程序不能与第三方配置应用程序同时运行。请关闭这两个程序，然后只打开要使用的程序。| 产品 = '<名称>'。 189

帏

平方根 82

讟

请求超时 75
请求间延迟 75

毢

求反 83

騑

驱动 70
驱动程序初始化失败。 208
驱动程序在处理标记期间可能不会更改此属性。 211
驱动程序在处理标记期间可能不会更改此页上的项。 211

曣

权限 33

绚

缺少服务器实例证书 '<证书位置>'。请使用 OPC UA Configuration Manager 重新颁发证书。 195

斅

日志设置 28
日志文件路径 28

儷

冗余 75

夢

如何... 130

戈

扫描模式 71
扫描速率覆盖 85

净

删除 73
删除对象 '<名称>' 失败
<原因>。 191

譿

设备 '<device>' 已禁用模拟模式。 183
设备 '<device>' 已禁用数据收集。 183
设备 '<device>' 已启用模拟模式。 183
设备 '<device>' 已启用数据收集。 183
设备 '<device>' 已自动降级。 178
设备 '<设备>' 的标记生成结果。| 创建的标记 = <计数>, 覆盖的标记 = <计数>。 184
设备 '<设备>' 的标记生成结果。| 创建的标记 = <计数>, 未覆盖的标记 = <计数>。 184
设备 '<设备>' 的标记生成结果。| 创建的标记 = <计数>。 184
设备创建向导 115
设备发现超出最大允许设备数 <count>。请限制发现范围并重试。 174
设备启动时 73
设备上的指定地址无效。| 无效地址 = '<地址>'。 211
设备属性 - 标记生成 73
设备属性 - 常规 69
设备属性 - 以太网封装 72
设备属性 - 自动降级 71
设备未响应。 209
设备未响应。| ID = '<设备>'。 210
设备需求轮询 138
设计项目 111
设置 24
设置 - ProgID 重定向 29
设置 - 服务端口 39
设置 - 管理 24
设置 - 配置 24
设置 - 配置 API 服务配置 35
设置 - 事件日志 27
设置 - 用户管理器 30, 34
设置 - 运行时进程 25
设置 - 运行时选项 26
设置 - 证书存储 37

习

什么是标记？ 76
什么是标记组？ 84
什么是别名映射？ 85
什么是设备？ 69
什么是事件日志？ 86
什么是信道？ 62

璿

生成 73
生成多个标记 119

童

时间同步 74
时间同步阈值 74
时区 74

伟

使用别名来优化项目 135
使用动态标记寻址 136
使用以太网封装 136

丫

事件 42
事件日志显示 86
事件日志消息 171

对

属性编辑器 42
属性标记 100
属性定义 142
属性更改时 73
属性类型 143

攘

数据 145
数据收集 70

謶

说明 70

绉

缩放 82

仳

体系结构 139

暟

替换项目的请求被拒绝，因为它与正在使用的项目相同
'<名称>'。 190

涛

添加标记缩放 121
添加对象到 '<名称>' 失败
<原因>。 191
添加和配置设备 114
添加和配置信道 112
添加静态文本 80
添加数字范围 80
添加文本序列 80
添加项时出错。| 项名称 = '<项名称>'。 198
添加用户定义标记 116

讣

调制解调器标记 103
调制解调器线路关闭
'<modem>'。 182
调制解调器线路开通
'<modem>'。 182

辺

通道不再有效，可能在等待用户输入时被移除。| 通道 = '<名称>'。 187
通道的设备发现已取消。| 通道 = '<名称>', 发现的设备 = '<计数>'。 190
通道的设备发现已取消。| 通道 = '<名称>'。 189
通道的设备发现已完成。| 通道 = '<名称>', 发现的设备 = '<计数>'。 189
通道分配 70
通道上出现串行通信错误。| 错误掩码 = <掩码>。 210
通道需要启用自动拨号和其电话簿中的至少一个号码，以使用共享调制解调器连接。| 通道 = '< 通道>'。 180
通信参数 72
通信超时 75
通信管理 107
通信序列化标记 105
通信诊断 168

繼

统计信息标记 101

为 '<device>' 设备设置的时区是 '<zone>'。对于系统而言，这是无效的时区。正在将时区更改为默认时区 '<zone>'。 180

曊

未对线路 '<line>' 的连接提供通信处理。 177
未加载设备驱动程序 DLL。 187, 190
未连接到事件记录器服务。 190
未能保存嵌入的相关文件。| 文件 = '<路径>'。 189
未能打开调制解调器线路 '<line>' [TAPI 错误 = <code>]。 173
未能读取构建清单资源
<名称>。 184
未能更新启动项目 '<名称>'
<原因>。 191
未能加载库
<名称>。 184
未能检索运行时项目。 187
未能启动通道诊断。 188
未能替换通道上的现有自动生成设备，删除失败。| 通道 = '<名称>'。 187
未能添加标记 '<tag>'，因为地址太长。最大地址长度为 <number>。 176
未能重置管理员密码。| 管理员名称 = '<名称>'。 186
未找到 <feature name>，或者无法加载。 174

未找到 <名称> 设备驱动程序，或者无法加载。 172
未找到或无法加载设备驱动程序。| 驱动程序 = '<名称>'。 187

擷

文件名包含一个或多个无效字符。 191
文件名不能覆盖现有文件
'<name>'。 190
文件名不能为空白。 191
文件名的格式必须是 <subdir>/<name>.{json,opf,sopf} 191

斀

无法保存项目文件 <名称> 174
无法创建串行 I/O 线程。 208
无法打开项目文件
'<名称>'。 190
无法导入服务器实例证书
'<证书位置>'。请使用 OPC UA Configuration Manager 重新颁发证书。 195
无法对线路 '<line>' 应用调制解调器配置。 178
无法访问功能 <名称> 的许可证 [错误=<代码>]，必须重新将其激活。 216
无法加载 '<名称>' 驱动程序，因为存在多个副本 ('<名称>' 和 '<name>')。请移除冲突的驱动程序并重新启动应用程序。 172
无法加载插件 DLL '<名称>'。 179
无法加载插件 DLL '<名称>'。原因 180
无法加载启动项目 '<名称>'
<原因>。 191
无法加载驱动程序 DLL '<名称>'。 178
无法加载驱动程序 DLL '<名称>'。原因 180
无法加载项目 <名称> 174
无法加载许可证界面，可能是因为缺失第三方依赖项。仅在“限时”模式下运行。 213
无法将项目备份到 '<路径>' [<原因>]。保存操作已中止。请验证目标文件未锁定并且具有读/写权限。要继续保存此项目而不进行备份，请在“工具”|“选项”|“常规”下取消选择备份选项，然后重新保存项目。 174
无法开始对通道进行设备发现。| 通道 = '<名称>'。 190
无法连接到现有的数据存储，因为该数据存储由较早版本的服务器创建。已重新创建数据存储。| 数据存储路径 = '<路径>'。 206-207
无法连接或附加到“存储并转发”数据存储。正在使用内存存储。| 内存存储大小 (更新数) = <计数>。 202
无法启动 Config API 服务。绑定到端口时可能出现了问题。 194, 197
无法启动“存储并转发”服务器。绑定到本地端口时出现 socket 错误。| 错误 = <错误>, 详细信息 = '<信息>'。 205-206
无法启动应用程序。| 应用程序 = '<路径>', 操作系统错误 = '<code>'。 188
无法启用对此服务器的 iFIX PDB 支持。| OS 错误 = '<错误>'。 196
无法删除“存储并转发”数据存储中已存储的更新。 203
无法使用通道 '<name>' 上的网络适配器 '<adapter>'。正在使用默认网络适配器。 177
无法使用指定参数配置 com 端口。| 端口 = COM<数字>，OS 错误 = <错误>。 208

无法替换通道上的设备，因为该设备有活动的引用计数。| 通道 = '<名称>'。 187
无法添加设备。此通道中可能存在重复设备。 175
无法添加项目。请求的 <数字> 计数会超过 <最大计数> 的许可证限制。 217
无法为设备 '<device>' 生成标记数据库 176
无法为设备 '<device>' 生成标记数据库。设备未响应。 175
无法写入设备 '<名称>' 上的地址 '<地址>'。 211
无法写入设备上的地址。| 地址 = '<地址>'。 210
无法写入项 '<名称>'。 192
无法写入项。| 项 = '<项名称>'。 197
无法应用由 ThingWorx Platform 引起的设置更改。权限被拒绝。| 用户 = '<用户名>'。 202
无法在平台上触发自动绑定完成事件。 198
无法在线路 '<line>' 拨号。 177
无法重启“事物”。| 名称 = '<事物名称>'。 200
无法自动绑定属性。| 名称 = '<属性名称>'。 200
无效的 ProgID 条目已从 ProgID 重定向列表删除。| ProgID = '<ID>'。 185
无效的 XML 文档 173, 188
无效的项目文件
'<名称>'。 190

肅

系统标记 90
系统服务 148
系统要求 16
系统正在拒绝对引用设备 '<channel device>' 上的型号类型的更改。 177

憧

夏令时 74

繟

线路 '<line>' 存在硬件错误。 176
线路 '<line>' 已使用。 176
线路 '<modem>' 的拨号已被用户取消。 182
线路 '<modem>' 检测到呼入。 182
线路 '<modem>' 已断开连接。 181
线路 '<modem>' 已连接，速率 <rate> 波特。 182
线路 '<modem>' 已连接。 183
线路在远程站点的 '<modem>' 断开。 182
线性 82

響

详细信息视图 42

唯

响应码 150

頊

项目不允许对象类型 '<name>'。 183
项目属性 43
项目属性 - DDE 46
项目属性 - FastDDE/SuiteLink 48
项目属性 - iFIX PDB 设置 49
项目属性 - OPC .NET 47
项目属性 - OPC AE 48
项目属性 - OPC DA 合规 43
项目属性 - OPC HDA 50
项目属性 - OPC UA 45
项目属性 - ThingWorx 本机接口 51
项目属性 - 标识 43
项目树状视图 41
项目文件无效。 173
项目文件由此软件的最新版本创建。 184

勣

协议 72

儹

写入标记 <设备名称>.<地址> 时检测到无效的数组大小。 210
写入属性失败。| 事物名称 = '<名称>', 属性名称 = '<名称>', 原因 = <原因>。 201
写入属性失败。| 属性名称 = '<名称>', 原因 = <原因>。 200

侁

信道创建向导 112

托

型号 70

藺

虚拟网络模式已更改。这会影响所有通道和虚拟网络。有关虚拟网络模式的详细信息，请参阅帮助。| 新模式 = '<mode>'。 189

辩

选项 - 常规 60
选项 - 运行时连接 61
选择合适的网线 134

騬

验证错误位于 '<tag>'
<error>。 178
缩放参数无效。 178

褡

要求组件 <名称> (<版本>) 版本与组件 <名称> (<版本>) 版本相匹配。 217
要使用此驱动程序，必须安装 Winsock V1.1 或更高版本。 209

禛

移动对象 '<名称>' 失败
<原因>。 191

嶒

已成功从“存储并转发”数据存储中删除已存储的数据。 205
已成功从服务控制管理器数据库移除 <名称>。 192
已成功配置 <名称> 作为系统服务运行。 192
已初始化“存储并转发”数据存储。| 数据存储位置 '<位置>'。 204
已初始化“存储并转发”数据存储。| 转发模式 = '<模式>' | 数据存储位置 = '<位置>'。 205
已处理一个或多个自动绑定请求。| 计数 = <计数>。 204
已达到“存储并转发”数据存储大小限制。 199
已更新启动项目 '<名称>'。 192

已恢复将属性更新推送到事物
错误条件已解决。| 事物名称 = '<名称>'。 204
已将服务器配置为针对每次扫描发送更新，但将一个或多个属性的推送类型设置为仅在值更改时推送。| 计数 = <计数>。 201
已将项目 '<名称>' 的备份创建到 '<路径>'。 183
已连接至 ThingWorx。| 平台 = <主机
端口资源>，事物名称 = '<名称>'。 203
已完成设备 '<device>' 的自动标记生成。 183
已自动升级设备 '<device>' 以确定是否可以重新建立通信。 184

丿

以扫描速率请求所有数据 71

嚀

因 Configuration API 引起了项目设置更改而正在重新初始化 ThingWorx 连接。 204
因接口关闭或重新初始化而正在放弃待处理的自动绑定。| 计数 = <计数>。 204
因连接缓冲区中空间不足，一或多个值更改更新丢失。| 丢失更新数 = <计数>。 198
因平台引起了项目设置更改而正在重新初始化 ThingWorx 连接。 204

帴

应用 42
应用程序数据目录缺失。 193
应用于配置会话的权限更改已分配到 <名称>。 194

旂

映射到 85

璨

用户界面导航 40
用户密码已更改。| 用户 = '<名称>'。 186
用户权限不足以替换运行时项目。 186
用户信息无效或缺失。 186
用户信息已替换为导入内容。| 导入的文件 = '<absolute file path>'。 185
用户已从用户组转移。| 用户 = '<名称>', 旧组 = '<名称>', 新组 '<name>'。 185
用户已禁用。| 用户 = '<名称>'。 186
用户已启用。| 用户 = '<名称>'。 186
用户已添加到用户组。| 用户 = '<名称>', 组 = '<名称>'。 185
用户已重命名。| 旧名称 = '<名称>', 新名称 = '<名称>'。 186

用户组的权限定义已更改。| 组 = '<名称>'。 185
用户组已创建。| 组 = '<名称>'。 185
用户组已禁用。| 组 = '<名称>'。 186
用户组已启用。| 组 = '<名称>'。 186
用户组已重命名。| 旧名称 = '<名称>', 新名称 = '<名称>'。 185

辨

优化服务器项目 133

瓦

由于磁盘已满，“存储并转发”数据存储无法存储数据。 199
由于存在启用的连接，已忽略关闭请求。| 启用的连接 = '<count>'。 189
由于驱动程序级别故障，无法添加设备。 173
由于驱动程序级别故障，无法添加通道。 173
由于缺少对象，无法加载项目。| 对象 = '<object>'。 175
由于数据存储路径无效，连接到数据存储时出错。| 路径 = '<路径>'。 205-206
由于文件 IO 错误或数据存储损坏，“存储并转发”数据存储已重置。 202

癸

与 ThingWorx 的连接已关闭。| 平台 = <主机
端口资源>。 199
与 ThingWorx 连接失败，原因未知。| 平台 = <主机
端口资源>，错误= <错误>。 198
与 ThingWorx 连接失败。| 平台 <主机
端口资源>，错误
<原因>。 197
与远程运行时同步失败。 188

頤

预览 80

匿

原始 82

轼

远端线路在 '<modem>' 未应答。 182
远端线路在 '<modem>' 占线。 182

傯

允许桌面交互作用 130
允许子组 74

轰

运行服务器 111
运行时服务已启动。 192
运行时关闭完成。 193
运行时进程已启动。 193
运行时项目从 '<名称>' 替换。 193
运行时项目更新失败。 187
运行时项目已另存为 '<名称>'。 193
运行时项目已替换。 193
运行时项目已替换。| 新项目 = '<路径>'。 190
运行时项目已替换为定义的启动项目。运行时项目将在下次重新启动时从 '<名称>' 恢复。 191
运行时项目已重置。 190
运行时正在退出。 193
运行时重新初始化已启动。 192
运行时重新初始化已完成。 192

瀞

灾难恢复 27

囈

在 <名称> 上，已超出特征计数限制。受时间限制的使用将于 <日期/时间> 过期。 216
在服务器上使用 DDE 135
在服务器项目中使用调制解调器 107
在功能 '<名称>' 上，已超出 <最大计数> 的 <对象类型名称> 限制。 215
在功能 '<名称>' 上，已超出 <最大计数> 的类型 <数字类型 ID> 限制。 214
在功能 <名称> 上，受时间限制的使用已过期。 216
在功能 <名称> 上，已超出对象计数限制。受时间限制的使用将于 <日期/时间> 过期。 216
在禁用项目文件加密的情况下保存项目文件，因为不支持 .SOPF 文件类型。支持的文件类型为 .OPF 和 .JSON。 191
在启用项目文件加密的情况下保存项目文件，因为不支持 .OPF 文件类型。支持的文件类型为 .SOPF 和 .JSON。 191

锟

长期数据存储 28

但

侦听客户端连接时发生套接字错误。| 端点 URL = '<端点 URL>'，错误 = <错误代码>，详细信息 = '<说明>'。 195

欖

正确命名通道、设备、标记和标记组 134
正在尝试对设备 '<device>' 自动生成标记。 183
正在打开项目。| 项目 = '<名称>'。 189
正在更改运行时操作模式。 184
正在关闭项目。| 项目 = '<名称>'。 189
正在关闭以执行安装。 190
正在忽略用户定义的启动项目，因为配置会话处于活动状态。 192
正在开始对调制解调器线路 '<modem>' 断开连接。 183
正在开始对通道进行设备发现。| 通道 = '<名称>'。 189
正在启动 <名称> 设备驱动程序。 181
正在停止 <名称> 设备驱动程序。 181
正在线路 '<modem>' 拨号 '<number>'。 181

譜

证书 38

抰

指定的区域无效。无法设置订阅筛选器。| 区域 = '<区域名称>'。 197
指定的网络适配器在通道 '%1' 上无效。| 适配器 = '%2'。 181
指定的源无效。无法设置订阅筛选器。| 源 = '<源名称>'。 197

酳

重命名失败。名称不能包含句点、双引号或以下划线开头。| 建议名称 = '<名称>'。 188
重命名失败。已存在具有该名称的对象。| 建议名称 = '<名称>'。 188
重置管理员密码失败。当前用户不是 Windows 管理员。| 管理员名称 = '<名称>', 当前用户 = '<名称>'。 186

賤

资源 219

膊

自动拨号 109
自动生成的标记 '<tag>' 已存在，无法覆盖。 175
自动生成造成的覆盖太多，已停止发布错误消息。 176

纂

组件 16

逕

遵循标签指定的扫描速率 71
遵循客户端指定的扫描速率 71